# Re: Dazed and Confused

***Source:***
http://www.tech−archive.net/Archive/Access/microsoft.public.access.tablesdbdesign/2007−01/msg00100.html

- *From*: "tina" <nospam@xxxxxxxxxxx>
- *Date*: Sun, 07 Jan 2007 03:59:35 GMT

> There are also many Requisitions that will go to many Purchase Orders and
> visa versa. After reading your comments, I'm sure this is another many to
> many relationship.

just to clarify your thinking a bit: to discover what type of relationship
exists between two tables, you need to consider how EACH record in TableA is
related to the records in TableB, *and* how EACH record in TableB is related
to the records in TableA.

a one−to−one relationship:
each record in TableA can be related to only one record in TableB, AND each
record in TableB is related to only one record in TableA. this relationship
is modeled with a one−to−one link FROM TableA TO TableB.

a one−to−many relationship:
each record in TableA may be related to many records in TableB, BUT each
record in TableB is related to only one record in TableA. this relationship
is modeled with a one−to−many link FROM TableA TO TableB.

a many−to−many relationship:
each record in TableA may be related to more than one record in TableB, AND
each record in TableB may be related to more than one record in TableA. this
relationship cannot be modeled via a direct link between the two tables. a
third, linking (aka join) table is utilized as the −many side of a
one−to−many relationship with each of the first two:
each record in TableA may be related to many records in TableC, BUT each
record in TableC is related to only one record in TableA. this relationship
is modeled with a one−to−many link FROM TableA TO TableC.
each record in TableB may be related to many records in TableC, BUT each
record in TableC is related to only one record in TableB. this relationship
is modeled with a one−to−many link FROM TableB TO TableC.

hth

"Nanette" <Nanette@xxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
news:84D95727−12D6−4921−A32F−B46A349785A8@xxxxxxxxxxxxxxxx

Hi Jeff,

Thanks for the clarification. I'm beginning to understand better.

The Part Detail Information that is being saved is Changes in Part

Revision,

Alternate Part Numbers, Actionee, Planner, etc. The Part Details table is being used mostly so we can keep track of changes in the above mentioned fields.

There are also many Requisitions that will go to many Purchase Orders and visa versa. After reading your comments, I'm sure this is another many to many relationship.

Does the below Entity Relationship design sound correct?

Purchase Order Details Table (many) to a Purchase Order Table (one)
Purchase Order Table (one) to a Junction table(many) with ID numbers.
Junction table (many) to the Purchase Requisition table (many).
Purchase Requisition Table (one) to Part Details Table (many).
Purchase Requisition Table (one) to Purchase Requisition Details Table

(many).

"Jeff Boyce" wrote:

> Nanette
>
> see comments in−line below...
>
> "Nanette" <Nanette@xxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
> news:34118449−ED0E−42AB−88BD−2272DEF0AF19@xxxxxxxxxxxxxxx
>
> > One to Many or Many to Many Relationship, and how do I
> > connect the
>
> data?
>
> > After many attempts, I'm still not sure, so...
> >
> > I have a Requisition Table and a Reqisition Detail Table.
> > Each
>
> Reqisition

will contain many parts numbers and many of the same part numbers will

be

on

multiple Requisitions.

So, a given Requisition might have many Parts (i.e., part numbers), and

a

given Part may show up in many Requisitions? If so, you need three

tables:

Requisition, RequisitionDetail, Part.

I also have a Purchase Order Table and a Purchase Order Detail Table.

Each

Purchase Order will contain many part numbers and many of the same

part

numbers will be on multiple Purchase Orders. (The Purchase Order Table
contains a Line Item Number, that, when combined with the Purchase

Order

Number creates a unique number for that specific part instance.)

I didn't get that last explanation, but this sounds analogous to the previous ... 1 − m − 1 (three tables: PurchaseOrder,

PurchaseOrderDetail,

Part)

I also have a Parts Table and a Part Details Table. This was created

at

the

beginning after acceptance of the proposal. It contains all the part

numbers

and their details and is used as a baseline so we can track changes in
pricing and quantities in conjunction with the Purchase Orders and
Reqisitions.

I don't get this. What kind of "part detail" information are you

saving?

The process is to create a reqisition first, followed by creating a

Purchase

Order second. Although, sometimes a Requisition is not created and

there

will

only be a Purchase Order.

So, a given Requisition can have how many PurchaseOrders? None (it

sounds

like), one, many? This will determine the relationship between these

two.

> Question One: Do I really have a many to many relationship between the
> Requisition and Parts Tables and a many to many relationship between

the

> Purchase Order Table and the Parts Table?

It sure sounds like it.

> Question Two: How do I connect/keep track of each Part Number in

conjunction

> with its respective Requisition Number and Purchase Order Number?

If I understand your data/relationship, you could use a query that joins

the

tables together on their respective IDs, then selects for PartNumber.

––
Regards

Jeff Boyce
Microsoft Office/Access MVP
http://mvp.support.microsoft.com/

Microsoft IT Academy Program Mentor
http://microsoftitacademy.com/

Microsoft Registered Partner
https://partner.microsoft.com/

.