

Università degli Studi di Pisa

Facoltà di Scienze Matematiche, Fisiche e Naturali
Corso di Laurea in Informatica

**Definizione di un MIB per un sistema di
Controllo Ambientale
(Environmental Control)**

Esame di Sistemi di Gestione Reti

Prof. Luca Deri

Giacomo Rutili: g.rutili@inwind.it
Riccardo Verona: riqui77@virgilio.it

INDICE

| | |
|--|---------------|
| 1. Introduzione | - 3 - |
| 2. Caratteristiche del sistema | - 5 - |
| 3. Scelte implementative | - 6 - |
| 3.1 Note | - 6 - |
| 4. Definizione MIB | - 7 - |
| 4.1 Note | - 22 - |
| 5. Conclusioni | - 23 - |
| 6. Bibliografia | - 23 - |
| Testi di Riferimento: | - 23 - |
| Modelli di Environmental Control di riferimento ed altri link utili: | - 23 - |

1. Introduzione

Per Controllo Ambientale, o Environmental Control, si intende il monitoraggio e il controllo simultaneo di caratteristiche ambientali all'interno di uno spazio chiuso, al fine di soddisfare le esigenze di personale, servizi o equipaggiamenti.

Un sistema di questo genere può rilevare le più piccole variazioni (subordinate alla sensibilità dei sensori disponibili) di uno dei parametri, modificarne manualmente alcuni, generare degli allarmi quando questi superano delle soglie prestabilite, o segnalare dei guasti.

L'utilizzo di questa tecnologia può essere svariato e si presume che nel futuro ambienti pubblici, uffici, case private potrebbero avere un sistema di questo tipo; il controllo automatizzato dell'ambiente permetterà un notevole risparmio energetico ottimizzando tutte le risorse disponibili (fonti luminose, climatizzazione) ed una maggiore sicurezza in caso di imprevisti o incidenti (allagamenti, incendi, intrusi).

Lo scopo del progetto è di creare quindi un MIB di un sistema remoto di Environmental Control per un ambiente che custodisce delle risorse.

Per questioni di realistica, abbiamo supposto che il sistema di monitoraggio ambientale si renda auspicabile (se non necessario) quando la risorsa custodita presenta queste due importanti caratteristiche:

- **la risorsa ha un valore intrinseco molto alto, sia in termini economici che strategici**
- **la risorsa è soggetta a variazioni sensibili di prestazioni , rottura o deperimento fisico qualora non sia garantita la stabilità di fattori ambientali in un range prefissato**

Queste due caratteristiche ci sembrano necessarie dato che allo stato attuale della tecnologia, sia lo sviluppo di un software adatto al monitoraggio, sia i sensori preposti al controllo, possono raggiungere costi decisamente elevati; abbiamo supposto quindi che sia realistico il loro utilizzo in casi in cui il possibile danno o un calo delle prestazioni sia a lungo termine più costoso del sistema di controllo stesso.

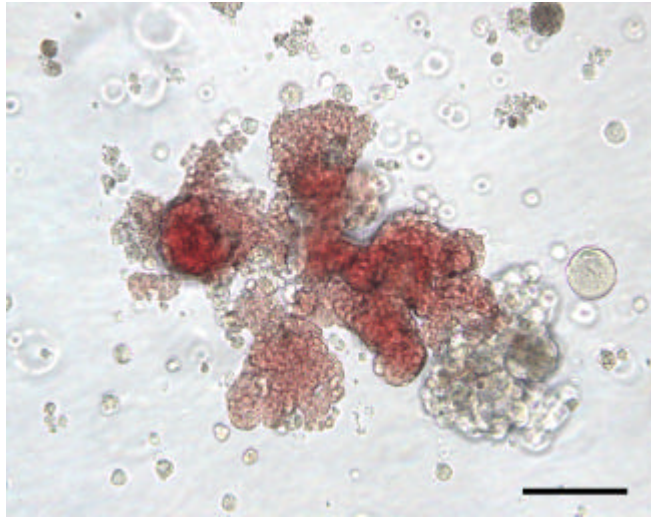
Esempi pratici di campi d'applicazione possono essere:

Stanza Server - PoP (Point of Presence) - Centrali per telecomunicazioni



Le prestazioni e la continuità del servizio sono elementi di vitale importanza e possono essere compromesse da vari fattori: temperatura ed un livello di umidità eccessivi, incendi, allagamenti, accessi non autorizzati a dati riservati o di importanza strategica e non ultimi sbalzi o interruzioni di corrente.

Centri di ricerca Biologica / Chimica - Ditte Farmaceutiche



La temperatura e l'umidità devono essere costantemente sotto controllo, in quanto anche una piccola variazione delle suddette può creare alterazioni delle reazioni chimiche (l'innalzamento della temperatura velocizza le reazioni chimiche) o compromettere la conservazione delle sostanze. Anche in questi casi la sicurezza è un fattore vitale per evitare lo spionaggio industriale e garantire la riservatezza delle ricerche

Caveau di una banca - Ambienti di Museo contenenti opere d'arte di grande valore



Nel caso del caveau la sicurezza è sicuramente un aspetto da privilegiare, mentre nel caso di un museo, dobbiamo tenere conto che molte opere d'arte sono composte da materiale altamente deperibile (tela, legno, colori ad olio) e molto sensibile alle variazioni di temperatura, umidità e in alcuni casi luce.

2. Caratteristiche del sistema

Il sistema utilizza un paradigma manager/agent, dando la possibilità ad una centrale remota di controllo (manager) di avere, periodicamente o su richiesta, uno “screenshot” delle condizioni dell’ambiente. Questa fase di “polling” permette al manager di tenere costantemente sotto controllo l’ambiente, di fare statistiche per controllare l’andamento temporale dei parametri o di modificarne manualmente alcuni; è possibile ad esempio inviare un tecnico per la pulizia del filtro del condizionatore prima l’impianto si blocchi, oppure ottenere un grafico con l’andamento della temperatura durante la giornata o aumentare la sensibilità dei dispositivi anti-intrusione durante l’orario non lavorativo.

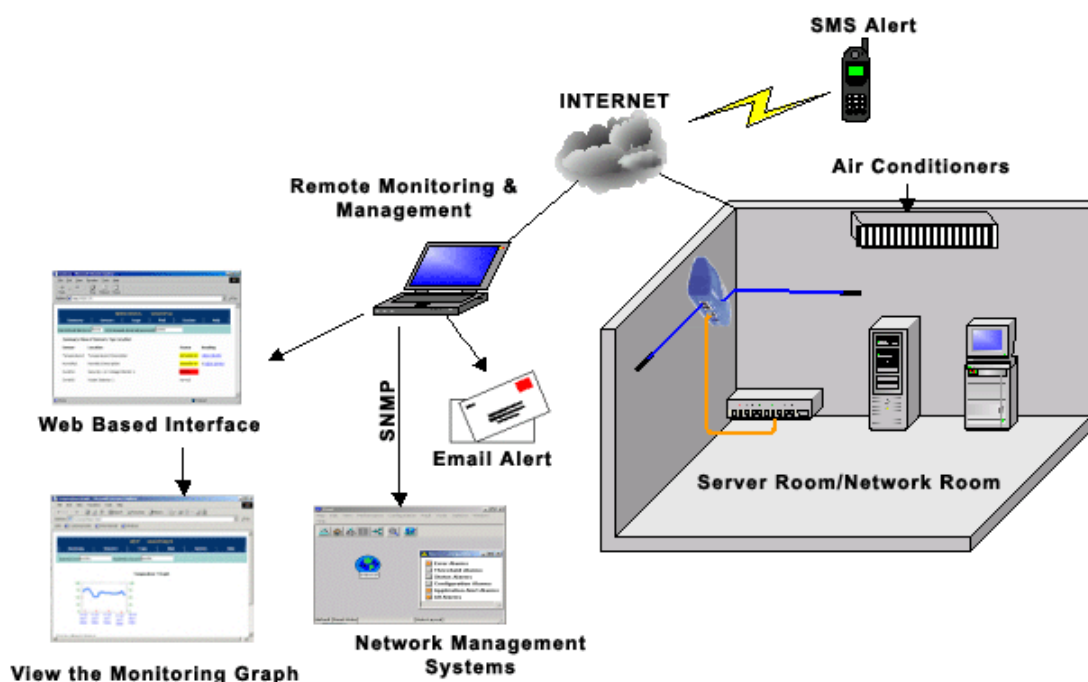
Ci sono inoltre delle situazioni particolari in cui il sistema di controllo presente nell’ambiente (agent) invia autonomamente degli allarmi al centro di controllo. Nel nostro caso specifico le situazioni che possono generare degli allarmi sono tre:

- nel caso in cui ci sia un malfunzionamento di uno dei componenti
- se viene raggiunta o superata qualche soglia prefissata
- nel momento in cui si tenta un’accesso all’ambiente ad esempio attraverso accessi autorizzati e non

Nell’immagine sottostante possiamo vedere come esempio una server room con impianto di condizionamento controllato in remoto via SNMP; dal lato manager si ha la possibilità attraverso una GUI di impostare o testare la temperatura e lo stato del condizionatore mentre l’agent può inviare degli allarmi in caso di problemi al filtro.

Nel caso specifico l’allarme può essere inoltre notificata anche via e-mail o SMS.

sensorProbe in computer room/network room monitoring



3. Scelte implementative

Come caso pratico abbiamo ritenuto opportuno considerare un ambiente come una Server Room, avendo constatato attraverso una ricerca in Internet, che l'Environmental Control in questa particolare situazione è un problema di interesse attuale e che sono molte le aziende che stanno creando soluzioni ad hoc: la AKCP, la OPTO22 e la MicroSystem per i PoP di Fastweb.

In questo tipo di ambiente sbalzi eccessivi di temperatura o d'umidità, la presenza di incendi o allagamenti, possono seriamente danneggiare le apparecchiature con conseguenti danni economici, sia per il ripristino dei servizi che per il rimpiazzo delle macchine danneggiate.

Un occhio di riguardo va posto anche sulla questione della sicurezza, in quanto un accesso non autorizzato può portare a manomissioni e sabotaggi, furti e spionaggio industriale.

Per una maggiore chiarezze e un più facile aggiornamento la struttura del MIB è stata divisa logicamente in cinque aree funzionali:

- **datiGenerali:** riassume tutte le variabili generiche del sistema, le informazioni sull'ambiente e sui server contenuti.
- **controlloAmbientale:** contiene tutte le informazioni per il monitoraggio delle condizioni ambientali per la salvaguardia ed il corretto funzionamento dei server.
- **sistemaSicurezza:** contiene le informazioni per i sottosistemi di sicurezza come il controllo anti-intrusione e le politiche d'accesso autorizzate.
- **controlloAlimentazione:** contiene le informazioni per i generatori di corrente che garantiscono la continua alimentazione e quindi un corretto funzionamento sia dei server che del sistema stesso.
- **allarmi:** gli allarmi possono essere di tre tipi: allarme, guasto e superamento di una soglia.
- **envControlConfermita:** indica le informazioni per la conformità del MIB

3.1 Note

- Non è stato possibile utilizzare lettere accentate all'interno delle descrizioni degli oggetti.
- La massiccia presenza di sensori nel sistema deriva in maniera necessaria dai presupposti presentati nell'introduzione

4. Definizione MIB

```

ENVIRONMENTAL-CONTROL-MIB DEFINITIONS ::= BEGIN

IMPORTS
    DisplayString,
    DateAndTime
        FROM SNMPv2-TC
    OBJECT-GROUP,
    NOTIFICATION-GROUP,
    MODULE-COMPLIANCE
        FROM SNMPv2-CONF
    enterprises,
    MODULE-IDENTITY,
    OBJECT-TYPE,
    NOTIFICATION-TYPE,
    Integer32,
    Unsigned32
        FROM SNMPv2-SMI;

environmentalControlMIB MODULE-IDENTITY
    LAST-UPDATED "200307261300Z"
    ORGANIZATION "Imlradis Corp."
    CONTACT-INFO
        "Giacomo Rutili
        Massa, ITALY
        g.rutili@inwind.it

        Riccardo Verona
        Pisa, ITALY
        riqui77@virgilio.it
        "
    DESCRIPTION
        "MIB per il controllo ambientale e della sicurezza di oggetti di
        grande interesse economico e/o strategico"
    ::= { enterprises 1 }

datiGenerali OBJECT IDENTIFIER      ::= { environmentalControlMIB 1 }

idSistema OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID univoco che identifica lo specifico sistema di controllo"
    ::= { datiGenerali 1 }

indirizzo OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Indirizzo fisico dell'ambiente sotto controllo"
    ::= { datiGenerali 2 }

responsabile OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Nome e cognome del Responsabile dell'ambiente"
    ::= { datiGenerali 3 }

contenutoAmbiente OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION

```

```

        "L'oggetto, piu l'eventuale servizio associato,
        contenuto all'interno dell'ambiente controllato"
 ::= { datiGenerali 4 }

```

```

superficieAmbiente OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Superficie, misurata in metri quadri, dell'ambiente"
 ::= { datiGenerali 5 }

```

```

volumeAmbiente OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Volume dell'ambiente, misurato in metri cubi"
 ::= { datiGenerali 6 }

```

```

noteAggiuntive OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Note aggiuntive sull'ambiente o sul suo contenuto"
 ::= { datiGenerali 7 }

```

```

dataInstallazione OBJECT-TYPE
    SYNTAX DateAndTime
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Data di installazione del sistema all'indirizzo indicato"
 ::= { datiGenerali 8 }

```

```

controlloAmbientale OBJECT IDENTIFIER ::= { environmentalControlMIB 2 }

```

```

sensoreFumoTable OBJECT-TYPE
    SYNTAX SEQUENCE OF SensoreFumoEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Tabella dei sensori per rilevare la presenza di fumo"
 ::= { controlloAmbientale 1 }

```

```

sensoreFumoEntry OBJECT-TYPE
    SYNTAX SensoreFumoEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella relativo al singolo sensore"
    INDEX {
        idSensoreFumo }
 ::= { sensoreFumoTable 1 }

```

```

SensoreFumoEntry ::= SEQUENCE {
    idSensoreFumo Unsigned32,
    sensibilitaSensoreFumo Unsigned32,
    statoSensoreFumo INTEGER }

```

```

idSensoreFumo OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID del singolo sensore"
 ::= { sensoreFumoEntry 1 }

```

```
sensibilitaSensoreFumo OBJECT-TYPE
    SYNTAX Unsigned32 (0..10)
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Sensibilita del sensore regolabile manualmente da 0 a 10."
    ::= { sensoreFumoEntry 2 }
```

```
statoSensoreFumo OBJECT-TYPE
    SYNTAX INTEGER {
        inattivo(0),
        attivo(1),
        allarme(2) }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "stato del sensore"
    ::= { sensoreFumoEntry 3 }
```

```
fonteiLuminosaTable OBJECT-TYPE
    SYNTAX SEQUENCE OF FonteLuminosaEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Tabella delle fonti luminose presenti nell'ambiente. Possono
        essere artificiali o regolazioni della luce naturale esterna"
    ::= { controlloAmbientale 2 }
```

```
fonteLuminosaEntry OBJECT-TYPE
    SYNTAX FonteLuminosaEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella relativa al signolo sensore"
    INDEX {
        idFonte }
    ::= { fonteiLuminosaTable 1 }
```

```
FonteLuminosaEntry ::= SEQUENCE {

    idFonte             Unsigned32,
    statoFonte          INTEGER,
    intensitaFonte      Integer32,
    classificazioneFonte  INTEGER,
    collocazioneFonteLuminosa DisplayString }
```

```
idFonte OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "id della fonte luminosa"
    ::= { fonteLuminosaEntry 1 }
```

```
statoFonte OBJECT-TYPE
    SYNTAX INTEGER {
        funzionante(0),
        guasta(1) }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "indicatore di stato della fonte luminosa"
```

```
::= { fonteLuminosaEntry 2 }
```

intensitaFonte OBJECT-TYPE

SYNTAX Integer32 (0..10)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"indica l'intensita della fonte luminosa: 0 = fonte spenta 10 = max intensita"

```
::= { fonteLuminosaEntry 3 }
```

classificazioneFonte OBJECT-TYPE

SYNTAX INTEGER {

artificialePrincipale(0),

artificialeEmergenza(1),

solare(2) }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indica il tipo di fonte luminosa"

```
::= { fonteLuminosaEntry 4 }
```

collocazioneFonteLuminosa OBJECT-TYPE

SYNTAX DisplayString

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"indica la collocazione della fonte luminosa"

```
::= { fonteLuminosaEntry 5 }
```

sensoresTemperatura OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Rileva la temperatura attuale"

```
::= { controlloAmbientale 5 }
```

sogliaTemperaturaMax OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Soglia di temperatura oltre la quale scatta un allarme"

```
::= { controlloAmbientale 6 }
```

sogliaTemperaturaMin OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Soglia sulla temperatura al di sotto della quale scatta un allarme"

```
::= { controlloAmbientale 7 }
```

sensoresAcquaTable OBJECT-TYPE

SYNTAX SEQUENCE OF SensoreAcquaEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Tabelle dei sensori rilevatori di acqua"

```
::= { controlloAmbientale 8 }
```

```

sensoreAcquaEntry OBJECT-TYPE
    SYNTAX SensoreAcquaEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella dei rilevatori di acqua"
    INDEX {
        idSensoreAcqua }
    ::= { sensoreAcquaTable 1 }

SensoreAcquaEntry ::= SEQUENCE {

    idSensoreAcqua    Unsigned32,
    statoSensoreAcqua INTEGER }

idSensoreAcqua OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID del rilevatore di presenza di acqua"
    ::= { sensoreAcquaEntry 1 }

statoSensoreAcqua OBJECT-TYPE
    SYNTAX INTEGER {
        inattivo(0),
        attivo(1),
        allarme(2) }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Stato del sensore acqua. Non e prevista una sensibilita
        in quanto la presenza di acqua anche in minime parti deve
        generare un allarme"
    ::= { sensoreAcquaEntry 2 }

sensoreLuceTable OBJECT-TYPE
    SYNTAX SEQUENCE OF SensoreLuceEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Tabella relativa ai sensori luminosi"
    ::= { controlloAmbientale 9 }

sensoreLuceEntry OBJECT-TYPE
    SYNTAX SensoreLuceEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella dei sensori d'illuminazione"
    INDEX {
        idSensoreLuce }
    ::= { sensoreLuceTable 1 }

SensoreLuceEntry ::= SEQUENCE {

    idSensoreLuce        Unsigned32,
    risultatoSensoreLuce Unsigned32,
    collocazioneSensoreLuce DisplayString,
    sensibilitaSensoreLuce Integer32,
    sogliaIlluminazioneMax Unsigned32 }

```

```

idSensoreLuce OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID del sensore luminoso"
    ::= { sensoreLuceEntry 1 }

risultatoSensoreLuce OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "risultato della rilevazione dal sensore"
    ::= { sensoreLuceEntry 2 }

collocazioneSensoreLuce OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "indica la collocazione del sensore luminoso"
    ::= { sensoreLuceEntry 3 }

sensibilitaSensoreLuce OBJECT-TYPE
    SYNTAX Integer32 (0..10)
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "indica la precisione del rilevatore di luce, da 0 a 10"
    ::= { sensoreLuceEntry 4 }

sogliaIlluminazioneMax OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Soglia d'illuminazione oltre la quale scatta un allarme"
    ::= { sensoreLuceEntry 5 }

climatizzatoreTable OBJECT-TYPE
    SYNTAX SEQUENCE OF ClimatizzatoreEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "tabella dei dati relativi ai climatizzatori"
    ::= { controlloAmbientale 10 }

climatizzatoreEntry OBJECT-TYPE
    SYNTAX ClimatizzatoreEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella dei climatizzatori"
    INDEX {
        idClimatizzatore }
    ::= { climatizzatoreTable 1 }

ClimatizzatoreEntry ::= SEQUENCE {
    idClimatizzatore          Unsigned32,
    temperaturaClimatizzatore INTEGER,
    statoClimatizzatore      INTEGER,

```

```
puliziaFiltroClimatizzatore INTEGER,
collocazioneClimatizzatore DisplayString }
```

idClimatizzatore **OBJECT-TYPE**

```
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "ID del climatizzatore"
 ::= { climatizzatoreEntry 1 }
```

temperaturaClimatizzatore **OBJECT-TYPE**

```
SYNTAX INTEGER (5..30)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "Temperatura attuale del climatizzatore. Regolabile da min 5 a max 30"
 ::= { climatizzatoreEntry 2 }
```

statoClimatizzatore **OBJECT-TYPE**

```
SYNTAX INTEGER {
    spento(0),
    acceso(1),
    guasto(2) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Stato del climatizzatore"
 ::= { climatizzatoreEntry 3 }
```

puliziaFiltroClimatizzatore **OBJECT-TYPE**

```
SYNTAX INTEGER {
    pulito(0),
    sporco(1) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Stato di pulizia del filtro"
 ::= { climatizzatoreEntry 4 }
```

collocazioneClimatizzatore **OBJECT-TYPE**

```
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Collocazione del climatizzatore"
 ::= { climatizzatoreEntry 5 }
```

sensoresUmidita **OBJECT-TYPE**

```
SYNTAX Integer32 (0..100)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Indica la percentuale d'umidita nell'intera stanza"
 ::= { controlloAmbientale 11 }
```

sogliaUmiditaMax **OBJECT-TYPE**

```
SYNTAX Integer32 (0..100)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "soglia d'umidita oltre la quale scatta un allarme"
 ::= { controlloAmbientale 12 }
```

```

sogliaUmiditaMin OBJECT-TYPE
    SYNTAX Integer32 (0..100)
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "soglia d'umidita sotto la quale scatta un allarme"
    ::= { controlloAmbientale 13 }

sistemaSicurezza OBJECT IDENTIFIER      ::= { environmentalControlMIB 3 }

sensoreContattoTable OBJECT-TYPE
    SYNTAX SEQUENCE OF SensoreContattoEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Tabella descrittiva dei sensori per rilevare tentativi d'accesso
        da vie non autorizzate"
    ::= { sistemaSicurezza 1 }

sensoreContattoEntry OBJECT-TYPE
    SYNTAX SensoreContattoEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella dei sensori anti-intrusione a contatto"
    INDEX {
        idSensoreContatto }
    ::= { sensoreContattoTable 1 }

SensoreContattoEntry ::= SEQUENCE {

    idSensoreContatto    Unsigned32,
    tipoSensoreContatto  INTEGER,
    statoSensoreContatto INTEGER }

idSensoreContatto OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID del sensore a contatto"
    ::= { sensoreContattoEntry 1 }

tipoSensoreContatto OBJECT-TYPE
    SYNTAX INTEGER {
        porta(0),
        finestra(1),
        tombino(2),
        lucernario(3) }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "indica su che apertura e applicato il sensore a contatto"
    ::= { sensoreContattoEntry 2 }

statoSensoreContatto OBJECT-TYPE
    SYNTAX INTEGER {
        spento(0),
        acceso(1),
        allarme(2) }
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION

```

```

        "Stato del sensore a contatto"
 ::= { sensoreContattoEntry 3 }

```

```

sensoriIntrusiTable OBJECT-TYPE
    SYNTAX SEQUENCE OF SensoreIntrusiEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Tabella riassuntiva dei sensori per individuare intrusi presenti
        nell'ambiente"
 ::= { sistemaSicurezza 2 }

```

```

sensoreIntrusiEntry OBJECT-TYPE
    SYNTAX SensoreIntrusiEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella sensore intrusi"
    INDEX {
        idSensoreIntrusi }
 ::= { sensoriIntrusiTable 1 }

```

```

SensoreIntrusiEntry ::= SEQUENCE {

    idSensoreIntrusi          Unsigned32,
    sensibilitaSensoreIntrusi Integer32,
    descrizioneSensoreIntusi  DisplayString,
    statoSensoreIntrusi      INTEGER }

```

```

idSensoreIntrusi OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID del sensore anti-intrusion"
 ::= { sensoreIntrusiEntry 1 }

```

```

sensibilitaSensoreIntrusi OBJECT-TYPE
    SYNTAX Integer32 (0..10)
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Precisione regolabile, da 0 a 10, del sensore anti-intrusione"
 ::= { sensoreIntrusiEntry 2 }

```

```

descrizioneSensoreIntusi OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Breve descrizione del tipo di sensore"
 ::= { sensoreIntrusiEntry 3 }

```

```

statoSensoreIntrusi OBJECT-TYPE
    SYNTAX INTEGER {
        spento(0),
        acceso(1),
        allarme(2) }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Stato del sensore anti-intrusione"
 ::= { sensoreIntrusiEntry 4 }

```

```

dispositivoAccessoTable OBJECT-TYPE
    SYNTAX SEQUENCE OF DispositivoAccessoEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Tabella riassuntiva dei dispositivi d'accesso consentito"
    ::= { sistemaSicurezza 3 }

dispositivoAccessoEntry OBJECT-TYPE
    SYNTAX DispositivoAccessoEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella dei dispositivi d'accesso"
    INDEX {
        idDispositivoAccesso }
    ::= { dispositivoAccessoTable 1 }

DispositivoAccessoEntry ::= SEQUENCE {

    idDispositivoAccesso      Unsigned32,
    infoDispositivoAccesso   DisplayString,
    statoDispositivoAccesso  INTEGER,
    tentativiMaxAccesso     Unsigned32 }

idDispositivoAccesso OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID univoco del dispositivo d'accesso"
    ::= { dispositivoAccessoEntry 1 }

infoDispositivoAccesso OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Caratteristiche del dispositivo d'accesso es: badge, PIN,
        retina, impronte digitali etc"
    ::= { dispositivoAccessoEntry 2 }

statoDispositivoAccesso OBJECT-TYPE
    SYNTAX INTEGER {
        spento(0),
        acceso(1),
        letturaOK(2),
        letturaFallita(3),
        guasto(4) }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Indica lo stato del dispositivo d'accesso"
    ::= { dispositivoAccessoEntry 3 }

tentativiMaxAccesso OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Indica il numero max di tentativi disponibili per ogni tipo di
        dispositivo"

```

```
 ::= { dispositivoAccessoEntry 4 }
```

```
controlloAlimentazione OBJECT IDENTIFIER ::= { environmentalControlMIB 4 }
```

```
generatoriCorrenteTable OBJECT-TYPE
    SYNTAX SEQUENCE OF GeneratoreCorrenteEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "tabella dei generatori di corrente che garantiscono
         alimentazione a tutto il sistema ed eventualmente
         all'oggetto all'interno dell'ambiente nel caso questo abbia
         bisogno di corrente"
    ::= { controlloAlimentazione 1 }
```

```
generatoreCorrenteEntry OBJECT-TYPE
    SYNTAX GeneratoreCorrenteEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Elemento della tabella dei generatori di corrente"
    INDEX {
        idGeneratoreCorrente }
    ::= { generatoriCorrenteTable 1 }
```

```
GeneratoreCorrenteEntry ::= SEQUENCE {

    idGeneratoreCorrente      Unsigned32,
    statoGeneratoreCorrente   INTEGER,
    collocazioneGeneratoreCorrente DisplayString,
    caricaGeneratoreCorrente  Integer32,
    sogliaCaricaMinima        Unsigned32 }
```

```
idGeneratoreCorrente OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "ID univoco del generatore di corrente"
    ::= { generatoreCorrenteEntry 1 }
```

```
statoGeneratoreCorrente OBJECT-TYPE
    SYNTAX INTEGER {
        spento(0),
        acceso(1),
        guasto(2) }
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Stato del generatore di corrente"
    ::= { generatoreCorrenteEntry 2 }
```

```
collocazioneGeneratoreCorrente OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Indica la collocazione del generatore per un rapido intervento
         in caso di guasto"
    ::= { generatoreCorrenteEntry 3 }
```

```
caricaGeneratoreCorrente OBJECT-TYPE
    SYNTAX Integer32 (0..100)
    MAX-ACCESS read-only
```

STATUS current
DESCRIPTION
"Livello di carica del generatore"
::= { [generatoreCorrenteEntry](#) 4 }

sogliaCaricaMinima **OBJECT-TYPE**
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Soglia di carica oltre la quale viene lanciato un allarme"
::= { [generatoreCorrenteEntry](#) 5 }

indicatoreConsumoCorrente **OBJECT-TYPE**
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Misuratore del consumo di corrente "
::= { [controlloAlimentazione](#) 2 }

livelloMaxConsumoCorrente **OBJECT-TYPE**
SYNTAX Unsigned32
MAX-ACCESS read-write
STATUS current
DESCRIPTION
*"Indica il livello max consentito di consumo di corrente, oltre
il quale scatta un allarme"*
::= { [controlloAlimentazione](#) 3 }

allarmi **OBJECT IDENTIFIER** ::= { [environmentalControlMIB](#) 5 }

allarmePresenzaFumo **NOTIFICATION-TYPE**
OBJECTS {
[idSensoreFumo](#),
[statoSensoreFumo](#),
[sensibilitaSensoreFumo](#)}
STATUS current
DESCRIPTION
*"Allarme che indica la presenza di fumo nell'ambiente, indicando
quale sensore ha dato l'allarme"*
::= { [allarmi](#) 1 }

allarmaPresenzaAcqua **NOTIFICATION-TYPE**
OBJECTS {
[idSensoreAcqua](#),
[statoSensoreAcqua](#)}
STATUS current
DESCRIPTION
*"Allarme che indica la presenza di acqua nell'ambiente, indicando
il sensore che ha dato l'allarme"*
::= { [allarmi](#) 2 }

allarmeFiltroClimatizzatore **NOTIFICATION-TYPE**
OBJECTS {
[idClimatizzatore](#),
[puliziaFiltroClimatizzatore](#)}
STATUS current
DESCRIPTION
"Allarme che scatta quando il filtro del climatizzatore e sporco"
::= { [allarmi](#) 3 }

allarmeIntrusione **NOTIFICATION-TYPE**

```

OBJECTS {
    idSensoreContatto,
    tipoSensoreContatto,
    statoSensoreContatto}
STATUS current
DESCRIPTION
    "Allarme che scatta quando uno dei sensori a contatto rileva un
    accesso da vie non autorizzate come finestre e lucernari"
 ::= { allarmi 4 }

```

```

allarmePresenzaIntrusi NOTIFICATION-TYPE
OBJECTS {
    idSensoreIntrusi,
    descrizioneSensoreIntusi,
    statoSensoreIntrusi}
STATUS current
DESCRIPTION
    "Allarme che scatta quando viene rilevato un intruso all'interno
    dell'ambiente
    attraverso ad esempio Motion Detector, sensori volumetrici "
 ::= { allarmi 5 }

```

```

allarmeAccessoNonAutorizzato NOTIFICATION-TYPE
OBJECTS {
    idDispositivoAccesso,
    infoDispositivoAccesso,
    statoDispositivoAccesso}
STATUS current
DESCRIPTION
    "Allarme che scatta quando fallisce il tentativo di accesso
    attraverso vie autorizzate, es. ingresso principale o secondari."
 ::= { allarmi 6 }

```

```

allarmeAccessoAutorizzato NOTIFICATION-TYPE
OBJECTS {
    idDispositivoAccesso,
    infoDispositivoAccesso,
    statoDispositivoAccesso}
STATUS current
DESCRIPTION
    "Allarme che notifica l'avvenuto accesso con successo attraverso
    le vie consentite"
 ::= { allarmi 7 }

```

```

guastoFonteLuminosa NOTIFICATION-TYPE
OBJECTS {
    idFonte,
    statoFonte,
    classificazioneFonte,
    collocazioneFonteLuminosa}
STATUS current
DESCRIPTION
    "allarme che notifica il guasto di una fonte luminosa"
 ::= { allarmi 8 }

```

```

guastoClimatizzatore NOTIFICATION-TYPE
OBJECTS {
    idClimatizzatore,
    statoClimatizzatore,
    collocazioneClimatizzatore}
STATUS current
DESCRIPTION
    "allarme che notifica il guasto di uno dei climatizzatori"
 ::= { allarmi 9 }

```

```

guastoDispositivoAccesso NOTIFICATION-TYPE
    OBJECTS {
        idDispositivoAccesso,
        infoDispositivoAccesso,
        statoDispositivoAccesso}
    STATUS current
    DESCRIPTION
        "allarme che notifica il guasto di uno dei dispositivi di accesso
        autorizzato"
    ::= { allarmi 10 }

guastoGeneratoreCorrente NOTIFICATION-TYPE
    OBJECTS {
        idGeneratoreCorrente,
        statoGeneratoreCorrente,
        collocazioneGeneratoreCorrente}
    STATUS current
    DESCRIPTION
        "allarme che notifica il guasto di uno dei generatori di
        corrente"
    ::= { allarmi 11 }

supSogliaIlluminazioneMax NOTIFICATION-TYPE
    OBJECTS {
        idSensoreLuce,
        collocazioneSensoreLuce,
        sogliaIlluminazioneMax}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        l'illuminazione max"
    ::= { allarmi 12 }

supSogliaTemperaturaMax NOTIFICATION-TYPE
    OBJECTS {
        sensoreTemperatura,
        sogliaTemperaturaMax}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        la temperatura max"
    ::= { allarmi 13 }

supSogliaTemperaturaMin NOTIFICATION-TYPE
    OBJECTS {
        sensoreTemperatura,
        sogliaTemperaturaMin}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        la temperatura min"
    ::= { allarmi 14 }

supSogliaUmiditaMax NOTIFICATION-TYPE
    OBJECTS {
        sensoreUmidita,
        sogliaUmiditaMax}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        l'umidita max"
    ::= { allarmi 15 }

```

```

supSogliaUmiditaMin NOTIFICATION-TYPE
    OBJECTS {
        sensoreUmidita,
        sogliaUmiditaMin}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        l'umidita min"
    ::= { allarmi 16 }

supSogliaTentativiAccesso NOTIFICATION-TYPE
    OBJECTS {
        idDispositivoAccesso,
        infoDispositivoAccesso,
        tentativiMaxAccesso}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        il numero dei tentativi d'accesso"
    ::= { allarmi 17 }

supSogliaCaricaGeneratore NOTIFICATION-TYPE
    OBJECTS {
        idGeneratoreCorrente,
        collocazioneGeneratoreCorrente,
        caricaGeneratoreCorrente,
        sogliaCaricaMinima}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        il livello di carica min del generatore"
    ::= { allarmi 18 }

supSogliaConsumoMaxCorrente NOTIFICATION-TYPE
    OBJECTS {
        indicatoreConsumoCorrente,
        livelloMaxConsumoCorrente}
    STATUS current
    DESCRIPTION
        "Allarme che notifica il superamento della soglia impostata per
        il consumo max di corrente"
    ::= { allarmi 19 }

envControlConformita OBJECT IDENTIFIER ::= { environmentalControlMIB 6 }

envControlGroups OBJECT IDENTIFIER ::= { envControlConformita 1 }

envControlCompliances OBJECT IDENTIFIER ::= { envControlConformita 2 }

envControlMandatory OBJECT-GROUP
    OBJECTS {
        idSistema, responsabile, contenutoAmbiente, superficieAmbiente,
        volumeAmbiente, dataInstallazione, statoSensoreFumo,
        statoFonte, classificazioneFonte, collocazioneFonteLuminosa,
        sensoreTemperatura, sogliaTemperaturaMax, sogliaTemperaturaMin,
        statoSensoreAcqua, temperaturaClimatizzatore,
        statoClimatizzatore, puliziaFiltroClimatizzatore,
        collocazioneClimatizzatore, sensoreUmidita, sogliaUmiditaMax,
        sogliaUmiditaMin, tipoSensoreContatto, statoSensoreContatto,
        descrizioneSensoreIntusi, statoSensoreIntrusi,
        statoDispositivoAccesso, statoGeneratoreCorrente,
        collocazioneGeneratoreCorrente, caricaGeneratoreCorrente,
        sogliaCaricaMinima, idSensoreFumo, idFonte, idSensoreAcqua,
        idClimatizzatore, idSensoreContatto, idSensoreIntrusi,
        idDispositivoAccesso, idGeneratoreCorrente }
    STATUS current
    DESCRIPTION

```

```

        "oggetti del MIB che devono necessariamente essere implementati"
 ::= { envControlGroups 1 }

envControlOptional OBJECT-GROUP
  OBJECTS {
    responsabile,noteAggiuntive,sensibilitaSensoreFumo,
    intensitaFonte,risultatoSensoreLuce,collocazioneSensoreLuce,
    sensibilitaSensoreLuce,sogliaIlluminazioneMax,
    sensibilitaSensoreIntrusi,infoDispositivoAccesso,
    tentativiMaxAccesso,indicatoreConsumoCorrente,
    livelloMaxConsumoCorrente,indirizzo,idSensoreLuce }
  STATUS current
  DESCRIPTION
    "oggetti opzionali per la implementazione da parte del agent"
 ::= { envControlGroups 3 }

envControlNotifyMandatory NOTIFICATION-GROUP
  NOTIFICATIONS {
    allarmePresenzaFumo,allarmaPresenzaAcqua,
    allarmeFiltroClimatizzatore,allarmeIntrusione,
    allarmePresenzaIntrusi,allarmeAccessoNonAutorizzato,
    allarmeAccessoAutorizzato,guastoFonteLuminosa,
    guastoClimatizzatore,guastoDispositivoAccesso,
    guastoGeneratoreCorrente,supSogliaTemperaturaMax,
    supSogliaTemperaturaMin,supSogliaUmiditaMax,
    supSogliaUmiditaMin,supSogliaCaricaGeneratore }
  STATUS current
  DESCRIPTION
    "allarmi che devono necessariamente essere implementati"
 ::= { envControlGroups 2 }

envControlNotificationOptional NOTIFICATION-GROUP
  NOTIFICATIONS {
    supSogliaIlluminazioneMax,supSogliaTentativiAccesso,
    supSogliaConsumoMaxCorrente }
  STATUS current
  DESCRIPTION
    "allarmi opzionali per implementazione da parte del agent"
 ::= { envControlGroups 4 }

envControlCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION
    "dichiarazione di conformita per il modulo Environmental Control
    MIB"

  MODULE ENVIRONMENTAL-CONTROL-MIB
  MANDATORY-GROUPS {
    envControlMandatory,envControlNotifyMandatory }

 ::= { envControlCompliances 1 }

END

```

4.1 Note

- ° Environmental Control MIB è stato validato senza errori né warnings dal validatore di www.simpleweb.org fino a livello 4 compreso.

5. Conclusioni

Attraverso un rapido *excursus* sui prodotti creati da alcune aziende che si occupano di implementare sistemi di Environmental Control, abbiamo creato un MIB che si avvicinasse il più possibile ad una soluzione realistica per questo genere di problematiche.

Ovviamente per questioni di tempo non abbiamo potuto approfondire in maniera troppo particolareggiata alcuni aspetti: ad esempio il sistema di autenticazione, non essendo possibile entrare nel dettaglio di ogni singolo sistema; anche la trattazione degli allarmi (trap) è stata limitata a quelli più importanti.

È prevedibile e auspicabile che in un futuro prossimo questo tipo di sistema possa diventare più accessibile di conseguenza più usufruibile anche per l'utente medio, per favorire il raggiungimento di un maggiore livello di comfort, sicurezza e risparmio energetico per tutta la società.

6. Bibliografia

Testi di Riferimento:

- J. Schonwalder, L. Deri - "Sistemi di Elaborazione dell'Informazione: Gestione di Rete"
- James F. Kurose, Keith W. Ross - "Computer Networking", Addison Wesley
- Informazioni su SNMPv.2 SMI DataType - <http://www.adventnet.com>
- Informazioni su ASN.1 - <http://asn1.elibel.tm.fr>
- RFC – <http://www.faqs.org/rfcs>
- RFC 1212
- RFC 1215

Modelli di Environmental Control di riferimento ed altri link utili:

- MicroSystems – <http://www.microsystem.it>
- AKCP – <http://www.akcp.ca> (anche per la consultazione dei sensori e dei relativi prezzi)
- OPTO22 – <http://www.ManageTheRealWorld.com>
- www.simpleweb.org
- <http://www.atis.org/tg2k/t1g2k.html> american national standard