Subject: grandparent - parent - child link entity
Posted by gpatti on Thu, 30 Dec 2010 16:15:17 GMT
View Forum Message <> Reply to Message

I have a link entity showing valid combinations of two other entities and easily code this using a LINK pattern.
However, I now want to relate that table back to itself using senior and junior relationships.

The completed relationships will give valid junior combinations, based upon valid senior combinations.

However, I can't determine if there is a pattern that will allow me to represent this within the framework. I think it needs a MULTI version of the LINK pattern.

Do you have any suggestions for this?

Thanks,

Graham

Subject: Re: grandparent - parent - child link entity
Posted by AJM on Fri, 31 Dec 2010 09:46:25 GMT
View Forum Message <> Reply to Message

Can you provide me with the relevant database schema so that I have something more concrete with which to work? I'm afraid your description is not accurate enough.

Subject: Re: grandparent - parent - child link entity
Posted by gpatti on Fri, 31 Dec 2010 15:22:16 GMT
View Forum Message <> Reply to Message

Hi Tony,

I'm afraid I don't have an easy way to draw the schema so I hope the following rambling description will suffice. Please let me know if this creates additional questions.

The application will be to allow a person to create a dance routine for a given dance type from a set number of figures.

Table dance_type
- dance_type_id (PK)
- other columns

Table figure
- figure_id (PK)

- other columns

Table dance_figure (link table allowing many-many between dance_type and figure)
- dance_type_id
- figure_id

Now a new link table is introduced and relationships used to join dance_figure to itself in a many-many relationship

Table follow
- dance_type_id
- figure_id
- follow_id

Two relationships between dance_figure and follow usng figure_id and follow_id as foreign key
1 - dance_type_id and figure id as snr
2 - dance_type_id and follow_id as jnr

The concept is that a dance_type consists of a number of figures. A figure may exist in more than one dance_type (hence many-many link relationship using dance_figure).

Any figure (within a particular dance_type) can have one or more allowed following figures.

Therefore, I need to be able to able to specify for each row in dance_figure, which other figures are valid follows (it isn't allowed to mix dance_types, so dance_id as follow does not need to be included in the follow table).

In terms of the framework, I wanted to be able to navigate from the dance_type list screen to a link screen allowing the valid dance_figures to be selected (this part is easy and is complete). However, for a selected dance_figure I then want to navigate to another link screen to select the valid follows.

Subject: Re: grandparent - parent - child link entity
Posted by AJM on Fri, 31 Dec 2010 17:15:00 GMT
View Forum Message <> Reply to Message

Are you aware that a LINK1 pattern is not the only way to maintain a one-to-many relationship? A many-to-many is actually two separate one-to-many relationships, and you can only deal with one of those relationships at a time.

The MENU system already contains examples of different patterns to maintain the same relationship:

in 'List Task (menu)' there are 'Menu Items (1)', 'Menu Items (2)' and 'Menu Items (3)'.
in 'List Task (process)' there are 'Nav Buttons (1)', 'Nav Buttons (2)' and 'Nav Buttons (3)'.

It is not possible to have a MULTI1 version of the LINK1 pattern as they deal with a different number of entities.

You will need to have a non-LINK1 task which maintains the contents of 'dance_figure' so that it can have navigation buttons which allow you to choose a row and jump to a task which maintains 'follow'.

Subject: Re: grandparent - parent - child link entity
Posted by gpatti on Fri, 31 Dec 2010 18:06:21 GMT
View Forum Message <> Reply to Message

Thanks Tony,

I've been using the system for a long time and it's only just dawned on me from your reply that I don't have to use a LINK pattern for a link table. I had never really looked at the (2) and (3) buttons for Nav and Menu. If I had I would have realised.

Thanks,

Graham
