



Manuale d'uso

User Manual

→ **LEGGI E CONSERVA
QUESTE ISTRUZIONI** ←
**READ AND SAVE
THESE INSTRUCTIONS**



Vogliamo farvi risparmiare tempo e denaro!

Vi assicuriamo che la completa lettura di questo manuale vi garantirà una corretta installazione ed un sicuro utilizzo del prodotto descritto.

We wish to save you time and money!

We can assure you that a thorough reading of this manual will guarantee correct installation and safe use of the product described.

INDICE

INDICE	5
1. Generalità	3
2. Installazione del Server OPC	3
2.1 Installazione DLL OPC Proxy	3
3. Configurazione del Server	4
3.1 DRIVER.INI	4
3.2 ND_0.NCF	4
3.3 Descrittori periferica: file .VAR.....	5
3.4 CARELSERVER.INI.....	6
4. Configurazione DCOM.....	7
5. Specifiche tecniche	11

1. Generalità

Tramite CAREL OPC Server qualsiasi applicazione Windows OPC Client, ovvero Scada/HMI, supervisori, gestionali o altri applicativi, può comunicare con tutti i controlli CAREL dotati di interfaccia verso supervisore.

L'applicazione OPC Client può utilizzare le modalità definite dallo standard OPC per interagire con tutte le variabili messe a disposizione dagli strumenti.

2. Installazione del Server OPC

Per il funzionamento del server OPC è necessario che il sistema sia abilitato a supportare DCOM (Win98 o superiore) e siano installate le DLL OPC proxy (vedi capitolo 2.1).

Il server *CarelOpcServer.exe* deve essere copiato in una sotto-directory assieme ai seguenti file:

Drvmng32.dll	Driver protocollo CAREL
Drvmng.lwm	Tabella periferiche CAREL low level
CarelOpcServer.exe	Server OPC
WtOPCSvr.DLL	Libreria OPC server
Driver.INI	Configurazione Driver protocollo CAREL: porte seriali, protocollo, baudrate
ND_0.NCF	Configurazione periferiche CAREL
*.VAR	Descrittori per ogni tipo di periferica CAREL

Una volta copiati i file, il server deve essere registrato tramite il file batch *RegisterCarelOpcServer.bat*, oppure eseguendo *CarelOpcServer.exe* con l'opzione della linea comandi */r* (o, equivalentemente, *-r*, */R*, *-R*). In questo modo l'eseguibile inserisce nel registro di Windows le chiavi necessarie al suo uso e termina immediatamente.

A questo punto il server è operativo e può essere attivato da qualsiasi client OPC.

Per eventualmente eliminare la registrazione dell'OPC server occorre lanciare *UnregisterCarelOpcServer.bat* oppure eseguire *CarelServer.exe* con l'opzione della linea comandi */u* (o, equivalentemente, *-u*, */U*, *-U*); l'eseguibile eliminerà dal registro di Windows le chiavi necessarie al suo uso e terminerà immediatamente.

2.1 Installazione DLL OPC Proxy

L'installazione e la registrazione delle DLL OPC è necessaria solo se nel sistema host non è già operante un altro server OPC, oppure, non siano già state installate dal setup del client OPC che si intende utilizzare.

Per l'installazione entrare nella directory "OPC Data Access Components" e lanciare per Win98 il programma di setup *OPC_DA20_Components.exe*, mentre per Win2000 o NT4 lanciare dalla sotto-directory Win2000 il programma *OPC_DA20_2kComponents.exe*. Il setup provvederà a copiare e a registrare le seguenti DLL:

```
opc_aeps.dll
opccomn_ps.dll
OPCDAAuto.dll
OpcEnum.exe
OPCProxy.dll
ACTXPRXY.DLL
COMCAT.DLL
```

3. Configurazione del Server

Il funzionamento di CarelOpcServer.exe deve essere configurato tramite i seguenti files:

DRIVER.INI
ND_0.NCF
CARELSERVER.INI
*.VAR

I files DRIVER.INI e ND_0.NCF possono essere editati manualmente, oppure essere prodotti dal setup del supervisore CAREL "PlantVisor" e poi copiati nella directory del server. I file *.VAR possono essere reperiti dalla directory "PlantVisor\Servercomponents\UnitT" del "PlantVisor" o richiesti a CAREL.

3.1 DRIVER.INI

Il file DRIVER.INI definisce la configurazione delle linee seriali fisiche utilizzate per le periferiche CAREL e il protocollo da usare.

Il file è nel formato "INI" di Windows. Vengono riconosciute la sezione [config] e le stringhe LINE1, LINE2, ..., LINE8 per le linee logiche 1, 2, ..., 8. Per ogni linea 'logica' (si fa riferimento al numero impostato nel file NCF) è possibile impostare la porta seriale a cui è fisicamente collegata (COM1, COM2, COM3, ...), la velocità di linea (baudrate) e il protocollo utilizzato.

La sintassi è la seguente:

LINEx = p, b, m	x	1..8	Indice logico della linea seriale che compare nel file NDS
	p	1..n	porta seriale fisica: 1 = COM1, 2 = COM2, ... n = COMn
	b	1..5	Baudrate: 1 = 1200 2 = 2400 3 = 4800 4 = 9600 5 = 19200
	m	485	Impone l'utilizzo del protocollo 485 e del driver CAREL serial.vxd (W95,W98) serial.sys (W2000,NT) per il convertitore PC485KIT00.
		485_232	Impone l'utilizzo del protocollo 485 con convertitori "intelligenti": PCGATE, CVSTDU***0. Se il parametro viene omissso la linea opera in modalità 422/232 utilizzando il protocollo CAREL 2.0 per le periferiche in Rs422 di vecchia produzione.

3.2 ND_0.NCF

Il file rappresenta la lista di tutte le periferiche presenti nell'impianto. Per ogni periferica sono indispensabili i seguenti dati:

- Indice logico della linea seriale a cui è collegata (vedi file DRIVER.INI, Line1, 2, ...).
- Indirizzo seriale fisico dello strumento.
- Tipo periferica espresso come nome del file .VAR senza estensione. Il descrittore (.VAR) contiene la lista delle variabili rese disponibili dalla periferica.
- Descrizione della periferica (dipenderà dalla funzione che la stessa svolge nell'impianto). Esso costituirà il prefisso di tutti i nomi delle variabili OPC associate alla periferica in questione.
- Indice progressivo della periferica.

Del file, che è nel formato **INI** di Windows, deve essere considerata solo la sezione [Units] (le altre sono riservate e ad uso interno del supervisore CAREL).

[Units]

LineIdx, Address, "Nome file .VAR", "Descrizione unità", GlobalIndex, "CAREL reserved", CAREL reserved

.....

LineIdx, Address, "Nome file .VAR", "Descrizione unità", GlobalIndex, "CAREL reserved", CAREL reserved

Nome campo	Tipo	Significato
LineIdx	Integer	Indice logico linea (vedi LINEx = nel file DRIVER.INI)
Address	Integer	Indirizzo fisico della periferica
Nome file .VAR	Stringa tra apici	Nome del file .VAR che descrive la periferica
Descrizione unità	Stringa tra apici	Nome simbolico della periferica (dipenderà dalla funzione che la stessa svolge nell'impianto)
GlobalIndex	Integer	Indice progressivo della periferica
CAREL reserved		Riservato ad uso CAREL
CAREL reserved		Riservato ad uso CAREL

Tab. 3.1

Esempio:

[Units]

1,001,"ir32un_t", "ir32 1", 1,"",0

1,002,"ir32un_t", "ir32 2", 2,"",0

1,003,"ir32un_t", "ir32 3", 3,"",0

1,004,"ir32", "ir32C 1", 4,"",0

Definisce una installazione costituita da 3 IR Universali di indirizzo da 1 a 3 e un IR per Refrigerazione di indirizzo 4.

3.3 Descrittori periferica: file .VAR

I file .VAR descrivono l'elenco di tutte le variabili messe a disposizione da una determinata periferica, il loro tipo, indice e attributi di input o output. A partire da esso viene generato lo spazio dei nomi OPC, vedi **Specifiche Tecniche**. Dovranno esistere tanti file .VAR quanti sono i tipi di dispositivi CAREL gestiti dal server.

Di tale file *CarelOpcServer* considera solo la sezione [VARLIST], contenente gli attributi e la descrizione delle variabili in lingua inglese.

Formato file .VAR

[VARLIST]

“VAR_NAME”; Type; Addr; Rw; “VarDescription”; “Min value”; “Max value”; optional fields

.....

“VAR_NAME”; Type; Addr; Rw; “VarDescription”; “Min value”; “Max value”; optional fields

[optional section 1]

.....

[optional section 2]

.....

Il significato dei campi è il seguente:

	Tipo	Significato	Esempio 1	Esempio 2
VAR_NAME	stringa tra apici max 30 caratteri	nome variabile interno	“SET_TEMP”	“HI_TEMP_AL”
Type	integer	tipo variabile: 1 = digitale 2 = analogica 3 = intera 4 = digitale – allarme 5 = intera – unsigned Nota¹	2	4
Addr	long	indice variabile	4	30
Rw	integer	Permesso lettura scrittura 1 = Read (output only) 2 = Write (input only) 3 = Read/write (input/output) I bit da considerare sono solo B0 e B1. Il valore Rw va quindi posto in AND con 3: Rw & 0X3	3	1
VarDescription	stringa tra apici max 50 caratteri	Descrizione della variabile	“Temperature set point”	“High temperature alarm”
Min val	stringa tra apici max 50 caratteri	minimo valore di una variabile di input o indice di un'altra variabile con lo stesso significato	“-10.0”	“”
Max value	stringa tra apici max 50 caratteri	minimo valore di una variabile di input o indice di un'altra variabile con lo stesso significato	“20.0”	“”
Optional fields	CAREL può aggiungere altri campi separati da punto e virgola			

Tab. 3.2

Note¹: la variabile di tipo 4 è ricevuta con il tipo 1 (digitale), la variabile di tipo 5 è trasmessa/ricevuta con il tipo 3 (intera).

3.4 CARELSERVER.INI

Le funzioni OPC del server possono essere configurate tramite il file *CarelServer.ini* che deve contenere la sezione “[init]”. All’interno di questa sezione possono essere assegnati valori per i seguenti parametri: “ServerRate”, “UpdateClusterSize”, “DebugMode”, “NoClientsTimeout”, “OfflineTime”.

I nomi della sezione e dei parametri, non sono “case sensitive”. La stessa configurazione può anche essere effettuata tramite il registro di Windows, creando una chiave \HKEY_CURRENT_USER\Software\CAREL\CarelServer\init contenente uno o più dei parametri (valori DWORD) sopra elencati. Se il file *CarelServer.ini* non viene trovato, l’applicazione esamina il registro di Windows.

Significato e valore di default dei parametri di configurazione:

- “ServerRate”: velocità di aggiornamento in ms delle connessioni client. Il valore di default è 250 ms.
- “UpdateClusterSize”: numero massimo delle tag aggiornate durante un singolo ciclo del server. Il valore di default è 20.
- “DebugMode”: se impostato ad un valore diverso da zero, indica all’applicazione di inserire nel file *CarelServer.dbg* (file in formato testo) dei messaggi di debug relativi all’esecuzione del programma. Se il file *CarelServer.dbg* non esiste viene creato, altrimenti i messaggi vengono aggiunti in coda al file. Si raccomanda di non eseguire a lungo l’applicazione in modalità debug, dato che i messaggi di debug sono numerosi e quindi il file *CarelServer.dbg* può assumere in breve dimensioni notevoli. Il valore di default per questo parametro è 0.
- “NoClientsTimeout”: tempo in ms dopo il quale, se non ci sono client collegati, l’applicazione termina automaticamente. Il valore di default è 5000 ms.
- “OfflineTime”: tempo di offline in ms per le periferiche CAREL che non rispondono. Il valore di default è 90000 ms.

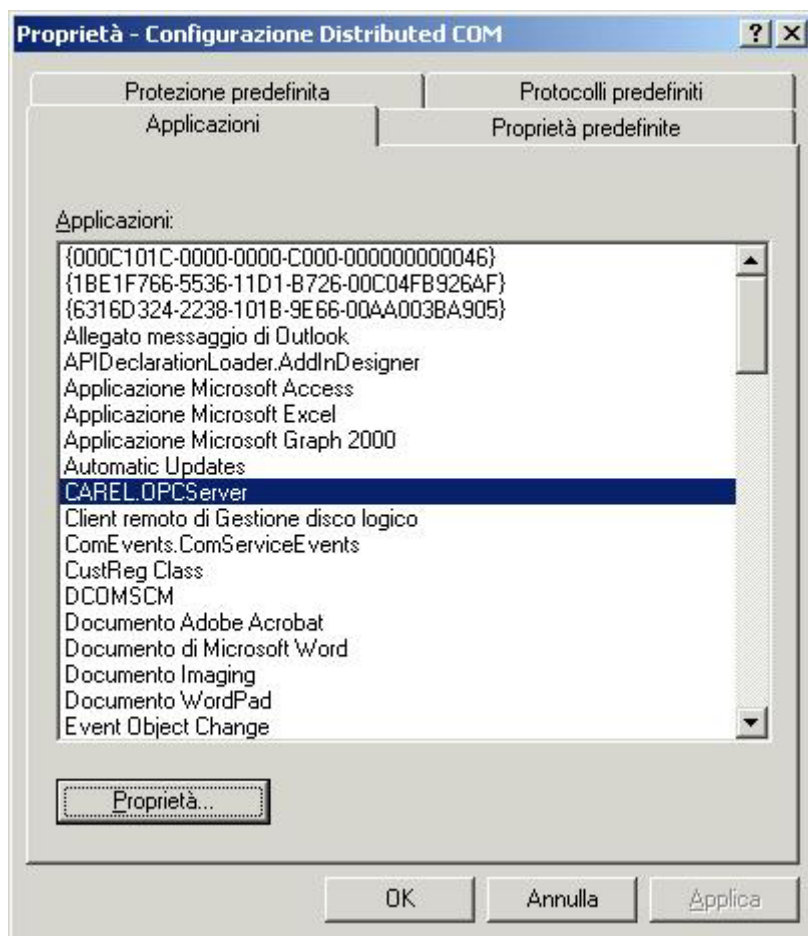
L’applicazione crea e aggiorna il file di testo *CarelServer.log*, che contiene i messaggi di stato della comunicazione con le periferiche CAREL ed eventuali messaggi d’errore.

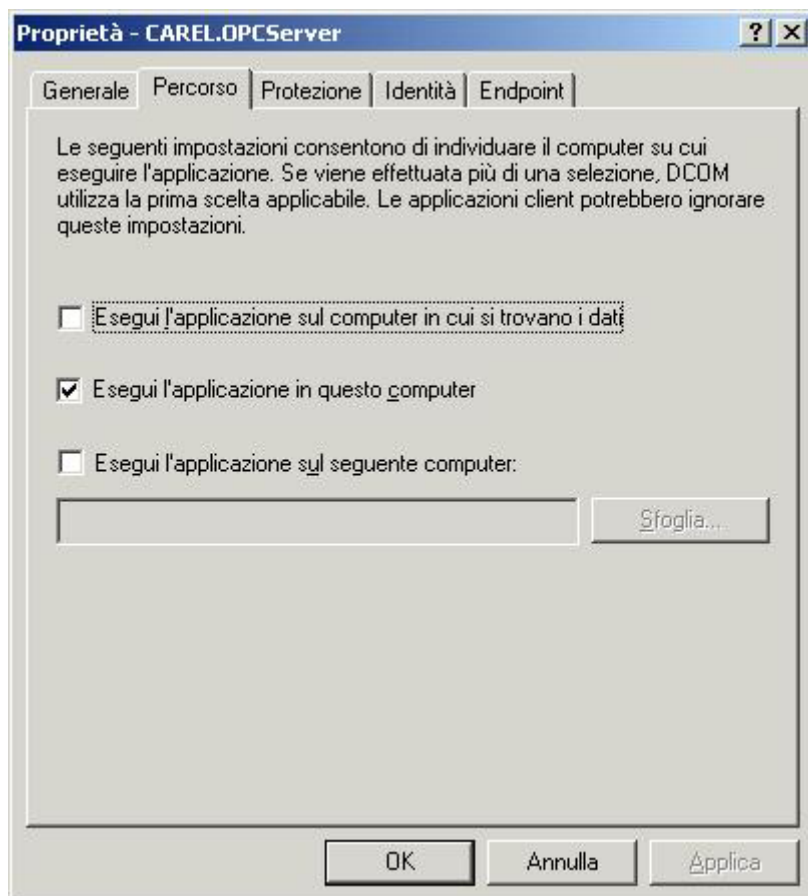
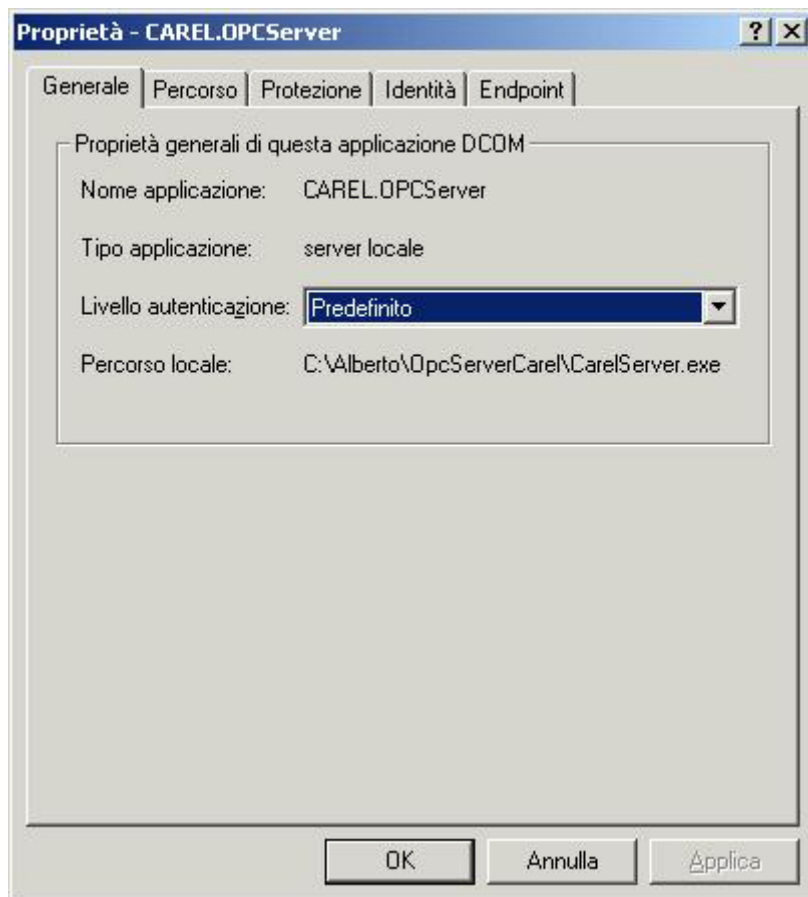
4. Configurazione DCOM

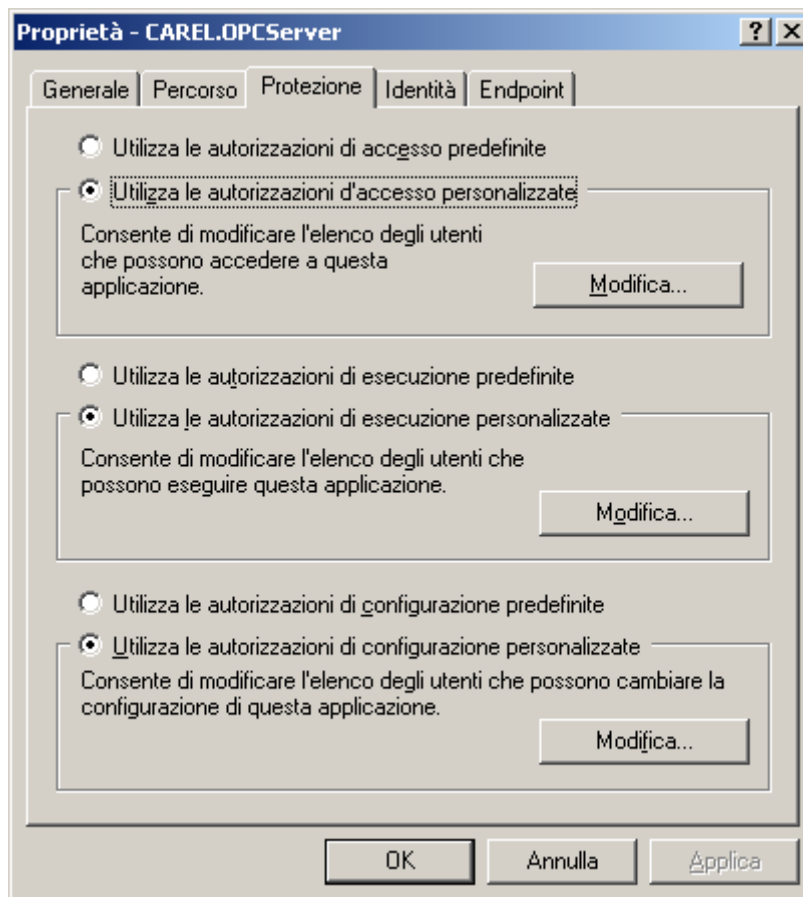
Per utilizzare il server OPC da un PC remoto è necessario configurare la funzione DCOM del sistema operativo tramite il programma DCOMCNFG.EXE. Questo programma è già installato dal sistema operativo per Win98 e Win2000 (NT4); eventuali aggiornamenti vengono installati implicitamente da Internet Explorer 5 oppure sono scaricabili dal sito della Microsoft.

Le seguenti immagini illustrano la configurazione DCOM dal lato server.

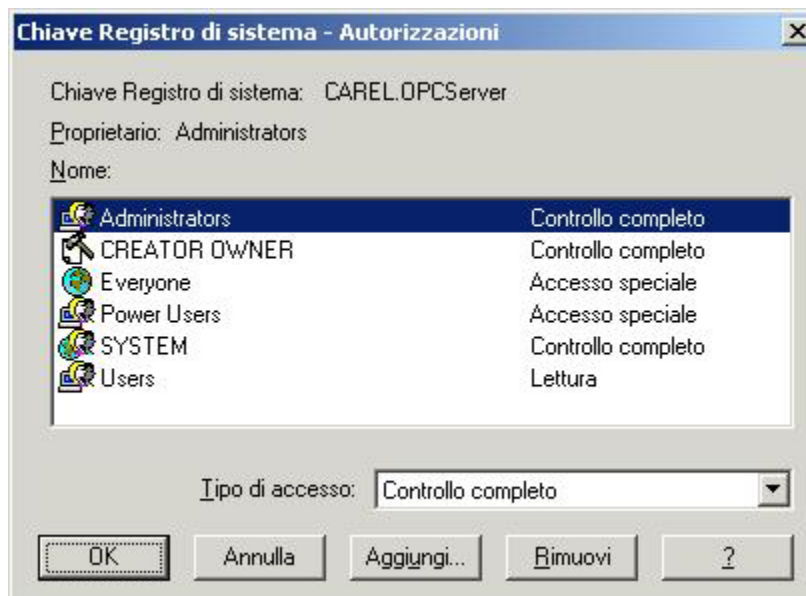
Per i test iniziali è consigliabile non utilizzare particolari protezioni e consentire l'accesso a "Everyone". Per quanto riguarda la proprietà "Identità" è essenziale che il server venga eseguito sempre da un utente specifico (quello con il quale è stato registrato), perché, altrimenti, due diversi client potrebbero causare l'esecuzione di due istanze diverse di *CarelOpcServer.exe* e la seconda istanza si chiuderebbe automaticamente, emettendo nel file Carelserver.log il messaggio: "Failure in the mutex object creation". Infatti in *CarelOpcServer.exe* è stato inserito un controllo sulla presenza di un'altra istanza attiva dell'applicazione, per evitare problemi nella gestione della libreria di comunicazione DRVMNG32.DLL.

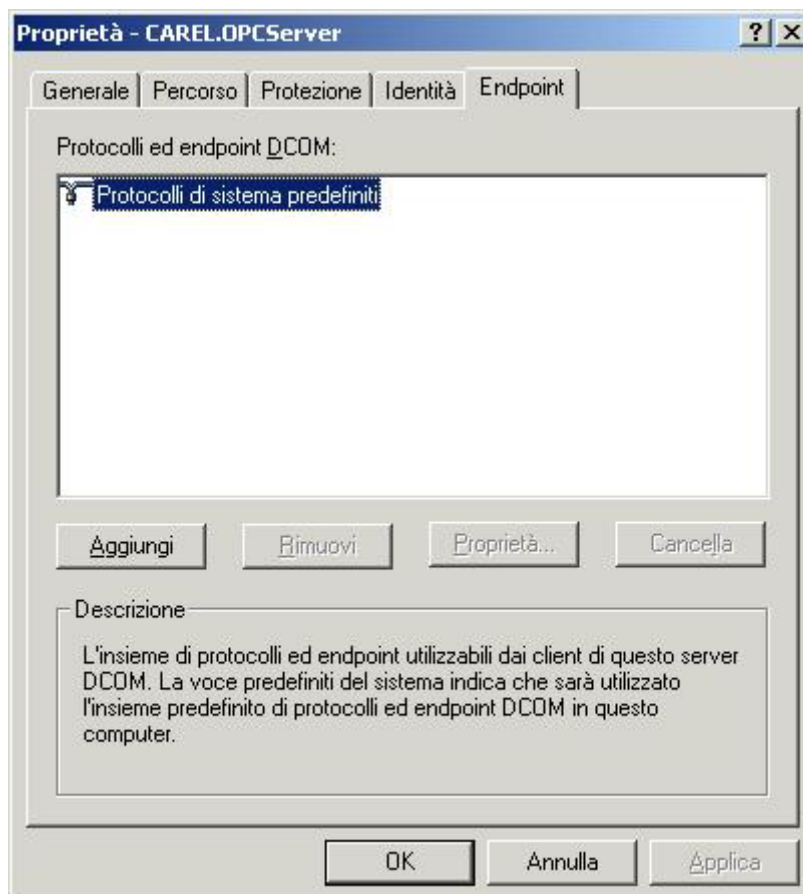
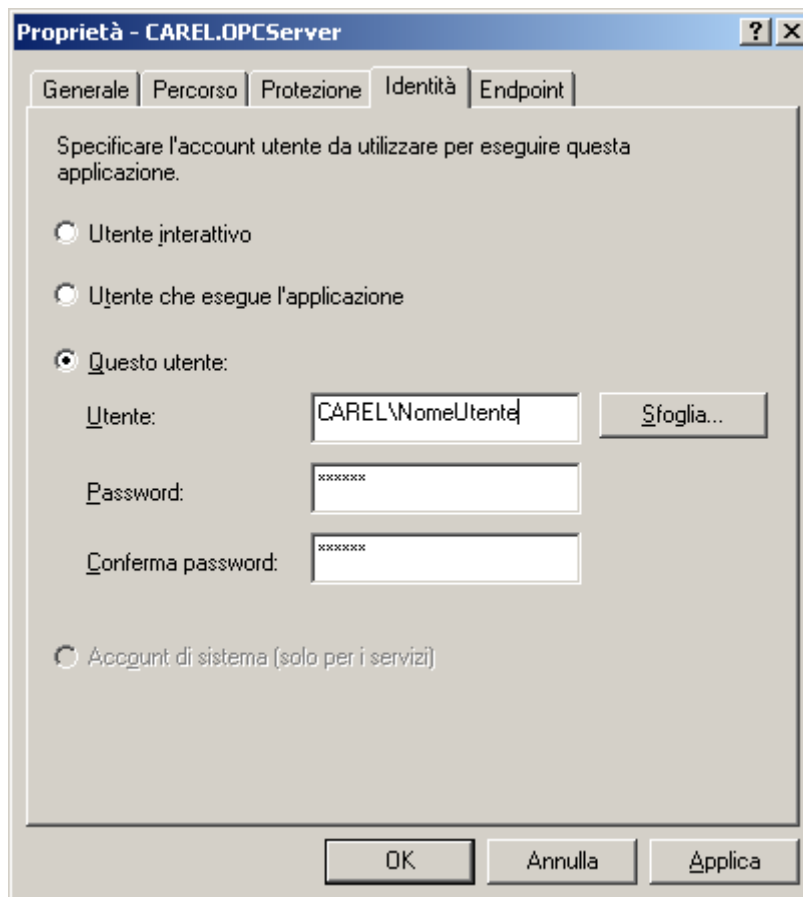






Tramite il tasto “Modifica”:





5. Specifiche tecniche

OPC Server è conforme alle specifiche “Data Access Custom Interface Standard Version 2.05A”.

Sistemi operativi compatibili: W98SE, W2000.

Implementazione: OutProc Server.

Il server supporta le letture sincrone dai dispositivi dei valori delle tag (richieste SyncIO::Read con parametro dwSource impostato a OPC_DS_DEVICE).

Lo spazio dei nomi delle “Item” create dall’OPC server viene definito tramite i file ND_0.NCF e *.VAR. Per ogni “Item” il nome ha il seguente formato:

<descrizione unità>.<descrizione variabile>

come nel seguente esempio:

ir32MEM100 1.Room temperature

La descrizione dell’unità viene ricavata dal file ND_0.NCF, quella della variabile dal file .VAR relativo al tipo di unità. Gli eventuali caratteri ‘.’ presenti nelle descrizioni vengono sostituiti con il carattere ‘_’.

I tipi standard OPC vengono ricavati dai file *.VAR secondo la seguente tabella

Codice	Tipo	Range valori	Tipo OPC
1,4	Digitale	0/1	BOOL
2	Analogico	da -3276.8 a +3276.7	FLOAT
3,5	Intero	da -32768 a +32767	SHORT

Tab. 5.1

Lo stato delle variabili OPC viene definito a partire da quello riportato dal driver di protocollo CAREL:

Stato variabile	Stato periferica	Qualità OPC	Substatus
Non acquisito	Non connessa	Bad=00	0110 = Comm failure
Non acquisito	Connessa	Uncertain=01	0000 = Non specific
Acquisito	Non Connessa	Bad=00	0101 = Last known value
Acquisito	Connessa	Good=11	0000 = Non specific

Tab. 5.2

Lo stato “Non connesso/connesso” di ogni periferica è ricavabile dalla variabile digitale di indirizzo 0. In ogni file *.VAR per la digitale 0 è definito il simbolo “**no connection**” che il server rende disponibile sempre con attributo “Quality good” indipendentemente dallo stato della periferica.

Stato periferica	Qualità OPC	Nome_strumento.no connection
Non connessa	Good	True
Connessa	Good	False

Tab. 5.3

CAREL si riserva la possibilità di apportare modifiche o cambiamenti ai propri prodotti senza alcun preavviso.

INDEX

1.	<i>Introduction</i>	15
2.	<i>Installation of the OPC Server</i>	15
2.1	<i>Installation of the OPC Proxy DLLs</i>	15
3.	<i>Configuring the Server</i>	16
3.1	<i>DRIVER.INI</i>	16
3.2	<i>ND_0.NCF</i>	16
3.3	<i>Device descriptor: .VAR files</i>	17
3.4	<i>CARELSERVER.INI</i>	18
4.	<i>DCOM configuration</i>	19
5.	<i>Technical specifications</i>	23

1. Introduction

From the CAREL OPC Server, any Windows OPC Client application, that is, Scada/HMI, supervisors, management programs or other applications, can communicate with all CAREL controllers fitted with the supervisor interface. The OPC Client application can use the mode defined by the OPC standard to interact with all the variables made available by the instruments.

2. Installation of the OPC Server

The operation of the OPC server requires a system with DCOM support enabled (Win98 or higher) and the installation of the OPC proxy DLLs (see chapter 2.1).

The **CarelOpcServer.exe** server must be copied to a sub-directory, together with the following files:

<i>Drvmng32.dll</i>	CAREL protocol driver
<i>Drvmng.lwm</i>	Table of CAREL low level peripherals
<i>CarelOpcServer.exe</i>	OPC server
<i>WtOPCSvr.DLL</i>	OPC server library
<i>Driver.INI</i>	CAREL protocol driver configuration: serial ports, protocol, baud rate
<i>ND_0.NCF</i>	CAREL peripheral configuration
<i>*.VAR</i>	Descriptors for each type of CAREL peripheral

Once the files have been copied, the server must be registered by the batch file **RegisterCarelOpcServer.bat**, or alternatively by running **CarelOpcServer.exe** with the command line option **/r** (or, equivalently, **-r**, **/R**, **-R**). In this way, the executable file adds the keys required for its operation to the Windows registry and then ends immediately.

At this point, the server is operating and can be activated from any OPC client.

To delete, if necessary, the registration of the OPC server, run **UnregisterCarelOpcServer.bat** or alternatively run **CarelServer.exe** with the command line option **/u** (or, equivalently, **-u**, **/U**, **-U**); the executable file will remove the keys required for its operation from the Windows registry and then end immediately.

2.1 Installation of the OPC Proxy DLLs

The installation and the registration of the OPC DLLs is only required if the host system is not already running another OPC server, or alternatively, they haven't already been installed during the setup of the OPC client.

To install the components, go to the "OPC Data Access Components" directory and, for Win98, run the setup program **OPC_DA20_Components.exe**, while for Win2000 or NT4, go to the Win2000 sub-directory and run the program **OPC_DA20_2kComponents.exe**. The setup procedure will copy and register the following DLLs:

```
opc_aeps.dll
opccomn_ps.dll
OPCDAAuto.dll
OpcEnum.exe
OPCProxy.dll
ACTXPRXY.DLL
COMCAT.DLL
```


Example:

```
[Units]
1.001,"ir32un_t", "ir32 1", 1,"",0
1.002,"ir32un_t", "ir32 2", 2,"",0
1.003,"ir32un_t", "ir32 3", 3,"",0
1.004,"ir32", "ir32C 1", 4,"",0
```

This defines an installation made up of 3 IR Universal, addresses from 1 to 3, and one IR for Refrigeration, address 4.

3.3 Device descriptor: .VAR files

.VAR files contain the list of all the supervisor variables exported from a given controller. Each variable has attributes like index, type and read/write permission. **CarelOpcServer** generates the OPC namespace from this list, see **Technical Specifications**.

For each CAREL device type connected to the server a .VAR file must exist.

CarelOpcServer reads only the [VARLIST] section, that contains the attributes and descriptions of the variables in English language.

.VAR file format

[VARLIST]

“VAR_NAME”; Type; Addr; Rw; “VarDescription”; “Min value”; “Max value”; optional fields

.....

“VAR_NAME”; Type; Addr; Rw; “VarDescription”; “Min value”; “Max value”; optional fields

[optional section 1]

.....

[optional section 2]

.....

The meaning of each field is the following:

	Type	Meaning	Example 1	Example 2
VAR_NAME	quoted string max 30 characters	Internal variable name	“SET_TEMP”	“HI_TEMP_AL”
Type	integer	variable type: 1 = digital 2 = analog 3 = integer 4 = digital – alarm 5 = integer – unsigned Note¹	2	4
Addr	long	variable index	4	30
Rw	integer	Read/write permission 1 = Read (output only) 2 = Write (input only) 3 = Read/write (input/output) The meaningful bits are: b0 and b1. The Rw value is anded with 3: Rw & 0X3	3	1
VarDescription	quoted string max 50 characters	Variabile description	“Temperature set point”	“High temperature alarm”
Min val	quoted string max 50 characterschar	minimum value of an input variable or index of another variable with the same meaning	“-10.0”	“”
Max value	quoted string max 50 characterschar	maximum value of an input variable or index of another variable with the same meaning	“20.0”	“”
Optional fields	CAREL may add others fields separated by semicolon			

Table 3.2

Note¹: variable type 4 is received with type 1 (digital), variable type 5 is transmitted /received with type 3 (integer).

3.4 CARELSERVER.INI

The OPC functions of the server can be configured by the **CarelServer.ini** file, which must contain the “[init]” section. This section is used to assign values to the following parameters: “ServerRate”, “UpdateClusterSize”, “DebugMode”, “NoClientsTimeout”, “OfflineTime”. The names of the sections and the parameters are not case sensitive. The configuration can also be performed by the Windows registry, creating a key \HKEY_CURRENT_USER\Software\CAREL\CarelServer\init containing one or more of the parameters (DWORD values) listed above. If the **CarelServer.ini** file cannot be found, the application examines the Windows registry.

Meanings and default values of the configuration parameters:

- “ServerRate”: update speed in ms of the client connections. The default value is 250 ms.
- “UpdateClusterSize”: maximum number of the tags updated during a single server cycle. The default value is 20.
- “DebugMode”: if set to a value other than zero, tells the application to add the run-time program debug messages to the **CarelServer.dbg** file (file in text format). If the **CarelServer.dbg** file does not exist it is created, otherwise the messages are appended to the file. The application should not be run in debug mode for too long, given that there are numerous debug messages and consequently the **CarelServer.dbg** file can become quite large in a short time. The default value for this parameter is 0.
- “NoClientsTimeout”: time in ms after which, if no clients are connected, the application ends automatically. The default value is 5000 ms.
- “OfflineTime”: offline time in ms for the CAREL peripherals that do not respond. The default value is 90000 ms.

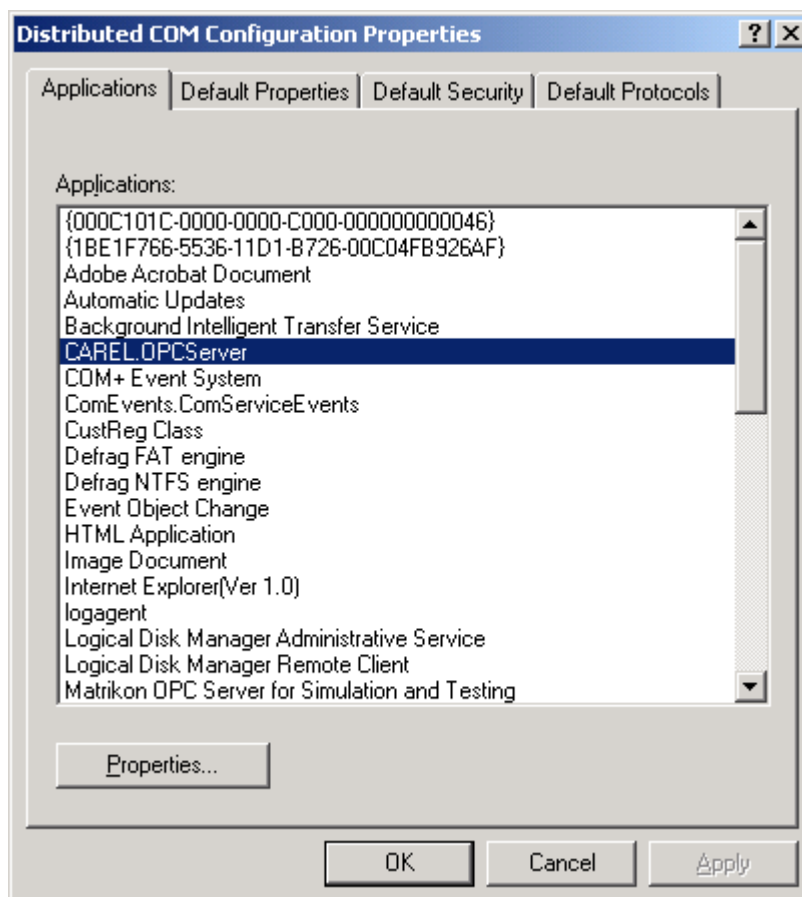
The application creates and updates the text file **CarelServer.log**, which contains the messages on the communication status with the CAREL peripherals and any error messages.

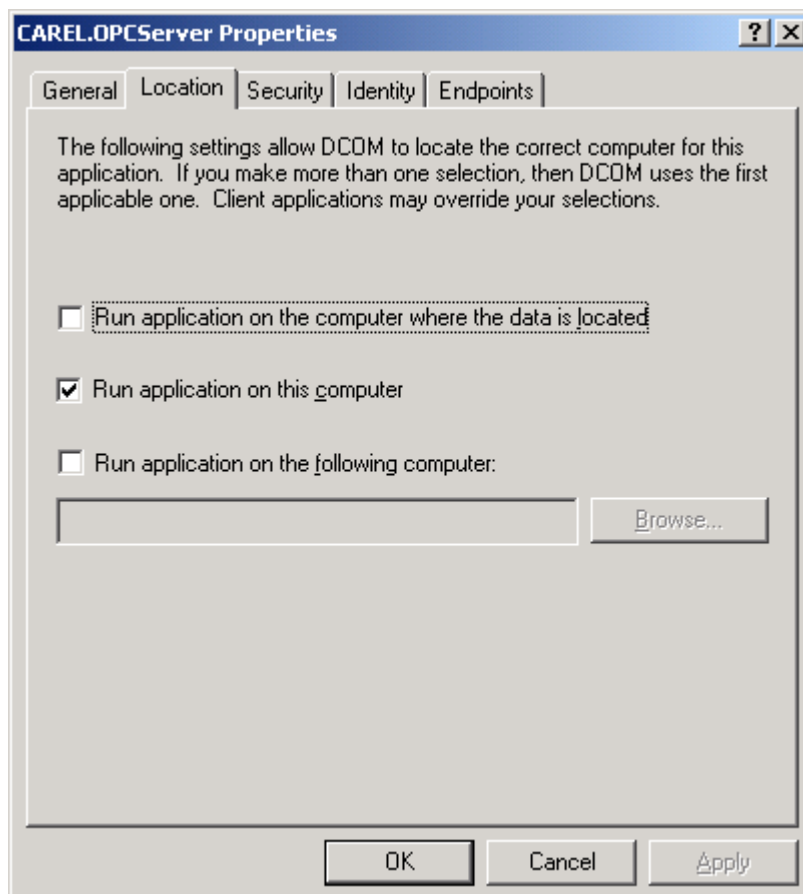
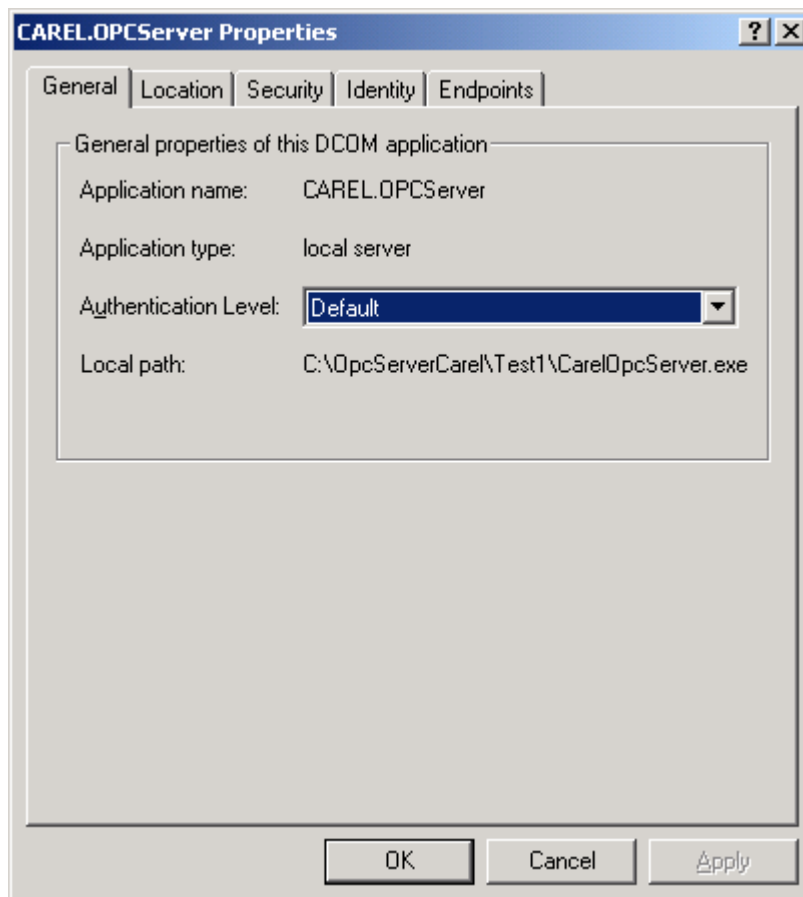
4. DCOM configuration

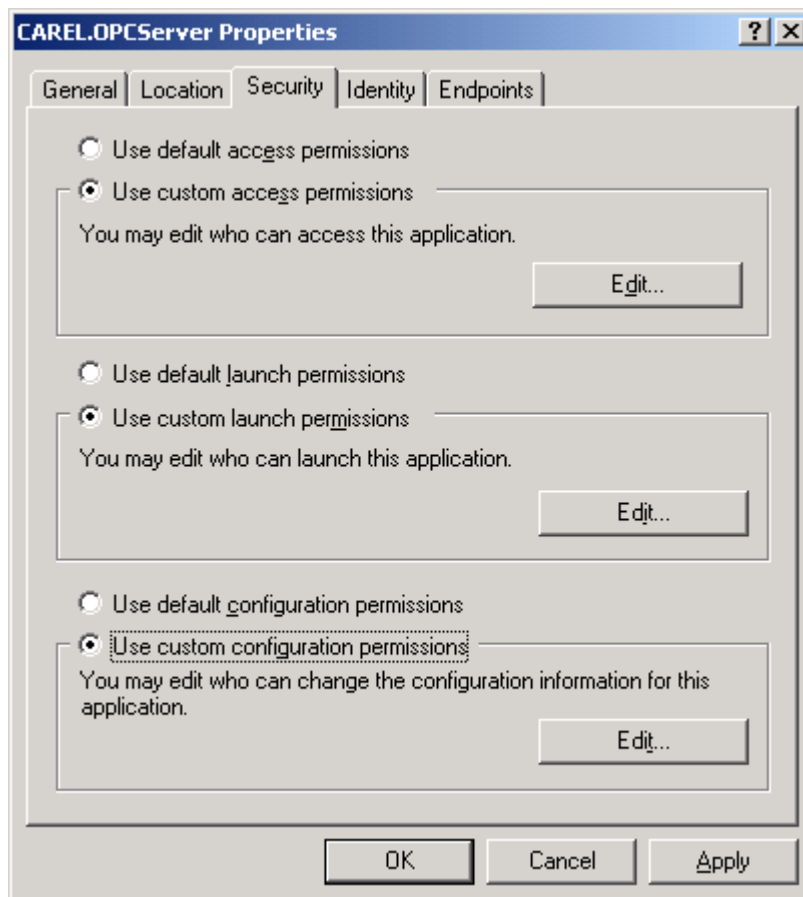
To use the OPC server from a remote PC, the DCOM function of the operating system must be configured using the DCOMCNFG.EXE program. This program is already installed by the operating system for Win98 and Win2000 (NT4); any upgrades are installed implicitly with Internet Explorer 5 or alternatively can be downloaded from the Microsoft site.

The following figures show the DCOM configuration on the server side.

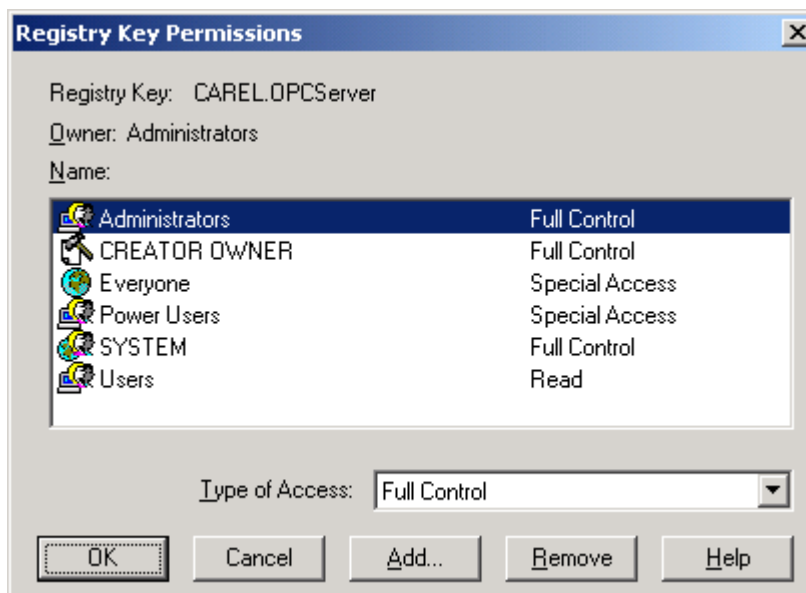
For the initial test it is recommended not to use special protection and allow access to "Everyone". As regards the "Identity" property, the server must always be run by a specific user (the registered user), as otherwise, two different clients may run two different instances of **CarelOpcServer.exe** and the second request would be closed automatically, adding the following message to the **Carelserver.log** file: "Failure in the mutex object creation". In fact, **CarelOpcServer.exe** features a check for the presence of another active request for the application, so as to avoid problems in the management of the communication library DRVMNG32.DLL.

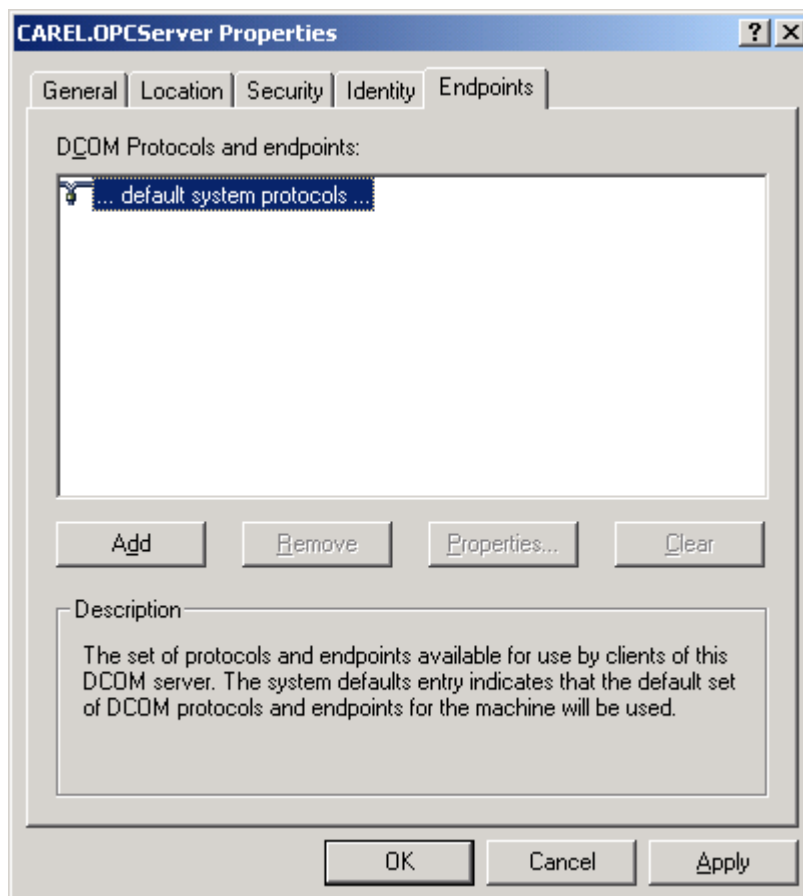
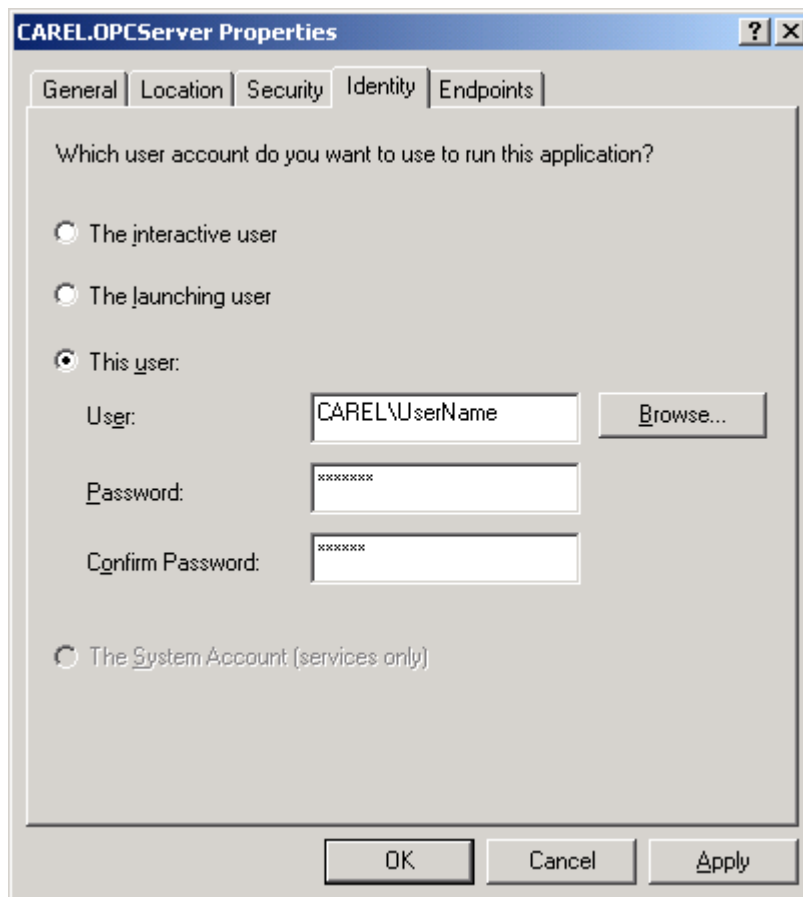






Clicking the "Edit" button:





5. Technical specifications

The OPC Server is compliant with the “Data Access Custom Interface Standard Version 2.05A” specifications.

Compatible operating systems: W98SE, W2000.

Implementation: OutProc Server.

The server supports the synchronous readings from the devices with the values of the tags (required SyncIO::Read with parameter dwSource set to OPC_DS_DEVICE).

The space of the “Item” names created by the OPC server is defined by the ND_0.NCF and *.VAR files. For each “Item”, the name has the following format:

<unit description>.<variable description>

as in the following example:

ir32MEM100 1.Room temperature

The description of the unit is taken from the ND_0.NCF file, and of the variable from the .VAR file relating to the type of unit. Any ‘.’ characters present in the descriptions are replaced by the ‘_’ character.

The standard OPC types are taken from the *.VAR files, according to the following table

Code	Type	Range of values	OPC type
1,4	Digital	0/1	BOOL
2	Analogue	from -3276.8 to +3276.7	FLOAT
3,5	Integer	from -32768 to +32767	SHORT

Table 5.1

The status of the OPC variables is defined based on that shown by the CAREL protocol driver:

Variable status	Peripheral status	OPC quality	Substatus
Not acquired	Not connected	Bad=00	0110 = Comm failure
Not acquired	Connected	Uncertain=01	0000 = Non specific
Acquired	Not Connected	Bad=00	0101 = Last known value
Acquired	Connected	Good=11	0000 = Non specific

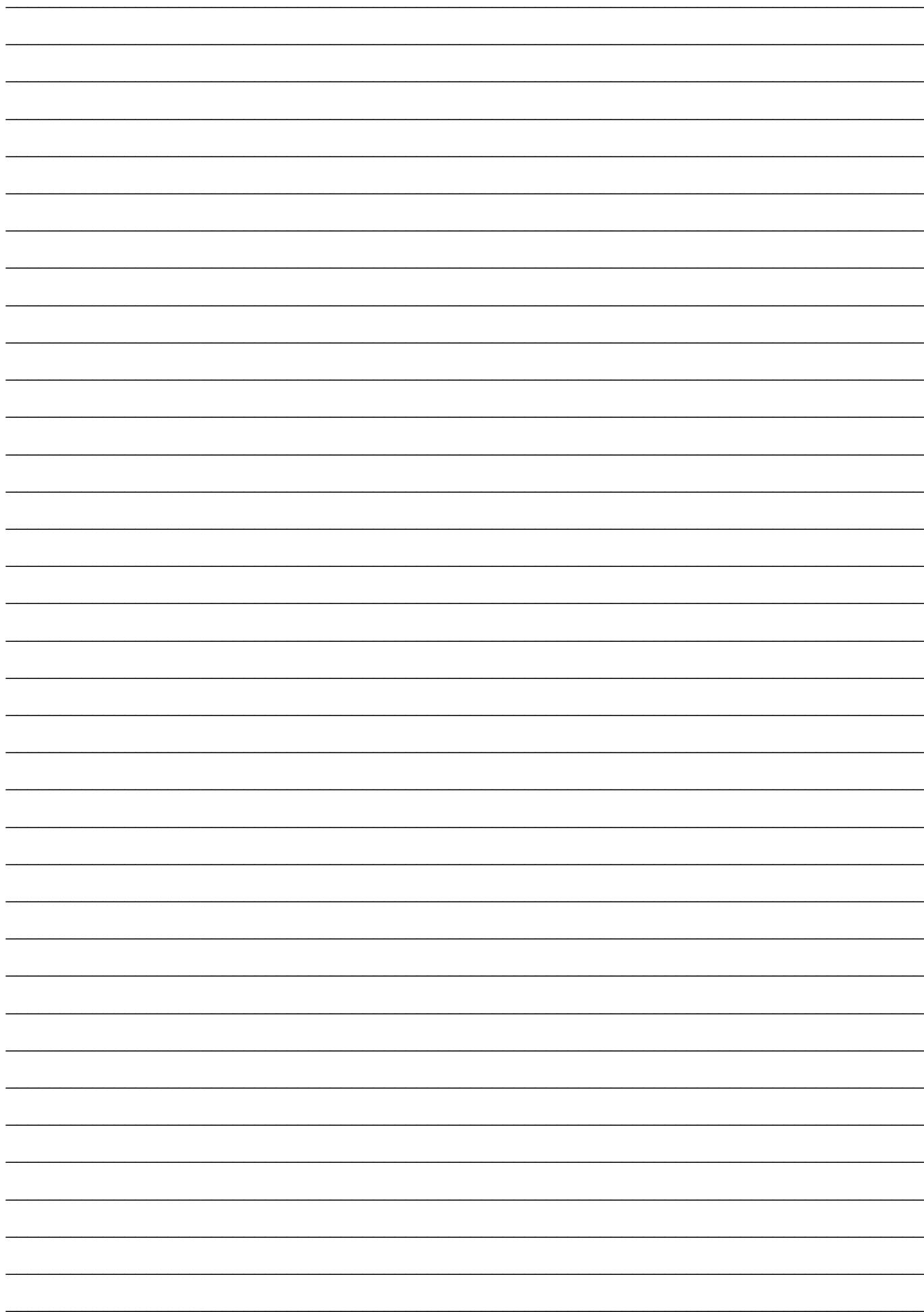
Table 5.2

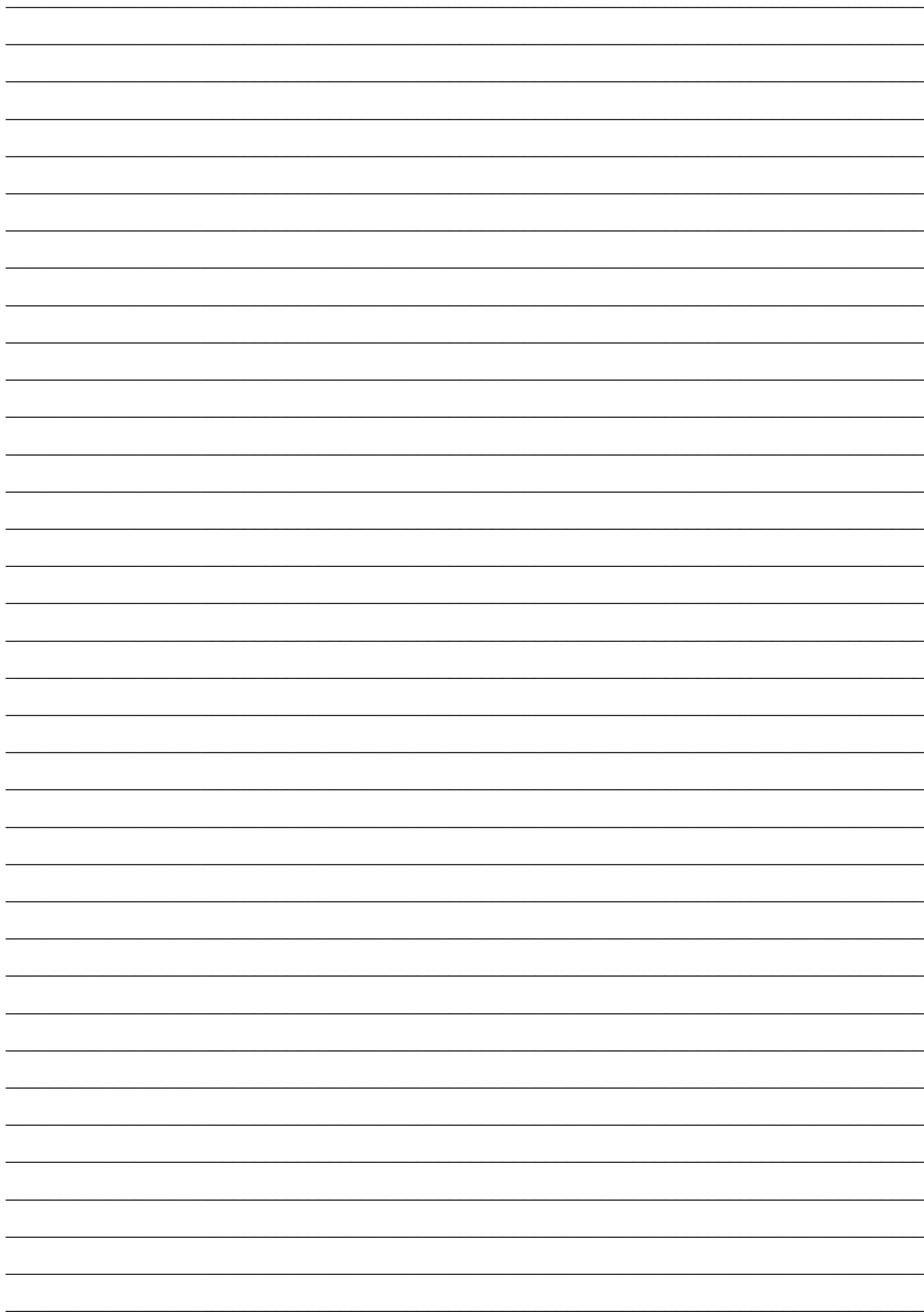
The “Not connected/connected” status of each peripheral can be recognised from the digital variable with address 0. In each *.VAR file, the “no connection” symbol is defined for digital variable 0, which the server always makes available with the “Quality good” attribute, irrespective of the status of the peripheral.

Peripheral status	OPC quality	Instrument name.no connection
Not connected	Good	True
Connected	Good	False

Table 5.3

CAREL SpA reserves the right to make modifications or changes to its products without prior notice.





All trademarks hereby referenced are the property of their respective owners.

CAREL

Technology & Evolution

CAREL S.p.A.
Via dell'Industria, 11 - 35020 Brugine - Padova (Italy)
Tel. (+39) 049.9716611 Fax (+39) 049.9716600
<http://www.carel.com> - e-mail: carel@carel.com

Agency:

Code: +030221140 rel. 1.0 dated 21/01/04