

## Alcuni limiti della gestione tradizionale con gli archivi

### Ridondanza, incongruenza e inconsistenza

È sempre possibile gestire le informazioni di un Sistema Informativo Automatizzato appoggiandosi per la loro memorizzazione esclusivamente a file di dati organizzati secondo i convenzionali metodi supportati dai principali linguaggi di programmazione (organizzazione sequenziale, ad accesso diretto e sequenziale con indice); ne sono un esempio i file con organizzazione sequenziale e *random* supportati sia dal linguaggio Delphi che dal Visual Basic (così come dal vecchio Basic) oppure i file con organizzazione *indexed* supportati dal linguaggio gestionale Cobol.

Tuttavia, la memorizzazione e il trattamento delle informazioni di un Sistema Informativo Automatizzato attraverso **archivi di dati convenzionali** pone una serie di limiti e di inconvenienti che solo un programmatore di provata esperienza può in parte eludere e che hanno portato a ricercare nuove vie per gestire in modo più semplice e flessibile le grandi quantità di informazioni raccolte ed elaborate.

Uno dei principali problemi con cui si ha a che fare consiste nel rischio di **uplicazione** dei dati e nell'introduzione di pericolose **ridondanze** all'interno del Sistema Informativo Automatizzato, per mancanza di integrazione e correlazione dei dati memorizzati. Infatti, *una stessa informazione può comparire più volte in archivi diversi*, in formati (o tracciati record) diversi.

Consideriamo, ad esempio, il caso di un Sistema Informativo Automatizzato di un ospedale. I dati di un paziente sono raccolti ed elaborati in un file (*Pazienti*) grazie a un programma installato su un PC *presso lo sportello di accettazione*. Alcune delle informazioni contenute nel file *Pazienti* compaiono, ma sotto forma di tracciato record diverso, anche in un altro file di dati (*Visite*), gestito da un programma installato su un PC *presso un ambulatorio*. Le stesse informazioni, infine, sono ancora salvate su un altro file di dati (*PazientiOrtopedia*) memorizzato su un PC *in dotazione nel reparto di ortopedia*, dove il paziente è stato ricoverato qualche mese prima. Mostriamo di seguito i loro tracciati record:

Tracciato record di *Pazienti*:

#### Paziente

<u>CodicePaziente</u> (stringa)	Cognome (stringa)	Nome (stringa)	Indirizzo (stringa)	Città (stringa)	Telefono (stringa)	DataRicovero (data)	Reparto (stringa)
------------------------------------	----------------------	-------------------	------------------------	--------------------	-----------------------	------------------------	----------------------

Tracciato record di *Visite*:

#### Visita

<u>Progressivo</u> (numerico)	NominativoPaziente (stringa)	Indirizzo (stringa)	Dottore (stringa)	DataVisita (data)	Ambulatorio (stringa)
----------------------------------	---------------------------------	------------------------	----------------------	----------------------	--------------------------

Tracciato record di *azientiOrtopedia*:

#### PazienteOrtopedia

Codice Paziente (stringa)	Cognome (stringa)	Nome (stringa)	Indirizzo (stringa)	DataRicovero (data)	NLetto (numerico)	Diagnosi (stringa)
------------------------------	----------------------	-------------------	------------------------	------------------------	----------------------	-----------------------

Consideriamo una possibile situazione per il paziente De Virgilio Carlo nei tre file appena descritti:

### Pazienti

P00567	De Virgilio	Carlo	Via Cussanio 7	Fossano	0172/333333	12/12/2006	Ortopedia

### Visite

17876	De Virgilio Carlo	Via Cussanio 7, Fossano	Ghiera Francesca	16/11/2006	Ortopedia 2	
20002	De Virgilio Carlo	Via Cussanio, Fossano	Ceccarini Barbara	03/12/2006	Cardiologia 2	

### PazientiOrtopedia

PZ234	De Virgilio	Carlo	Via Cussanio 7, Fossano	12/12/2006	75	Distorsione caviglia dx con complicazioni
...						

Il metodo di archiviazione ha portato ad avere dati del paziente ripetuti più volte non solo in file diversi, ma addirittura nello stesso file!

Duplicando i dati si può facilmente arrivare all'introduzione di **inconsistenze**, a causa di **incongruenze** che possono verificarsi per il mancato contemporaneo aggiornamento di tutte le occorrenze della stessa informazione: pensiamo al semplice cambio di indirizzo di un paziente, dove, se la modifica non viene apportata nello stesso istante in tutti i file in cui compaiono le informazioni di quel paziente si ha una situazione non omogenea, che riduce l'affidabilità del Sistema Informatico.

Osserviamo, inoltre, che nel file *Visite* l'indirizzo del paziente compare con e senza il numero civico, rendendo più difficoltosa la ricerca e, comunque, ponendo problemi di consistenza dei dati. Ancora, nel file *Pazienti* è presente il campo per l'inserimento della città oltre a quello dell'indirizzo, mentre negli altri file considerati l'informazione sulla città di residenza è direttamente inserita nel campo indirizzo.

Se poi volessimo effettuare una ricerca dei dati medici di un paziente di cui conosciamo il cognome e il nome, l'algoritmo di ricerca dovrà tenere conto che nel file *Visite* questi due valori sono inseriti nello **stesso campo** (*NominativoPaziente*), mentre nei file *Pazienti* e *PazientiOrtopedia* corrispondono a **due campi** distinti.

È evidente che solo un buon livello di integrazione e correlazione tra i dati, supportato da una corretta modellazione della loro struttura (e, nel nostro esempio, da una LAN che permetta di collegare in rete i PC dell'ospedale), **può permettere** di superare gli inconvenienti in modo da avere un unico archivio dei pazienti condiviso, con visualizzazioni dei dati differenziate a seconda dell'ufficio o del reparto che accede agli stessi.