

Foto gallery 6: gestire il *caching* delle immagini

Aggiungere al programma un modulo che gestisca il *caching* delle immagini, in modo da aumentare le prestazioni del programma, soprattutto relativamente alla funzione di filtro.

Note sull'implementazione

Attualmente, la visualizzazione di un'immagine implica il suo caricamento in memoria a partire dal file che la contiene. Nel caso in cui un'immagine venga visualizzata più volte (come nella funzione di filtro), si tratta di un processo che consuma memoria, poiché la stessa immagine viene caricata più volte. Si può rendere il processo molto più efficiente riutilizzando le immagini precedentemente caricate. È un tipico scenario di *caching*.

Caching delle immagini

La visualizzazione di un'immagine è attualmente implementata nel seguente modo:

```
picFoto.Image = Image.FromFile(listaFoto[i].NomeFile);
```

Si può pensare di sostituire il metodo di caricamento con un proprio metodo, collocato in un modulo:

```
picFoto.Image = Immagini.Carica(listaFoto[i].NomeFile);
```

Il metodo carica, invece di caricare l'immagine direttamente dal file, esegue il seguente procedimento:

1. Verifica se l'immagine è già stata caricata in precedenza.
2. Se sì, restituisce l'immagine già caricata.
3. Se no, carica l'immagine e la restituisce.

Naturalmente, per implementare questa funzione è necessario che:

- Le immagini caricate siano memorizzate da qualche parte.
- A ogni immagine sia associata una chiave, in modo che il metodo possa stabilire se l'immagine richiesta è già stata caricata oppure no.

La soluzione consiste nel memorizzare le immagini in un dizionario, nel quale la chiave è rappresentata dal nome del file, mentre il valore dall'immagine stessa (`Image`). Il metodo `Carica()` cerca prima nel dizionario e se trova l'immagine la restituisce. Altrimenti, carica l'immagine, la aggiunge al dizionario e la restituisce.