

definizione di un MIB SNMP per il monitoraggio di un'ambulanza

Massimo Bruschi

15 febbraio 2007



1 Introduzione

Il progetto si pone l'obiettivo ambizioso di fornire alla centrale operativa di un pronto soccorso uno strumento informatico che le consenta di conoscere in ogni istante lo stato delle ambulanze che gestisce. Questo sistema si rivelerebbe utile durante gli interventi di routine ma, in particolare, in quei casi in cui il mezzo di soccorso sia costretto ad effettuare una lunga missione in zone periferiche o sia impossibilitato al rientro (per esempio a causa di calamità naturali o maltempo). Segue una sezione che può essere utile a comprendere il dominio del problema.

1.1 Le unità mobili di soccorso

Un'ambulanza è un veicolo, permanentemente adibito al trasporto di feriti o malati. In Italia la costruzione delle ambulanze¹ è regolamentata dal Decreto del Ministero dei Trasporti e della Navigazione². Il decreto individua due tipologie di ambulanza:

- tipo A: con carrozzeria definita autoambulanza di soccorso, attrezzate per il trasporto di infermi o infortunati e per il servizio di pronto soccorso, dotate di specifiche attrezzature di assistenza;
- tipo B: con carrozzeria definita autoambulanza di trasporto, attrezzate essenzialmente per il trasporto di infermi o infortunati, con eventuale dotazione di semplici attrezzature di assistenza.



Figura 1: Autoambulanza di soccorso

¹formalmente MSB, o Mezzi di Soccorsi di Base, per distinguerle dalle Automediche, o MSA

²n. 553 del 17 dicembre 1987

La normativa sulle caratteristiche delle ambulanze comprende anche un successivo decreto³ che aggiorna e integra il precedente, definendo le ambulanze di soccorso per emergenze speciali, veicoli adibiti al trasporto, al trattamento di base e al monitoraggio dei pazienti. Tra queste ricadono le autoambulanze di soccorso per emergenze speciali in proprietà o usufrutto di aziende sanitarie locali, ospedali, cliniche, Croce Rossa Italiana ed associazioni di pubblica assistenza o volontaristiche riconosciute⁴ (tratto da (17)).

1.1.1 Equipaggiamento

Tutte le ambulanze devono essere dotate di attrezzature essenziali, riportiamo degli esempi:

- Laccio emostatico
- Barella a cucchiaio
- Dispositivo di erogazione ossigeno⁵
- Kit di medicazione

Ma sono i mezzi di soccorso di tipo A che devono avere anche dispositivi più avanzati come:

- Kit Rianimazione⁶
- Ricetrasmittente⁷
- Telefono cellulare
- Monitor elettrocardiogramma (ECG) e defibrillatore⁸.
- Elettrocardiografo⁹
- Plussosimetro^{10 11}

³20 novembre 1997 n. 487

⁴ad esempio: croce Bianca Milano, ANPAS, Misericordia

⁵Tipo fisso e mobile.

⁶Pediatrico e neonatale

⁷Portatile e veicolare

⁸con registratore su carta, portatile e a batteria

⁹portatile e a batteria

¹⁰Lo strumento permette il monitoraggio non invasivo della saturazione in ossigeno dell'emoglobina (ossigenazione del sangue) e quindi il riconoscimento precoce dell'ipossiemia e la frequenza cardiaca.

¹¹portatile a batteria con sonde per adulti e bambini



Figura 2: Interno di una autoambulanza di tipo B

- Estricatore di sicurezza¹²
- Pompe da infusione a siringa

(tratto da (7))

1.2 Problematiche

Durante la breve fase di progettazione il problema è stato quello di dover stabilire un canale trasmissivo tra un mezzo in movimento e la centrale operativa (che possono essere separati da distanze anche chilometriche). Tuttavia la questione è già stata affrontata dai costruttori di antifurti satellitari (6).

1.3 Antifurti satellitari

Un antifurto di questo tipo è composto da: una ricetrasmittente, installata nel veicolo, che invia i dati tramite SMS¹³ sulla la rete GSM¹⁴ e li riceve

¹²Respiuratore automatico

¹³Short Message Service

¹⁴Global System for Mobile Communications



Figura 3: Interno di una autoambulanza di tipo A

utilizzando il GPS¹⁵ e da una componente per l'elaborazione dei dati ricevuti dal veicolo (Figura 4). Allo stato dell'arte esistono funzionalità più avanza-

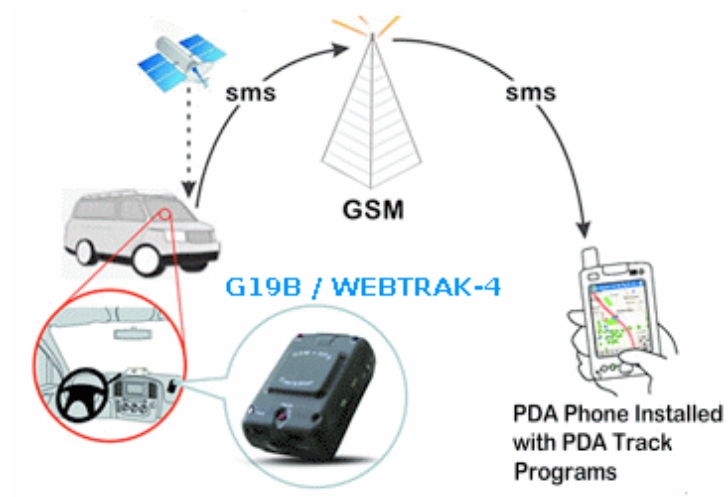


Figura 4: Funzionamento di un antifurto satellitare

te che, per trasmettere i dati, utilizzano il servizio GPRS¹⁶ (20) della rete GSM. Questo servizio consente la comunicazione a modulazione di pacchetto, in particolare possono essere trasmessi pacchetti IP. Questo tipo di comunicazione si differenzia dalla precedente in quanto ora la ricetrasmittente si connette ad un Server che, tramite internet, permette ai clients di aggiornare una mappa dove è indicata la posizione del veicolo (Figura 5).

¹⁵Global Positioning System

¹⁶General Packet Radio Service

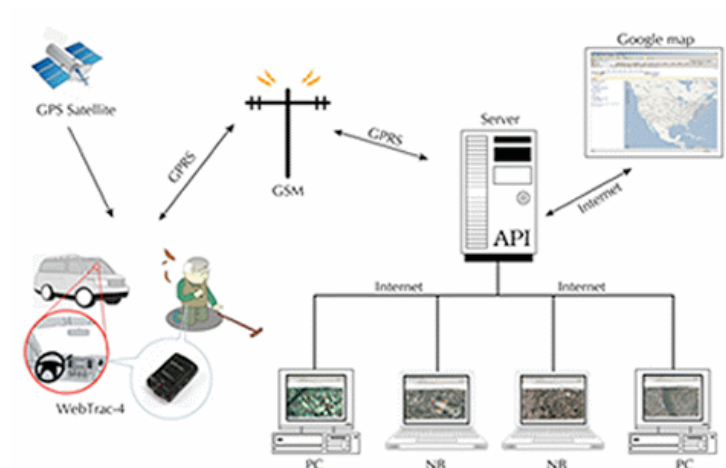


Figura 5: Antifurto satellitare connesso ad Internet

2 Il Progetto

2.1 Scelte progettuali

Utilizzando una tecnologia simile a quella sopra descritta la centrale può conoscere solo la posizione di ogni ambulanza. Il fine del progetto è quello di mettere la centrale operativa in condizione di monitorare continuamente anche i parametri vitali del paziente rilevati a bordo e lo stato dei dispositivi del mezzo di soccorso. Analizziamo in dettaglio questi due aspetti. Il primo è importante per poter predisporre gli interventi sul paziente al suo arrivo in ospedale e la sua priorità rispetto ad altri pazienti¹⁷. Il secondo aspetto è molto rilevante perchè programmare tempestivamente gli interventi di manutenzione o riparazione abbatta i tempi di inattività dell'ambulanza e, quindi, il rischio di intervenire sul campo in modo non ottimale. Per quanto sopra osservato è necessario definire un MIB (9) che supporti tali funzionalità. Il sistema sembra tuttavia sovradimensionato per le ambulanze di tipo B che non dispongono di molte delle attrezzature elettroniche monitorabili, quindi, nel seguito, ci riferiremo sempre a quelle di tipo A.

Comunicazione La comunicazione vocale tra gli operatori sul campo e quelli della centrale è garantita dal fatto che le ambulanze di tipo A sono già dotate di telefono cellulare. Un ulteriore tipo di comunicazione, forzatamente testuale, appare ridondante e poco adatta a momenti di concitazione. Va anche notato che se si verificasse una disfunzione nella rete GSM, per cui il

¹⁷La priorità è classificata con una scala cromatica che va dal bianco al nero

cellulare non fosse utilizzabile, allora neanche il servizio GPRS lo sarebbe. La comunicazione finalizzata al controllo dei dispositivi elettro-meccanici è inutile perché devono essere solo e soltanto i soccorritori a manovrare tali apparecchi senza ingerenze da parte di entità remote¹⁸. Per tutti questi motivi si è scelto di utilizzare un canale di comunicazione unidirezionale.

Hardware Su ogni mezzo di soccorso sarà presente un notebook collegato in rete utilizzando il servizio GPRS. Il portatile dovrà anche essere collegato ad un ricevitore GPS e a tutte le attrezzature di bordo che dovranno mettere a disposizione una interfaccia adeguata (Figura 6). La centrale operativa

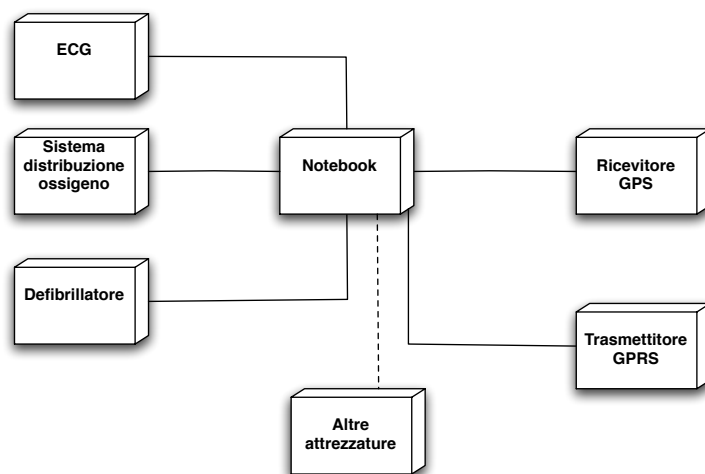


Figura 6: Schema dei dispositivi presenti in una ambulanza di tipo A

dovrà essere dotata di un ricevitore GPRS al quale accede un PC tramite un'interfaccia (Figura 7).

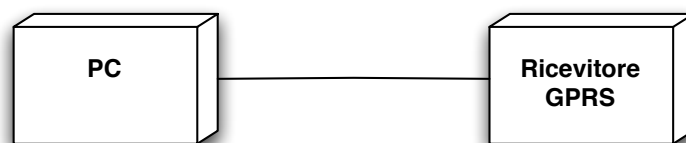


Figura 7: Schema dei dispositivi presenti nella centrale operativa

¹⁸come operatori della centrale operativa o routine automatiche

2.2 MIB

2.2.1 Aspetti generali

Paradigma Manager/Agent A questo punto appare naturale che l'agente SNMP debba essere ospitato nel notebook dell'ambulanza da dove accede agli oggetti da monitorare e invia, quando necessario, i resoconti al Manager. Il Manager, specularmente, si trova nel PC della centrale operativa da dove riceve i messaggi dagli agent e li passa ad un'applicazione che reagisce adeguatamente¹⁹.

Il sistema deve monitorare molti veicoli che necessitano di trasmettere una grossa mole di dati²⁰ contemporaneamente. Perciò si è scelto di consentire agli agents di inviare le traps indipendentemente, valutandone la necessità tramite i valori di soglia ed il tipo di dispositivi attivi in quel momento. Questa scelta è dovuta anche al fatto che, dalla centrale, non è possibile sapere cosa accade nell'ambulanza e, quindi, effettuare ottimamente i sondaggi sugli oggetti. In pratica questo significa che non c'è un'azione di controllo (polling) del Manager ma solo un flusso di informazione che va dagli Agent verso il Manager (dovuto alle traps). In realtà questo modo di procedere contraddice parzialmente la natura del SNMP che attribuisce le responsabilità ed il carico di lavoro maggiori al manager.

Managed Objects Si è adottato un approccio nel quale gli oggetti da astrarre sono tutte le apparecchiature²¹ monitorabili di un mezzo di soccorso di tipo A. Tuttavia appare naturale, vista la loro diversa natura, cercare una differenziazione. Le due tipologie individuate sono distinte in base all'interazione con il paziente:

- Apparecchiature direttamente applicabili al paziente
 - Sistema di distribuzione ossigeno²²
 - Elettrocardiografo
 - Defibrillatore
 - Plussosimetro
 - Impianto di aspirazione endotracheale²³
 - Ventilatore Polmonare

¹⁹in questa fase occorrono notifiche all'operatore della centrale operativa

²⁰ad esempio i valori dell'ECG

²¹alcuni sinonimi usati nel seguito: apparecchi, attrezzature, dispositivi...

²²noto anche come impianto di ossigenoterapia è composto da tubi e bombole

²³composto da un aspiratore e appositi utensili

- Apparecchiature non direttamente applicabili al paziente
 - Centralina Elettronica²⁴
 - Monitor multiparametrico
 - Impianto di climatizzazione
 - Ventilatore/Aspiratore
 - Sterilizzatore ambiente e aria
 - Lampeggianti
 - Luci interne
 - Sirena
 - Frigorifero

(tratto da (1)). In realtà non c'è un equipaggiamento standard ma solo uno minimale, quello preso in esame è uno dei possibili. Per semplicità nel seguito si assumerà che la flotta sia omogenea dal punto di vista delle attrezzature. Con questa premessa possono essere ignorate tutte le informazioni relative ai modelli delle apparecchiature e, perciò, assumere che abbiano tutte lo stesso funzionamento e le stesse funzionalità.

2.2.2 Albero degli Oggetti

Dal ramo dell'albero ISO²⁵ di registrazione che è stato assegnato all'**ambulanceMIB** (nel seguito si suppone che sia il numero 1) partono il ramo degli **apparecchi** e quello delle **traps**²⁶.

(1.1) **apparecchi** Segue una descrizione qualitativa degli oggetti utile a comprendere le strutture dati astraendosi dai dettagli implementativi che saranno affrontati nei paragrafi successivi. Per molti valori non saranno specificate le unità di misura per non rendere macchinosa la trattazione.

In alcuni casi, evidenti dal contesto, sarà tenuta traccia di dati storici²⁷ dei dispositivi in modo da mantenere un quadro complessivo dell'evoluzione dei rilevamenti. La procedura precedentemente illustrata sarà applicata solo alle apparecchiature del primo tipo (vedi 2.2.1) in quanto sono le uniche a

²⁴termine generico utilizzato per individuare l'insieme di dispositivi per l'alimentazione delle apparecchiature a 12 e 220 volts

²⁵Organizzazione Internazionale per le Standardizzazioni (21)

²⁶in realtà partono anche un ramo per i gruppi di oggetti ed un ramo per i gruppi di notifiche ma sono poco rilevanti semanticamente

²⁷per farlo si può utilizzare, quando disponibile, anche la memoria dei dispositivi

rilevare dati utili. L'obiettivo è, cinicamente, quello di ottenere una scatola nera del paziente.

Anche dove non sarà esplicitato è ovvio che tutti i dispositivi devono disporre di un'interfaccia per essere connessi ad un PC²⁸, che, in questo caso, è il notebook.

Sarà introdotto un attributo che indica se un dispositivo è acceso/spento quando tale informazione non è direttamente deducibile da altri attributi.

Notazione: La notazione utilizzata per la descrizione degli oggetti sarà la seguente:

X.Y.Z.W²⁹ **identificatoreUnivoco**³⁰ *nome esteso* breve descrizione e/o note TIPO[N]³¹

Tipi: in questa sezione si tratteranno tipi generici (non necessariamente corrispondenti a quelli in SMIV2) come: booleano, intero, reale, stringa, carattere, maschera di bit, strutturato o un tipo precedentemente descritto.

1. **ambId** *identificatore ambulanza* univoco INTERO
2. **ambStat** può valere -1,0, 1 o 2³² INTERO
3. **ambLat** *latitudine* distanza dall'equatore REALE
4. **ambLon** *longitudine* distanza dal meridiano di Greenwich REALE
5. **appDir** *apparecchiatura direttamente applicabile al paziente*
 - 5.1 **sisOxi** *sistema di distribuzione ossigeno* composto da bombole³³, tubazioni, pannello con prese e riduttore di pressione³⁴ (13).
 - 5.1.1 **sisOxiMan** *manometro* valori da 0 a 500 INTERO
 - 5.1.2 **sisOxiLit** *litri* sono i litri rimanenti. lit = volume x pressione³⁵ (7). REALE

²⁸in realtà molti ne dispongono e sono addirittura forniti con un software ad hoc (vedi schede tecniche in bibliografia)

²⁹gli interi X,Y,Z, W identificano l'oggetto tramite la sua posizione gerarchica nell'albero

³⁰si utilizza la notazione nota come lowerCaseCamel

³¹N indica le dimensioni per i tipi non atomici

³²rispettivamente: in riparazione, non disponibile, disponibile, in servizio

³³per l'omogeneità delle apparecchiature tutte le bombole hanno lo stesso volume

³⁴si suppone che sia dotato di manometro

³⁵attributo ridondante, inserito in questo ambito per completezza

- 5.2 *elCar elettrocardiografo* Portatile, dotato di uscita monitor esterno. Dispone di un monitor proprio, può memorizzare le tracce lette e individuare la presenza di un peacemaker (11).
- 5.2.1 *elCarLivBat livello batteria* valori da 1 a 10 INTERO
- 5.2.2 *elCarAlimOk stato alimentazione* BOOLEANO
- 5.2.3 *elCarProtDefOk stato protezione da defibrillatore* BOOLEANO
- 5.2.4 *elCarOk stato generale* vale false se rilevato un errore generico
BOOLEANO
- 5.2.5 *elCarStampOk stato stampante* BOOLEANO
- 5.2.6 *elCarMonOK stato monitor* BOOLEANO
- 5.2.7 *elCarDisCar disponibilità carta* BOOLEANO
- 5.2.8 *elCarPazMon numero di pazienti monitorati* valori da 0 a 12
INTERO
- 5.2.9 *elCarNumTrac numero tracce memorizzate* valori da 0 a 24
INTERO
- 5.2.10 *elCarTracs tabella delle tracce* contiene gli attributi di tutte le tracce `ELCARTRAC[24]`
- 5.2.10.1 *elCarTrac attributi di una traccia* STRUTTURATO
- *elCarTabAIndex indice della traccia* individua univocamente la traccia nelle tabelle degli attributi, deve essere coerente con l'indice della tabella delle letture. assume valori da 1 a 24 INTERO
 - *elCarUlt numero dell'ultima lettura* utilizzato per ricostruire la cronologia dei valori rilevati assume valori da 1 a 2147483647 ³⁶ INTERO
 - *elCarInEs in esame* vale true se la traccia è in evoluzione BOOLEANO
 - *elCarPeaMak pacemaker* BOOLEANO
- 5.2.11 *elCarLets tabella delle letture* contiene gli ultimi valori rilevati `ELCARLET[2147483647]`
- 5.2.12 *elCarLet attributi di una lettura* contiene gli attributi di una lettura STRUTTURATO
- *elCarTabLIndex* individua la traccia associata alla lettura, deve essere coerente con l'indice della tabella degli attributi. assume valori da 1 a 24 INTERO
 - *elCarNumSeq numero di sequenza* sono gestiti come vettore circolare INTERO

³⁶Massimo numero intero positivo rappresentabile con 32 Bit

- `e1CarVal` *valore letto* REALE
- 5.3 `defi defibrillatore` portatile, dotato di ECG, monitor e uscita monitor esterno. Dispone di peacemaker esterno e piastre³⁷, può memorizzare i dati delle scariche effettuate (10).
- 5.3.1 `defiLivBat` *livello batteria* valori da 1 a 10 INTERO
 - 5.3.2 `defiAlimOk` *stato alimentazione* BOOLEANO
 - 5.3.3 `defiOk` *stato generale* vale false se rilevato un errore generico BOOLEANO
 - 5.3.4 `defiMonOk` *stato monitor* BOOLEANO
 - 5.3.5 `defiECGOk` *stato elettrocardiografo* BOOLEANO
 - 5.3.6 `defiUlt` *numero dell'ultima scarica* utilizzato per ricostruire la cronologia delle scariche assume valori da 1 a 100 INTERO
 - 5.3.7 `defiScariche` contiene le ultime 1000 scariche SCARICA[1000]
 - 5.3.8 `defiScarica` contiene i dati di ogni scarica STRUTTURATO
 - `defiIndex` *indice* individua univocamente una scarica, assume valori da 1 a 1000 INTERO
 - `defiNRG` *energia* INTERO
 - `defiDisar` *disarmo* vale true se è intervenuto il disarmo automatico³⁸ BOOLEANO
 - `defiInter` *interne* vale true se sono state utilizzate le piastre interne, false altrimenti BOOLEANO
 - `defiIntCar` *intervallo di caricamento* numero di secondi trascorsi dalla scarica precedente INTERO
 - `defiDurSca` *durata della scarica* INTERO
- 5.4 `plus plussosimetro` portatile, dotato di rilevatore battito cardiaco, monitor e uscita monitor esterno. Supporta un solo paziente alla volta (5).
- 5.4.1 `plusLivBat` *livello batteria* valori da 1 a 10 INTERO
 - 5.4.2 `plusAlimOk` *stato alimentazione* BOOLEANO
 - 5.4.3 `plusOk` *stato generale* vale false se rilevato un errore generico BOOLEANO
 - 5.4.4 `plusMonOk` *stato monitor* BOOLEANO
 - 5.4.5 `plusProtDefOk` *stato protezione da defibrillatore* BOOLEANO

³⁷per uso interno ed esterno

³⁸interviene dopo 30 sec

- 5.4.6 **plusUlt** *numero dell'ultimo rilevamento* utilizzato per ricostruire la cronologia dei rilevamenti assume valori da 1 a 1000 INTERO
- 5.4.7 **plusRilevamenti** contiene gli ultimi 1000 rilevamenti RILEVAMENTO[1000]
- 5.4.8 **plusRilevamento** contiene i dati di un rilevamento istantaneo STRUTTURATO
- **plusIndex** *indice* individua univocamente un rilevamento assume valori da 1 a 1000 INTERO
 - **plusOss** *ossimetria percentuale* assume valori da 0 a 99 INTERO
 - **plusBat** *battito cardiaco* assume valori da 20 a 300 INTERO
- 5.5 **aspir** *aspiratore* utilizzato per l'aspirazione endotracheale. portatile, dotato di compressore (3).
- 5.5.1 **aspirLivBat** *livello batteria* valori da 1 a 10 INTERO
- 5.5.2 **aspirAlimOk** *stato alimentazione* BOOLEANO
- 5.5.3 **aspirOk** *stato generale* vale false se rilevato un errore generico BOOLEANO
- 5.5.4 **aspirLivOli** *livello olio* valori da 1 a 10 BOOLEANO
- 5.5.5 **aspirUlt** *numero dell'ultima sessione* utilizzato per ricostruire la cronologia delle sessioni assume valori da 1 a 100 INTERO
- 5.5.6 **aspirSessioni** contiene le ultime 100 sessioni di utilizzo, sono gestite come un vettore circolare SESUTIL[100]
- 5.5.7 **aspirSesUtil** *sessione di utilizzo* contiene i dati relativi ad un utilizzo dell'aspiratore STRUTTURATO
- **aspirIndex** *indice* individua univocamente una sessione assume valori da 1 a 100 INTERO
 - **aspirComp** *compressore* vale true se la sessione è stata effettuata in modalità compressore, false altrimenti BOOLEANO
 - **aspirPress** *pressione* assume valori da 0 a 30 INTERO
 - **aspirTemp** *tempo di utilizzo* secondi di attività INTERO
 - **aspirInUso** vale true se il dispositivo è in funzione, cioè se la sessione è in evoluzione BOOLEANO
- 5.6 **vent** *ventilatore polmonare* dotato di batteria, allarme pressione massima, minima e apnea, regolatore soglia di allarme e due modalità di ventilazione (14).

- 5.6.1 **ventAlimOk** *stato alimentazione* BOOLEANO
- 5.6.2 **ventLivBat** *livello batteria* valori da 1 a 10 INTERO
- 5.6.3 **ventOk** *stato generale* vale false se rilevato un errore generico
BOOLEANO
- 5.6.4 **ventUlt** *numero dell'ultimo stato* utilizzato per ricostruire la
cronologia degli stati assume valori da 1 a 100 INTERO
- 5.6.5 **ventStati** contiene gli ultimi 100 stati, deve essere aggiornato
ad ogni variazione STATO[100]
- 5.6.6 **ventStato** contiene lo stato istantaneo del dispositivo STRUT-
TURATO
 - **ventIndex** *indice* individua univocamente uno stato as-
sume valori da 1 a 100 INTERO
 - **ventVent** *ventilazione* può valere 1³⁹ o 2⁴⁰ INTERO
 - **ventPsMax** *pressione massima* livello di allarme della pres-
sione massima, assume valori da 0 a 5 REALE
 - **ventPsMin** *pressione minima* livello di allarme della pres-
sione minima, assume valori da 0.5 a 3 REALE
 - **ventCOss** *concentrazione ossigeno* valori da 60 a 100 IN-
TERO
 - **ventManV** *manovuotometro*⁴¹ valori da -20 a 60 INTERO
 - **ventAlmMax** *allarme pressione massima* vale true se l'al-
larne suona BOOLEANO
 - **ventAlmMin** *allarme pressione massima* vale true se l'al-
larne suona BOOLEANO
 - **ventAlmApn** *allarme apnea* vale true se l'allarme suona
BOOLEANO

6. **appIndir** *apparecchiatura non direttamente applicabile al paziente*

6.1 **cenElet** *centralina Elettronica* Comandata da un pannello⁴², ali-
menta tutte le apparecchiature di bordo con prese da 12 e 220
Volts (13).

6.1.1 **cenEletCenOk** *stato centralina* componente intermedio tra il
pannello di controllo ed il pannello elettrico di potenza BOO-
LEANO

³⁹controllata

⁴⁰assistita/controllata

⁴¹strumento per la regolazione della pressione

⁴²quadro generale di controllo e di comando

- 6.1.2 `cenEletPanElPOk` *stato pannello elettrico di potenza* installato nel cofano motore e controllato dalla centralina BOOLEANO
- 6.1.3 `cenEletIntGe` *interruttore generale* vale false se è disattivato⁴³ BOOLEANO
- 6.1.4 `cenEletIntLBnD` *interruttore luci bianche diurne* le luci sono interne. vale true se posizionato su on BOOLEANO
- 6.1.5 `cenEletIntLB1N` *interruttore luci blu notturne* le luci sono interne. vale true se posizionato su on BOOLEANO
- 6.1.6 `cenEletRegVA` *regolatore velocità ventilatore/apiratore* assume valori da 0⁴⁴ a 3 INTERO
- 6.1.7 `cenEletIntVA` *interruttore ventilatore/aspiratore* vale true se posizionato su ventilatore, false altrimenti BOOLEANO
- 6.1.8 `cenEletInt12` *interruttore prese 12 volts* vale true se posizionato su on BOOLEANO
- 6.1.9 `cenElet220Ok` *stato prese da 220 volts* BOOLEANO
- 6.1.10 `cenEletOr10K` *stato orologio* BOOLEANO
- 6.1.11 `cenEletTerOK` *stato termometro* BOOLEANO
- 6.1.12 `cenEletTempI` *temperatura interna* INTERO
- 6.1.13 `cenEletTempE` *temperatura esterna* INTERO
- 6.1.14 `cenEletFusOK` *stato fusibili* contiene un bit per ogni fusibile⁴⁵, se un bit vale 0 il fusibile non fuziona altrimenti si.⁴⁶ MASCHERADIBIT[30]
- 6.2 `monMul` *monitor multiparametrico* collegabile ai dispositivi (vedi sopra) per una migliore visualizzazione, dotato di batteria e memoria interna⁴⁷ (12).
 - 6.2.1 `monMulLivBat` *livello batteria* valori da 1 a 10 INTERO
 - 6.2.2 `monMulAlimOk` *stato alimentazione* BOOLEANO
 - 6.2.3 `monMulOk` *stato generale* vale false se rilevato un errore generico BOOLEANO
 - 6.2.4 `monMulStampOk` *stato stampante* BOOLEANO
 - 6.2.5 `monMulMonOK` *stato monitor* BOOLEANO
 - 6.2.6 `monMulDisCar` *disponibilità carta* BOOLEANO

⁴³in questo caso tutti i dispositivi di bordo non dotati di batteria sono inutilizzabili

⁴⁴ventilatore spento

⁴⁵in SNMPv2 si può definire la dimensione delle stringhe in byte, per questo saranno utilizzati 32 bit cioè 2 più del necessario

⁴⁶l'ordine nel vettore è lo stesso in cui i fusibili si trovano nel dispositivo

⁴⁷non verrà utilizzata perchè sostituita da quella dei dispositivi

- 6.3 *impCli* *impianto di climatizzazione* a tetto, completo di diffusore e pannello di controllo da collocare all'interno del veicolo. Può funzionare a 220 o 12 Volts⁴⁸ (2) (19).
 - 6.3.1 *impCliAlimOk* *stato alimentazione* BOOLEANO
 - 6.3.2 *impCliEvapOk* *stato evaporatore* BOOLEANO
 - 6.3.3 *impCliCompOk* *stato compressore* BOOLEANO
 - 6.3.4 *impCliCondOk* *stato condensatore* BOOLEANO
 - 6.3.5 *impCliRisOk* *stato riscaldatore* funziona solo a 220 Volts BOOLEANO
 - 6.3.6 *impCliOk* *stato generale* vale false se rilevato un errore generico BOOLEANO
 - 6.3.7 *impCliPanOk* *stato pannello* BOOLEANO
 - 6.3.8 *impCliOn* BOOLEANO
- 6.4 *ventAsp* *ventilatore / aspiratore* alimentato a 12 o 220 Volts dalla centralina. (15)
 - 6.4.1 *ventAspAlimOk* *stato alimentazione* BOOLEANO
 - 6.4.2 *ventAspVel* *regolatore velocità ventilatore* assume valori da -3 a 3⁴⁹ INTERO
- 6.5 *sterAmb* *sterilizzatore ambientale* alimentato a 220 Volts⁵⁰. (4)
 - 6.5.1 *sterAmbAlimOk* *stato alimentazione* BOOLEANO
 - 6.5.2 *sterAmbTemp* *tempo di attività* in secondi INTERO
 - 6.5.3 *sterAmbOk* *stato generale* vale false se rilevata un'avaria non meglio identificabile BOOLEANO
- 6.6 *lamp* *lampeggianti* controllati da un pannello situato in cabina (8)
 - 6.6.1 *lampAlimOk* *stato alimentazione* BOOLEANO
 - 6.6.2 *lampTemp* *tempo di attività* in secondi INTERO
 - 6.6.3 *lampOk* *stato generale* vale false se rilevata un'avaria non meglio identificabile BOOLEANO
- 6.7 *sir* *sirena* controllata da un pannello situato in cabina (8)
 - 6.7.1 *sirAlimOk* *stato alimentazione* BOOLEANO
 - 6.7.2 *sirTemp* *tempo di attività* in secondi INTERO

⁴⁸dipendentemente se alimentato da rete o dal motore

⁴⁹per valori negativi il dispositivo funziona da aspiratore, per valori positivi da ventilatore e per il valore '0' è spento

⁵⁰utilizza raggi ultravioletti

6.7.3 **sirOk** *stato generale* vale false se rilevata un'avaria non meglio identificabile BOOLEANO

6.8 **frig** *frigorifero* alimentato a 12 o 220 Volts assicura sia il freddo che il caldo (16)

6.8.1 **frigAlimOk** *stato alimentazione* BOOLEANO

6.8.2 **frigMod** *modalità* vale -1, 0 o 1 ⁵¹ INTERO

6.8.3 **frigTerOk** *stato termostato* BOOLEANO

6.8.4 **frigMotOk** *stato motore* BOOLEANO

6.8.5 **frigTemp** *temperatura* valori da -10 a 80 INTERO

(1.2) **Traps** Segue una descrizione ad alto livello delle notifiche che gli agent inviano al manager. La traps sono classificate in base al dispositivo che ne causa l'invio.

Per esigenze di sinteticità, durante la trattazione, frasi del tipo '*trap inviata quando...*' sono intese come implicite.

Quando si citeranno dei valori di soglia essi sono scelti arbitrariamente da chi scrive⁵².

Notazione: La notazione utilizzata per la descrizione delle notifiche sarà simile a quella utilizzata per gli oggetti:

X.Y.Z.W **identificatoreUnivoco** breve descrizione e/o note

1. **varAmb** trap inviata ogni 30 secondi e ad ogni variazione dello stato dell'ambulanza, notifica anche la posizione

2. **trapsAppDir** Stato delle apparecchiature direttamente applicabili al paziente

2.1 **oxi** rimangono meno di 2 litri nell'impianto di distribuzione ossigeno

2.2 **ecg** Elettrocardiografo

2.2.1 **ecgAvar** si verifica un'avaria o una situazione anomala

2.2.2 **ecgPaz** cambia il numero di pazienti monitorati o il numero di tracce memorizzate

2.2.3 **ecgTracAtt** trap inviata per notificare alla centrale una variazione di riga nella tabella degli attributi

⁵¹rispettivamente: freddo, spento o caldo

⁵²una soluzione realistica dovrebbe rendere variabili questi valori e, soprattutto, li dovrebbe stabilire in accordo con esperti del settore

2.2.4 **ecgTracLett** trap inviata per aggiornare la centrale sull'evoluzione di una traccia ⁵³. Si invia una riga della tabella delle letture. Questa notifica deve ricorrere con la massima frequenza possibile ma solo se la traccia risulta in esame. hanno la precedenza le righe più recenti che possono essere identificate dal valore **e1CarUlt** della tabella degli attributi.

2.3 **defiT** defibrillatore

2.3.1 **defiTavar** si verifica un'avaria o una situazione anomala

2.3.2 **defiTTrac** trap inviata per notificare alla centrale eventuali aggiornamenti delle scariche.

2.4 **plusPlussosimetro**

2.4.1 **plusAvar** si verifica un'avaria o una situazione anomala

2.4.2 **defiRil** trap inviata per notificare alla centrale eventuali aggiornamenti dei rilevamenti.

2.5 **aspirT** Aspiratore

2.5.1 **aspirTavar** si verifica un'avaria o una situazione anomala

2.5.2 **aspirTSess** trap inviata per notificare alla centrale eventuali aggiornamenti delle sessioni di utilizzo.

2.6 **ventTVentilatore**

2.6.1 **ventTavar** si verifica un'avaria o una situazione anomala

2.6.2 **ventTStat** trap inviata per notificare alla centrale eventuali aggiornamenti dello stato.

3. **trapsAppIndir** Stato delle apparecchiature non direttamente applicabili al paziente

3.1 **cen** Centralina elettronica⁵⁴

3.1.1 **cenAvar** si verifica un'avaria in uno dei componenti

3.1.2 **cenInt** si agisce su uno degli interruttori

3.1.3 **cenTemp** la temperatura, interna o esterna, sale sopra i 40 gradi o scende sotto i -3 ⁵⁵. In queste situazioni alcuni dispositivi possono avere dei malfunzionamenti.

⁵³ricordiamo che per ogni lettura si registra l'indice della traccia corrispondente nella tabella degli attributi

⁵⁴dalla centralina elettronica si possono avere informazioni relative ad altri dispositivi, ad esempio se le prese a 12 Volts non sono in funzione è ovvio che i dispositivi dotati di batteria stanno attingendo ad essa

⁵⁵questo è l'intervallo di ampiezza massima ricavato dalle schede tecniche dei dispositivi

- 3.2 monAvar si verifica un'avaria o una situazione anomala nel monitor multiparametrico
- 3.3 cli Impianto di climatizzazione
 - 3.3.1 cliOn si accende/spenge il climatizzatore
 - 3.3.2 cliAvar si verifica un'avaria non in uno dei componenti
- 3.4 vA Ventilatore aspiratore
 - 3.4.1 vAAlim si verifica una disfunzione nel sistema di alimentazione
 - 3.4.2 vAVel si verifica una variazione nella velocità
- 3.5 ster Sterilizzatore ambientale
 - 3.5.1 sterAvar si verifica un'avaria
 - 3.5.2 sterTemp trap inviata sia quando il dispositivo viene spento, indica il tempo di utilizzo.
- 3.6 lampT Lampeggianti
 - 3.6.1 lampTAvar si verifica un'avaria
 - 3.6.2 lampTTemp trap inviata sia quando il dispositivo viene spento, indica il tempo di utilizzo.
- 3.7 sirT Sirena
 - 3.7.1 sirTAvar si verifica un'avaria
 - 3.7.2 sirTTemp trap inviata sia quando il dispositivo viene spento, indica il tempo di utilizzo.
- 3.8 frigT Frigorifero
 - 3.8.1 frigTAvar si verifica un malfunzionamento nel motore
 - 3.8.2 frigTStat si verifica una variazione nella modalità di utilizzo

2.2.3 Definizione MIB

```

AMBULANCE-MIB DEFINITIONS ::= BEGIN

IMPORTS MODULE-IDENTITY, OBJECT-TYPE,
        NOTIFICATION-TYPE, OBJECT-IDENTITY,
        Unsigned32, Integer32 FROM SNMPv2-SMI
TEXTUAL-CONVENTION, TruthValue FROM SNMPv2-TC
        MODULE-COMPLIANCE, OBJECT-GROUP,
        NOTIFICATION-GROUP          FROM SNMPv2-CONF;

ambulanceMib MODULE-IDENTITY

```

```
LAST-UPDATED "200701241133Z"
ORGANIZATION "Massimo Bruschi"
CONTACT-INFO "Massimo Bruschi
  Universita' degli studi di Pisa
  Pisa, Italia
  e-mail: bruschi@cli.di.unipi.it"
DESCRIPTION "Modulo MIB per il monitoraggio un'ambulanza"
 ::= { 1 1 }
```

```
-- convenzioni
```

```
LivelloBatteria ::= TEXTUAL-CONVENTION
  STATUS      current
  DESCRIPTION
    "Il valore di questo oggetto identifica il livello
    di carica di una batteria"
  SYNTAX      INTEGER (0..10)
```

```
RealType ::= TEXTUAL-CONVENTION
DISPLAY-HINT "4d^4d"
STATUS current
DESCRIPTION
  "Rappresenta un numero reale positivo normalizzato.
  Le cifre che precedono il simbolo '^' sono le
  cifre della rappresentazione, quelle che lo seguono
  rappresentano l'esponente in base 10.
  quindi data la rappresentazione a^b
  il numero da rappresentare e'  $x = 0,a * 10^b$ "
SYNTAX OCTET STRING (SIZE(8))
```

```
-- principali gruppi di oggetti
```

```
apparecchi OBJECT IDENTIFIER ::= { ambulanceMib 1 }
```

```
traps OBJECT IDENTIFIER ::= { ambulanceMib 2 }
```

```
appDir OBJECT IDENTIFIER ::= { apparecchi 5 }
```

```
appInir OBJECT IDENTIFIER ::= { apparecchi 6 }
```

```
trapsAppDir OBJECT IDENTIFIER ::= { traps 1 }
```

```

trapsAppIndir OBJECT IDENTIFIER ::= { traps 2 }

tipi OBJECT IDENTIFIER ::= { ambulanceMib 3 }

-- oggetti del ramo apparecchi

ambId OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Identificatore univoco dell'ambulanza"
::= { apparecchi 1 }

ambStat OBJECT-TYPE
SYNTAX INTEGER { inRiparazione(-1), nonDisponibile(0),
disponibile(1), inServizio(2) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Riassume lo stato dell'ambulanza"
::= { apparecchi 2 }

ambLat OBJECT-TYPE
SYNTAX RealType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Latitudine"
::= { apparecchi 3 }

ambLon OBJECT-TYPE
SYNTAX RealType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Longitudine"
::= { apparecchi 4 }

-- oggetti del ramo appDir

```

```

-- sistema di distribuzione ossigeno
sisOxi OBJECT IDENTIFIER ::= { appDir 1 }

sisOxiMan OBJECT-TYPE
SYNTAX INTEGER (0..500)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"valore indicato dal manometro"
::= { sisOxi 1 }

sisOxiLit OBJECT-TYPE
SYNTAX RealType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"litri di ossigeno rimanenti"
::= { sisOxi 2 }

-- elettrocardiografo
elCar OBJECT IDENTIFIER ::= { appDir 2 }

elCarLivBat OBJECT-TYPE
SYNTAX LivelloBatteria
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indica il livello di carica della batteria"
::= { elCar 1 }

elCarAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
::= { elCar 2 }

elCarProtDefOk OBJECT-TYPE

```

SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nella protezione da defibrillatore"
::= { elCar 3 }

elCarOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
::= { elCar 4 }

elCarStampOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nella stampante"
::= { elCar 5 }

elCarMonOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel monitor"
::= { elCar 6 }

elCarDisCar OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se la scorta di carta non e'

```

essaurita"
::= { elCar 7 }

elCarPazMon OBJECT-TYPE
SYNTAX INTEGER (0..12)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"numero di pazienti attualmente monitorati"
::= { elCar 8 }

elCarNumTrac OBJECT-TYPE
SYNTAX INTEGER (0..24)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"numero di tracce in memoria"
::= { elCar 9 }

-- tabella degli attributi
elCarTracs OBJECT-TYPE
SYNTAX SEQUENCE OF ElCarTrac
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"tabella degli attributi delle tracce"
::= { elCar 10 }

-- entry nella tabella degli attributi
elCarTrac OBJECT-TYPE
SYNTAX ElCarTrac
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Entry nella tabella degli attributi delle tracce"
INDEX { elCarTabAIndex }
::= { elCarTracs 1 }

ElCarTrac ::= SEQUENCE {
elCarTabAIndex INTEGER,
elCarUlt INTEGER,

```



```
elCarInEs TruthValue,  
elCarPeaMak TruthValue  
}
```

```
elCarTabAIndex OBJECT-TYPE  
SYNTAX INTEGER (1..24)  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"indicizza la tabella degli attributi, deve essere  
coerente con l'indice della tabella delle letture"  
 ::= { elCarTrac 1 }
```

```
elCarUlt OBJECT-TYPE  
SYNTAX INTEGER (1..2147483647)  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"contiene il numero di sequenza dell'ultima lettura  
effettuata"  
 ::= { elCarTrac 2 }
```

```
elCarInEs OBJECT-TYPE  
SYNTAX TruthValue  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"vale 'TRUE' se la traccia e' in evoluzione"  
 ::= { elCarTrac 3 }
```

```
elCarPeaMak OBJECT-TYPE  
SYNTAX TruthValue  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"vale 'TRUE' se e' stata rilevata la presenza di un  
peacemaker"  
 ::= { elCarTrac 4 }
```

```
-- tabella delle letture  
elCarLets OBJECT-TYPE
```

```

SYNTAX SEQUENCE OF ElCarLet
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"tabella delle letture delle tracce"
 ::= { elCar 11 }

-- entry nella tabella delle letture
elCarLet OBJECT-TYPE
SYNTAX ElCarLet
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Entry nella tabella delle letture delle tracce"
INDEX { elCarTabLIndex, elCarNumSeq }
 ::= { elCarLets 1 }

ElCarLet ::= SEQUENCE {
elCarTabLIndex INTEGER,
elCarNumSeq INTEGER,
elCarVal RealType
}

elCarTabLIndex OBJECT-TYPE
SYNTAX INTEGER (1..24)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indicizza la tabella delle letture, deve essere
coerente con l'indice della tabella degli attributi"
 ::= { elCarLet 1 }

elCarNumSeq OBJECT-TYPE
SYNTAX INTEGER (1..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"numero di sequenza temporale della lettura"
 ::= { elCarLet 2 }

elCarVal OBJECT-TYPE

```

```

SYNTAX RealType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"valore letto"
 ::= { elCarLet 3 }

-- defibrillatore

defi OBJECT IDENTIFIER ::= { appDir 3 }

defiLivBat OBJECT-TYPE
SYNTAX LivelloBatteria
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indica il livello di carica della batteria"
 ::= { defi 1 }

defiAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
 ::= { defi 2 }

defiOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
 ::= { defi 3 }

defiMonOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current

```

```

DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel monitor"
::= { defi 4 }

defiECGOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'elettrocardiografo installato nel defibrillatore"
::= { defi 5 }

defiUlt OBJECT-TYPE
SYNTAX INTEGER (1..100)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indice dell'ultima scarica effettuata"
::= { defi 6 }

defiScariche OBJECT-TYPE
    SYNTAX SEQUENCE OF DefiScarica
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "tabella contenente i valori delle scariche"
    ::= { defi 7 }

defiScarica OBJECT-TYPE
    SYNTAX DefiScarica
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Entry nella tabella delle scariche."
    INDEX { defiIndex }
    ::= { defiScariche 1 }

DefiScarica ::= SEQUENCE {
defiIndex INTEGER,

```

```

defiNRG Unsigned32,
defiDisar TruthValue,
defiInter TruthValue,
defiIntCar Unsigned32,
defiDurSca Unsigned32
    }

defiIndex OBJECT-TYPE
    SYNTAX INTEGER (1..1000)
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Individua univocamente una scarica."
    ::= { defiScarica 1 }

defiNRG OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Energia della scarica."
    ::= { defiScarica 2 }

defiDisar OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Vale 'TRUE' se e' intervenuto il disarmo automatico."
    ::= { defiScarica 3 }

defiInter OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Vale 'TRUE' se sono state utilizzate le piastre interne."
    ::= { defiScarica 4 }

defiIntCar OBJECT-TYPE
    SYNTAX Unsigned32

```

```

MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Durata in secondi della ricarica che a preceduto la scarica."
 ::= { defiScarica 5 }

defiDurSca OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Durata in secondi della scarica."
 ::= { defiScarica 6 }

--plussosimetro
plus OBJECT IDENTIFIER ::= { appDir 4 }

plusLivBat OBJECT-TYPE
SYNTAX LivelloBatteria
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indica il livello di carica della batteria"
 ::= { plus 1 }

plusAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
 ::= { plus 2 }

plusOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"

```

```

 ::= { plus 3 }

plusMonOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel monitor"
 ::= { plus 4 }

plusProtDefOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nella protezione da defibrillatore"
 ::= { plus 5 }

plusUlt OBJECT-TYPE
SYNTAX INTEGER (1..1000)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indice dell'ultimo rilevamento effettuato"
 ::= { plus 6 }

--tabella dei rilevamenti
plusRilevamenti OBJECT-TYPE
SYNTAX SEQUENCE OF PlusRilevamento
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"tabella contenente i rilevamenti"
 ::= { plus 7}

--entry nella tabella dei rilevamenti
plusRilevamento OBJECT-TYPE
SYNTAX PlusRilevamento
MAX-ACCESS not-accessible

```

```
STATUS current
DESCRIPTION
"Entry nell tabella contenente i rilevamenti"
INDEX { plusIndex }
::= { plusRilevamenti 1 }
```

```
PlusRilevamento ::= SEQUENCE {
plusIndex INTEGER,
plusOss INTEGER,
plusBatt INTEGER
}
```

```
plusIndex OBJECT-TYPE
SYNTAX INTEGER (1..1000)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indice della tabella dei rilevamenti"
::= { plusRilevamento 1 }
```

```
plusOss OBJECT-TYPE
SYNTAX INTEGER (0..99)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"ossimetria percentuale"
::= { plusRilevamento 2 }
```

```
plusBatt OBJECT-TYPE
SYNTAX INTEGER (20..300)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"battito cardiaco"
::= { plusRilevamento 3 }
```

```
--aspiratore
aspir OBJECT IDENTIFIER ::= { appDir 5 }
```

```
aspirLivBat OBJECT-TYPE
SYNTAX LivelloBatteria
```



```

MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indica il livello di carica della batteria"
::= { aspir 1 }

aspirAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
::= { aspir 2 }

aspirOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
::= { aspir 3 }

aspirLivOli OBJECT-TYPE
SYNTAX LivelloBatteria
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indica il livello attuale dell'olio"
::= { aspir 4 }

aspirUlt OBJECT-TYPE
SYNTAX INTEGER (1..100)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indice dell'ultima sessione di utilizzo"
::= { aspir 5 }

-- tabella delle sessioni

```

aspirSessioni OBJECT-TYPE
SYNTAX SEQUENCE OF AspirSesUtil
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"tabella contenente le sessioni di utilizzo"
::= { aspir 6 }

aspirSesUtil OBJECT-TYPE
SYNTAX AspirSesUtil
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"entry della tabella contenente le sessioni di utilizzo"
INDEX { aspirIndex }
::= { aspirSessioni 1 }

AspirSesUtil ::= SEQUENCE {
aspirIndex INTEGER,
aspirComp TruthValue,
aspirPress INTEGER,
aspirTemp Integer32,
aspirInUso TruthValue
}

aspirIndex OBJECT-TYPE
SYNTAX INTEGER (1..100)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indice della tabella delle sessioni"
::= { aspirSesUtil 1 }

aspirComp OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se la sessione e' stata effettuata in modalita'
compressore, 'FALSE' altrimenti"
::= { aspirSesUtil 2 }

```

aspirPress OBJECT-TYPE
SYNTAX INTEGER (0..30)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"pressione della sessione"
::= { aspirSesUtil 3 }

aspirTemp OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"tempo di utilizzo in secondi"
::= { aspirSesUtil 4 }

aspirInUso OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se la sessione e' in evoluzione"
::= { aspirSesUtil 5 }

--ventilatore
vent OBJECT IDENTIFIER ::= { appDir 6 }

ventLivBat OBJECT-TYPE
SYNTAX LivelloBatteria
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indica il livello di carica della batteria"
::= { vent 1 }

ventAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION

```

```

"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
::= { vent 2 }

ventOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
::= { vent 3 }

ventUlt OBJECT-TYPE
SYNTAX INTEGER (1..100)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indice dell'ultimo stato"
::= { vent 4 }

-- tabella degli stati
ventStati OBJECT-TYPE
SYNTAX SEQUENCE OF VentStato
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"tabella degli stati"
::= { vent 5 }

-- Entry nella tabella degli stati
ventStato OBJECT-TYPE
SYNTAX VentStato
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Entry nella tabella degli stati"
INDEX { ventIndex }
::= { ventStati 1 }

VentStato ::= SEQUENCE {

```

```
ventIndex INTEGER,  
ventVent INTEGER,  
ventPsMax RealType,  
ventPsMin RealType,  
ventCOss INTEGER,  
ventManV INTEGER,  
ventAlmMax TruthValue,  
ventAlmMin TruthValue,  
ventAlmApn TruthValue  
}
```

```
ventIndex OBJECT-TYPE  
SYNTAX INTEGER(1..100)  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"indice della tabella degli stati"  
 ::= { ventStato 1 }
```

```
ventVent OBJECT-TYPE  
SYNTAX INTEGER { controllata(1), assistitaControllata(2) }  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"modalita' di funzionamento"  
 ::= { ventStato 2 }
```

```
ventPsMax OBJECT-TYPE  
SYNTAX RealType  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Livello di allarme della pressione massima.  
Deve assumere valori da 0 a 5"  
 ::= { ventStato 3 }
```

```
ventPsMin OBJECT-TYPE  
SYNTAX RealType  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION
```

"Livello di allarme della pressione minima.
deve assumere valori da 0.5 a 3"
::= { ventStato 4 }

ventCOss OBJECT-TYPE
SYNTAX INTEGER(60..100)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"percentuale di concentrazione ossigeno"
::= { ventStato 5 }

ventManV OBJECT-TYPE
SYNTAX INTEGER(-20..60)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"manovuotometro"
::= { ventStato 6 }

ventAlmMax OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Allarme pressione massima.
vale 'TRUE' solo nel caso in cui l'allarme sia attivo"
::= { ventStato 7 }

ventAlmMin OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Allarme pressione minima.
vale 'TRUE' solo nel caso in cui l'allarme sia attivo"
::= { ventStato 8 }

ventAlmApn OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only

```

STATUS current
DESCRIPTION
"Allarme apnea.
vale 'TRUE' solo nel caso in cui l'allarme sia attivo"
::= { ventStato 9 }

-- oggetti del ramo appInDir
cenElet OBJECT IDENTIFIER ::= { appInir 1 }

cenEletCenOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nella parte elettronica delle centralina"
::= { cenElet 1 }

cenEletPanElPOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel pannello elettrico di potenza"
::= { cenElet 2 }

cenEletIntGe OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se l'interruttore e' su on
'FALSE' altrimenti"
::= { cenElet 3 }

cenEletIntLBnD OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION

```

"vale 'TRUE' se l'interruttore delle luci bianche
diurne e' su on, 'FALSE' altrimenti"
::= { cenElet 4 }

cenEletIntLB1N OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se l'interruttore delle luci blu
notturne e' su on, 'FALSE' altrimenti"
::= { cenElet 5 }

cenEletRegVA OBJECT-TYPE
SYNTAX INTEGER (0..3)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"velocita'. Se vale 0 il ventilatore e'
spento"
::= { cenElet 6 }

cenEletIntVA OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se l'interruttore e' posizionato su
ventilatore, vale 'FALSE' se posizionato su aspiratore"
::= { cenElet 7 }

cenEletInt12 OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se l'interruttore delle prese a
12 Volt e' posizionato su on, 'FALSE' altrimenti"
::= { cenElet 8 }

cenElet2200k OBJECT-TYPE

SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nelle prese da 220 Volt"
::= { cenElet 9 }

cenEletOr10k OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'orologio"
::= { cenElet 10 }

cenEletTerOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel termometro"
::= { cenElet 11 }

cenEletTempI OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"temperatura interna"
::= { cenElet 12 }

cenEletTempE OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"temperatura esterna"
::= { cenElet 13 }

```

-- fusibili
cenEletFusOk OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(4))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Ogni bit rappresenta un fusibile. se un bit e' posto a zero il
fusibile e' danneggiato, altrimenti funziona.
L'ordine dei bit e' lo stesso di quello in cui sono
istallati i fusibili nella centralina"
::= { cenElet 14 }

-- monitor multiparametrico
monMul OBJECT IDENTIFIER ::= { appInir 2 }

monMulLivBat OBJECT-TYPE
SYNTAX LivelloBatteria
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"indica il livello di carica della batteria"
::= { monMul 1 }

monMulAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
::= { monMul 2 }

monMulOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
::= { monMul 4 }

```

```
monMulStampOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nella stampante"
::= { monMul 5 }
```

```
monMulMonOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel monitor"
::= { monMul 6 }
```

```
monMulDisCar OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se la scorta di carta non e'
essaurita"
::= { monMul 7 }
```

```
-- tabella ingressi
```

```
-- impianto di climatizzazione
impCli OBJECT IDENTIFIER ::= { appInir 3 }
```

```
impCliAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
::= { impCli 2 }
```

impCliEvapOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'evaporatore"
::= { impCli 3 }

impCliCompOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel compressore"
::= { impCli 4 }

impCliCondOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel condensatore"
::= { impCli 5 }

impCliRisOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel riscaldatore"
::= { impCli 6 }

impCliOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
::= { impCli 7 }

impCliPanOk OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"vale 'TRUE' se non sono state rilevate avarie
nel pannello di controllo"
::= { impCli 8 }

impCliOn OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"vale 'TRUE' se l'impianto di climatizzazione
e' acceso"
::= { impCli 9 }

ventAsp OBJECT IDENTIFIER ::= { appInir 4 }

ventAspAlimOk OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
::= { ventAsp 1 }

ventAspVel OBJECT-TYPE

SYNTAX INTEGER (-3..3)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"velocita' del ventilatore. per valori negativi il
dispositivo funziona da aspiratore, per valori positivi da

```

ventilatore e per il valore '0' e' spento "
::= { ventAsp 2 }

-- sterilizzatore ambientale
sterAmb OBJECT IDENTIFIER ::= { appInir 5 }

sterAmbAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
::= { sterAmb 1 }

sterAmbTemp OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"tempo di attivita' in secondi"
::= { sterAmb 2 }

sterAmbOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
::= { sterAmb 3 }

-- lampeggianti

lamp OBJECT IDENTIFIER ::= { appInir 6 }

lampAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION

```

```
"vale 'TRUE' se non sono state rilevate avarie  
nell'alimentazione"  
::= { lamp 1 }
```

```
lampTemp OBJECT-TYPE  
SYNTAX Unsigned32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"tempo di attivita' in secondi"  
::= { lamp 2 }
```

```
lampOk OBJECT-TYPE  
SYNTAX TruthValue  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"vale 'TRUE' se non sono state rilevate avarie  
non meglio identificate"  
::= { lamp 3 }
```

```
-- sirena
```

```
sir OBJECT IDENTIFIER ::= { appInir 7 }
```

```
sirAlimOk OBJECT-TYPE  
SYNTAX TruthValue  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"vale 'TRUE' se non sono state rilevate avarie  
nell'alimentazione"  
::= { sir 1 }
```

```
sirTemp OBJECT-TYPE  
SYNTAX Unsigned32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"tempo di attivita' in secondi"  
::= { sir 2 }
```

```

sirOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
non meglio identificate"
 ::= { sir 3 }

-- frigorifero

frig OBJECT IDENTIFIER ::= { appInir 8 }

frigAlimOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nell'alimentazione"
 ::= { frig 1 }

frigMod OBJECT-TYPE
SYNTAX INTEGER { freddo(-1), spento(0), caldo(1) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Riassume lo stato del frigorifero"
 ::= { frig 2 }

frigTerOk OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel termostato"
 ::= { frig 3 }

frigMotOk OBJECT-TYPE

```



```

SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"vale 'TRUE' se non sono state rilevate avarie
nel motore"
::= { frig 4 }

frigTemp OBJECT-TYPE
SYNTAX INTEGER (-10..80)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"temperatura"
::= { frig 5 }

-- traps

-- stato ambulanza
varAmb NOTIFICATION-TYPE
OBJECTS { ambId, ambStat, ambLat, ambLon }
STATUS current
DESCRIPTION
"Notifica inviata ogni 30 secondi e quando
si verifica una variazione nello stato dell'ambulanza"
::= { trapsAppDir 1 }

-- sistema di distribuzione ossigeno
oxi NOTIFICATION-TYPE
OBJECTS { sisOxiMan, sisOxiLit}
STATUS current
DESCRIPTION
"Notifica inviata ogni minuto da quando
la riserva di ossigeno scende sotto i 2 litri"
::= { trapsAppDir 2 }

-- elettrocardiografo
ecg OBJECT IDENTIFIER ::= { trapsAppDir 3 }

ecgAvar NOTIFICATION-TYPE
OBJECTS { elCarLivBat, elCarAlimOk,

```

```

elCarProtDefOk, elCarOk, elCarStampOk,
elCarMonOk, elCarDisCar }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
od una situazione pericolosa"
::= { ecg 1 }

ecgPaz NOTIFICATION-TYPE
OBJECTS { elCarPazMon, elCarNumTrac }
STATUS current
DESCRIPTION
"Notifica inviata quando cambia il numero di pazienti
monitorati od il numero di tracce memorizzate"
::= { ecg 2 }

ecgTracAtt NOTIFICATION-TYPE
OBJECTS { elCarTabAIndex, elCarUlt,
elCarInEs, elCarPeaMak }
STATUS current
DESCRIPTION
"Notifica inviata per aggiornare la centrale sulla variazione di
una riga nella tabella degli attributi.
Per uno stesso indice deve essere inviata con frequenza massima di
1 minuto"
::= { ecg 3 }

ecgTracLett NOTIFICATION-TYPE
OBJECTS { elCarTabLIndex, elCarNumSeq,
elCarVal }
STATUS current
DESCRIPTION
"Notifica inviata per aggiornare la centrale sulla variazione di
dati relativi ad un indice nella tabella delle letture.
La trap viene inviata solo se la traccia di indice corrispondente
nella tabella degli attributi risulta in esame.
Si inviano le notifiche dalla piu' recente (identificabile con l'attributo
elCarUlt) e senza ripetizioni."
::= { ecg 4 }

-- defibrillatore

```

```

defiT OBJECT IDENTIFIER ::= { trapsAppDir 4 }

defiT Avar NOTIFICATION-TYPE
OBJECTS { defiLivBat, defiAlimOk,
defiOk, defiMonOk, defiECGOk }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
od una situazione pericolosa"
::= { defiT 1 }

defiT Trac NOTIFICATION-TYPE
OBJECTS { defiIndex, defiNRG,
defiDisar, defiInter, defiIntCar,
defiDurSca }
STATUS current
DESCRIPTION
"Notifica inviata quando varia il valore di defiUlt, aggiorna
la centrale sul'ultima scarica"
::= { defiT 2 }

-- plussosimetro
plusT OBJECT IDENTIFIER ::= { trapsAppDir 5 }

plusT Avar NOTIFICATION-TYPE
OBJECTS { plusLivBat, plusAlimOk,
plusOk, plusMonOk, plusProtDefOk}
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
od una situazione pericolosa"
::= { plusT 1 }

plusT Ril NOTIFICATION-TYPE
OBJECTS { plusUlt, plusIndex, plusOss,
plusBatt }
STATUS current
DESCRIPTION
"Notifica inviata quando varia il valore di plusUlt, aggiorna
la centrale sul'ultimo rilevamento"
::= { plusT 2 }

```

```

-- aspiratore
aspirT OBJECT IDENTIFIER ::= { trapsAppDir 6 }

aspirTAvar NOTIFICATION-TYPE
OBJECTS { aspirLivBat, aspirAlimOk,
aspirOk, aspirLivOli }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
od una situazione pericolosa"
::= { aspirT 1 }

aspirTRil NOTIFICATION-TYPE
OBJECTS { aspirUlt, aspirIndex, aspirComp,
aspirPress, aspirTemp, aspirInUso }
STATUS current
DESCRIPTION
"Notifica inviata quando varia il valore di aspirUlt, aggiorna
la centrale sull'ultima sessione di utilizzo"
::= { aspirT 2 }

-- ventilatore
ventT OBJECT IDENTIFIER ::= { trapsAppDir 7 }

ventTAvar NOTIFICATION-TYPE
OBJECTS { ventLivBat, ventAlimOk,
ventOk }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
od una situazione pericolosa"
::= { ventT 1 }

ventTSess NOTIFICATION-TYPE
OBJECTS { ventUlt, ventIndex, ventVent,
ventPsMax, ventPsMin, ventCOss,
ventManV, ventAlmMax, ventAlmMin,
ventAlmApn }
STATUS current
DESCRIPTION

```

```

"Notifica inviata quando varia il valore di ventUlt, aggiorna
la centrale sul'ultimo stato"
::= { ventT 2 }

-- centralina elettronica
cen OBJECT IDENTIFIER ::= { trapsAppIndir 1 }

cenAvar NOTIFICATION-TYPE
OBJECTS { cenEletCenOk, cenEletPanElPOk, cenEletOrlOk,
cenEletTerOk, cenEletFusOk, cenElet2200k }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
in uno dei componenti"
::= { cen 1 }

cenInt NOTIFICATION-TYPE
OBJECTS { cenEletIntLBnD, cenEletIntLB1N, cenEletIntGe,
cenEletRegVA, cenEletIntVA, cenEletInt12 }
STATUS current
DESCRIPTION
"Notifica inviata quando si agisce
su uno degli interruttori"
::= { cen 2 }

cenTemp NOTIFICATION-TYPE
OBJECTS { cenEletTempI, cenEletTempE }
STATUS current
DESCRIPTION
"Notifica inviata quando la temperatura interna o
quella esterna sale sopra i 40 gradi o sotto i -3"
::= { cen 3 }

-- monitor multiparametrico
monAvar NOTIFICATION-TYPE
OBJECTS { monMulLivBat, monMulAlimOk, monMulOk,
monMulStampOk, monMulMonOk, monMulDisCar }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
od una situazione pericolosa nel monitor multiparametrico"

```

```

 ::= { trapsAppIndir 2 }

-- impianto di climatizzazione
cli OBJECT IDENTIFIER ::= { trapsAppIndir 3 }

cliOn NOTIFICATION-TYPE
OBJECTS { impCliOn }
STATUS current
DESCRIPTION
"Notifica inviata quando si accende/spegne
l'impianto"
 ::= { cli 1 }

cliAvar NOTIFICATION-TYPE
OBJECTS { impCliAlimOk, impCliEvapOk, impCliCompOk,
        impCliCondOk, impCliRisOk, impCliOk,
        impCliPanOk }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
in uno dei componenti"
 ::= { cli 2 }

-- ventilatore aspiratore
vA OBJECT IDENTIFIER ::= { trapsAppIndir 4 }

vAAlim NOTIFICATION-TYPE
OBJECTS { ventAspAlimOk }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria
nell'alimentazione"
 ::= { vA 1 }

vAVel NOTIFICATION-TYPE
OBJECTS { ventAspVel }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica una variazione di velocita'"
 ::= { vA 2 }

```

```

-- sterilizzatore ambientale
ster OBJECT IDENTIFIER ::= { trapsAppIndir 5 }

sterAvar NOTIFICATION-TYPE
OBJECTS { sterAmbAlimOk, sterAmbOk }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria"
::= { ster 1 }

sterTemp NOTIFICATION-TYPE
OBJECTS { sterAmbTemp }
STATUS current
DESCRIPTION
"Notifica inviata quando si spegne il dispositivo"
::= { ster 2 }

-- lampeggianti
lampT OBJECT IDENTIFIER ::= { trapsAppIndir 6 }

lampTAvar NOTIFICATION-TYPE
OBJECTS { lampAlimOk, lampOk }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria"
::= { lampT 1 }

lampTTemp NOTIFICATION-TYPE
OBJECTS { lampTemp }
STATUS current
DESCRIPTION
"Notifica inviata quando si spegne il dispositivo"
::= { lampT 2 }

-- sirena
sirT OBJECT IDENTIFIER ::= { trapsAppIndir 7 }

sirTAvar NOTIFICATION-TYPE
OBJECTS { sirAlimOk, sirOk }
STATUS current
DESCRIPTION

```

```

"Notifica inviata quando si verifica un'avaria"
::= { sirT 1 }

sirTTemp NOTIFICATION-TYPE
OBJECTS { sirTemp }
STATUS current
DESCRIPTION
"Notifica inviata quando si spegne il dispositivo"
::= { sirT 2 }

--frigorifero
frigT OBJECT IDENTIFIER ::= { trapsAppIndir 8 }

frigTAvar NOTIFICATION-TYPE
OBJECTS { frigAlimOk, frigTerOk,
frigMotOk }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica un'avaria"
::= { frigT 1 }

frigTStat NOTIFICATION-TYPE
OBJECTS { frigMod, frigTemp }
STATUS current
DESCRIPTION
"Notifica inviata quando si verifica una variazione
nella modalita' di utilizzo"
::= { frigT 2 }

-- gruppi
ambAllObj OBJECT-GROUP
    OBJECTS {
        ambId,
ambStat,
ambLat,
ambLon,
sisOxiMan,
sisOxiLit,
elCarLivBat,
elCarAlimOk,
elCarProtDefOk,

```


elCarOk,
elCarStampOk,
elCarMonOk,
elCarDisCar,
elCarPazMon,
elCarNumTrac,
elCarTabAIndex,
elCarUlt,
elCarInEs,
elCarPeaMak,
elCarTabLIndex,
elCarNumSeq,
elCarVal,
defiLivBat,
defiAlimOk,
defiOk,
defiMonOk,
defiECGOk,
defiUlt,
defiIndex,
defiNRG,
defiDisar,
defiInter,
 defiIntCar,
 defiDurSca,
plusLivBat,
plusAlimOk,
plusOk,
plusMonOk,
plusProtDefOk,
plusUlt,
plusIndex,
plusOss,
plusBatt,
aspirLivBat,
aspirAlimOk,
aspirOk,
aspirLivOli,
aspirUlt,
aspirIndex,
aspirComp,

aspirPress,
aspirTemp,
aspirInUso,
ventLivBat,
ventAlimOk,
ventOk,
ventUlt,
ventIndex,
ventVent,
ventPsMax,
ventPsMin,
ventCOss,
ventManV,
ventAlmMax,
ventAlmMin,
ventAlmApn,
cenEletCenOk,
cenEletPanElPOk,
cenEletIntGe,
cenEletIntLBnD,
cenEletIntLB1N,
cenEletRegVA,
cenEletIntVA,
cenEletInt12,
cenElet2200k,
cenEletOr10k,
cenEletTerOk,
cenEletTempI,
cenEletTempE,
cenEletFusOk,
monMulLivBat,
monMulAlimOk,
monMulOk,
monMulStampOk,
monMulMonOk,
monMulDisCar,
impCliAlimOk,
impCliEvapOk,
impCliCompOk,
impCliCondOk,
impCliRisOk,

```

impCliOk,
impCliPanOk,
impCliOn,
ventAspAlimOk,
ventAspVel,
sterAmbAlimOk,
sterAmbTemp,
sterAmbOk,
lampAlimOk,
lampTemp,
lampOk,
sirAlimOk,
sirTemp,
sirOk,
frigAlimOk,
frigMod,
frigTerOk,
frigMotOk,
frigTemp
    }
    STATUS      current
    DESCRIPTION
        "Gruppo contenente tutti gli oggetti"
    ::= { apparecchi 7 }

    ambAllNot NOTIFICATION-GROUP
    NOTIFICATIONS {
varAmb,
oxi,
ecgAvar,
ecgPaz,
ecgTracAtt,
ecgTracLett,
defiTavar,
defiTTrac,
plusTavar,
plusTRil,
aspirTavar,
aspirTRil,
ventTavar,
ventTSess,

```

```

cenAvar,
cenInt,
cenTemp,
monAvar,
cliOn,
cliAvar,
vAAlim,
vAVel,
sterAvar,
sterTemp,
lampTAvar,
lampTTemp,
sirTAvar,
sirTTemp,
frigTAvar,
frigTStat
    }
    STATUS      current
    DESCRIPTION
        "The ATM Forum M4 notification list."
    ::= { apparecchi 8 }

-- AMBULANCE-MIB: Management Information Base that
-- support a remote ambulance monitoring.
-- Copyright (C) 2007 Massimo Bruschi

-- This program is free software; you can redistribute it and/or
-- modify it under the terms of the GNU General Public License
-- as published by the Free Software Foundation; either version 2
-- of the License, or (at your option) any later version.

-- This program is distributed in the hope that it will be useful,
-- but WITHOUT ANY WARRANTY; without even the implied warranty of
-- MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
-- See the GNU General Public License for more details.

-- You should have received a copy of the GNU General Public License
-- along with this program; if not, write to the Free Software
-- Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
-- MA 02110-1301, USA.
END

```

2.2.4 Convalida

La validazione del codice è stata effettuata con successo agli indirizzi:

- <http://www.agentpp.com/mibtools/mibtools.html>
- <http://snmp.cs.utwente.nl/ietf/mibs/validate>

Il primo indirizzo è il sito di riferimento che non produce segnalazioni di errori. L'ultima pagina consente di selezionare 6 livelli di severità della convalida, l'*ambulance-MIB* passa senza segnalazioni fino al livello 5: *'something that might be ok, but might not be recommended under some circumstances'*.

3 Conclusioni e futuro

La soluzione proposta è verosimile ma non ha nessuna pretesa di essere corretta né completa. Chi scrive non conosce le effettive necessità di chi opera nel settore del pronto intervento medico e neppure il funzionamento specifico delle apparecchiature.

Alcune questioni sono state ignorate (vedi 2.2.2). Una di queste riguarda la possibilità di un *'trap storm'* nel caso in cui molti agents trasmettano notifiche contemporaneamente. Una soluzione possibile (ma probabilmente difficile da implementare) al problema è quella di assegnare ad ogni tipo di trap una priorità che deve essere valutata, considerando l'affollamento della rete, al momento della trasmissione. Comunque, per quanto riguarda gli oggetti che rappresentano gli apparecchi di bordo, si è cercato di attenersi ai documenti tecnici citati in bibliografia.

Utile sarebbe estendere il monitoraggio anche alle ambulanze di tipo B che, in caso di necessità, sarebbero chiamate a sostituire quelle di tipo A. Un'altro possibile miglioramento potrebbe consistere nel monitorare anche lo stato elettro-meccanico del veicolo (motore, ABS (18)...) tramite la centralina elettronica della quale sono dotati tutti i veicoli moderni. Tale servizio potrebbe essere utile per intervenire tempestivamente in caso di incidente o guasto meccanico oppure per conoscere l'autonomia del mezzo.

Indice

1	Introduzione	2
1.1	Le unità mobili di soccorso	2
1.1.1	Equipaggiamento	3
1.2	Problematiche	4
1.3	Antifurti satellitari	4
2	Il Progetto	6
2.1	Scelte progettuali	6
2.2	MIB	8
2.2.1	Aspetti generali	8
2.2.2	Albero degli Oggetti	9
2.2.3	Definizione MIB	19
2.2.4	Convalida	61
3	Conclusioni e futuro	61

Riferimenti bibliografici

- [1] “Costruttore ambulanze,” 2006. [Online]. Available: www.bollani.it
- [2] Autoclima, “Condizionatore a tetto dual power,” Autoclima quality and comfort, Tech. Rep., 2006.
- [3] P. . Bird, “Portable aspirator/compressor.”
- [4] C.F.S., “Sterilizzatore ambientale sterilair 3000.” [Online]. Available: <http://www.cfs-shop.it/index.php?productID=1542>
- [5] Criticare, “503dx minispo2t,” Criticare systems, Tech. Rep., 2006.
- [6] Elettronicanet, “Antifurti satellitari.” [Online]. Available: http://www.elettronicanet.com/eshop/Localizzatori_satellitari_gen.htm
- [7] G. Formazione, “Le attrezzature,” Compagnia P.A. Humanitas, Scandicci (FI), Tech. Rep.
- [8] Inav, “Elettronica - prodotti.” [Online]. Available: http://www.intav.it/htm/ele_prodotti.htm
- [9] L.Deri, “Network management,” 2005.
- [10] I. Medical, “Cardio-aid 200b,” Innomed, Tech. Rep., 2006.
- [11] —, “Heartscreen 80g-l,” Innomed, Tech. Rep., 2006.
- [12] —, “Innocare-s,” Innomed, Tech. Rep., 2006.
- [13] V. Speciali, “Ambulanza di soccorso avanzato serie t5,” G.G.G. Elettromeccanica, Tech. Rep., 2006.
- [14] Spencer, “Spencer 202,” Spencer Italia Life Support, Tech. Rep., 2006.
- [15] Truma, “Ventilatore trumavent.” [Online]. Available: http://www.truma.com/truma05/it/prodotti/detail1_it_75429.html
- [16] R. WAECO, “Me20 - thermobox frigorifero.” [Online]. Available: http://www.elettromedicali.it/product_357.html
- [17] Wikipedia, “Ambulanza.” [Online]. Available: <http://it.wikipedia.org/wiki/Ambulanza>
- [18] —, “Antilock braking system.” [Online]. Available: http://it.wikipedia.org/wiki/Antilock_Braking_System

- [19] —, “Condizionatore.” [Online]. Available: <http://it.wikipedia.org/wiki/Condizionatore>
- [20] —, “Gprs.” [Online]. Available: <http://it.wikipedia.org/wiki/GPRS>
- [21] —, “Iso.” [Online]. Available: <http://it.wikipedia.org/wiki/Iso>