

# Relazioni

## Iter di gestione delle relazioni

La gestione delle relazioni può venir definita in 3 fasi principali così divise:

### Inizializzazione

Operazione da effettuare nell'initVars, definisce a livello generico le relazioni fra la tabella principale e le tabelle collegate. Per ogni relazione è necessario richiamare il metodo della superclasse

**addDescribeRelation** come definito quì sotto:

**addDescribeRelation(\$tableName, \$keyMapping, \$tipoRelazione, \$alias = null)** Il metodo definisce la relazione fra la tabella principale ed una tabella collegata e prende i seguenti parametri:

- **\$tableName** E' il nome della tabella collegata alla principale
- **\$keyMapping** E' un array contenente i campi della foreign key fra tabella principale e tabella collegata:

```
$keyMapping = array(  
    'PROGNOTE' => 'PROGNOTE'  
);
```

- **\$tipoRelazione** Indica il tipo di relazione (1:1, 1:n, n:n, etc.)
- **\$alias** Indica un alias per la relazione, è necessario nel caso la tabella primaria abbia più relazioni verso la stessa tabella (es. tabella delle località che può essere collegata ad un soggetto sia per il luogo di nascita che per la residenza).

Esempio d'uso:

```
$this->addDescribeRelation('BOR_UTELIV', array('CODUTE' => 'CODUTENTE'), itaModelServiceData::RELATION_TYPE_ONE_TO_MANY);
```

### Definizione operazioni

Risulta necessario definire le operazioni relative all'inserimento, modifica e cancellazione dal database delle righe specifiche sulle tabelle relazionate, questa operazione può venir fatta man mano che le operazioni in questione vengono definite o fornite in blocco in fase di preAggiungi/preAggiorna/preConfermaCancella.

Sono previste 3 operazioni:

## addInsertOperation()

Inserimento di una nuova riga, prende in input 2 parametri:

- **\$table**: nome della tabella collegata alla tabella principale
- **\$alias**: (opzionalmente) l'alias della relazione come definito in fase di inizializzazione.

## addUpdateOperation()

Modifica di una riga preesistente, prende in input 3 parametri:

- **\$table**: nome della tabella collegata alla tabella principale
- **\$keys**: array associativo contenente la primary key della riga da modificare. L'array è in forma 'chiave'⇒\$valore.
- **\$alias**: (opzionalmente) l'alias della relazione come definito in fase di inizializzazione.

## addDeleteOperation()

Eliminazione di una riga preesistente, prende in input 3 parametri:

- **\$table**: nome della tabella collegata alla tabella principale
- **\$keys**: array associativo contenente la primary key della riga da eliminare. L'array è in forma 'chiave'⇒\$valore.
- **\$alias**: (opzionalmente) l'alias della relazione come definito in fase di inizializzazione.

## Restituzione dei dati relazionati

La superclasse, durante la fase di scrittura sul database, interrogherà la pagina che ha originato la richiesta per ricevere i dati relativi alle tabelle relazionate. Risulterà dunque, nelle pagine che fanno uso di relazioni, necessario ridefinire due metodi:

### getDataRelation()

Il metodo getDataRelation restituisce un array contenente tutti i dati relazionati così come vengono letti dal database, quindi i dati definiti precedentemente alle modifiche effettuate. Il metodo riceve in input il tipo di operazione che sta venendo effettuata sul record principale (inserimento/aggiornamento/eliminazione). Il metodo deve restituire in output un array in tale forma:

```
return array(  
    'Alias_Rel1'=>array(  
        array(  
            'Campo1'=>'Valore1_1',  
            'Campo2'=>'Valore1_2',  
            'Campo3'=>'Valore1_3',  
            ...
```

```
    ),  
    array(  
        'Campo1'=>'Valore2_1',  
        'Campo2'=>'Valore2_2',  
        'Campo3'=>'Valore2_3',  
        ...  
    ),  
    ...  
),  
'Alias_Rel2'=>array(  
    ...  
)  
);
```

## getDataRelationView()

Questo metodo restituisce i soli dati modificati di una specifica relazione. Il metodo fornisce in input il nome della tabella e l'eventuale alias della relazione di cui sta richiedendo i dati. I dati devono essere restituiti sotto forma di array di array e contenere unicamente i dati modificati. Nel caso di eliminazione di righe tali righe non devono comparire nell'array restituito da `getDataRelationView()`.

From:

<https://wiki.nuvolaitalsoft.it/> - **wiki**

Permanent link:

[https://wiki.nuvolaitalsoft.it/doku.php?id=sviluppo:cityware\\_relazioni](https://wiki.nuvolaitalsoft.it/doku.php?id=sviluppo:cityware_relazioni)

Last update: **2018/03/19 10:45**

