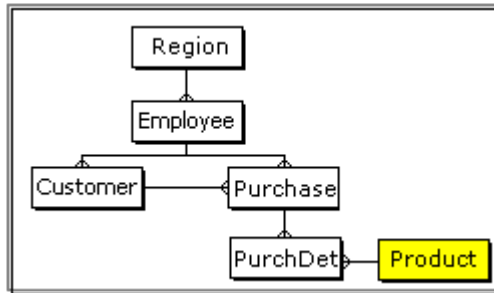


Work Sets in the Sample Database

Decisions about work sets are governed by what you want your application to do. The selection of base tables depends entirely upon your own business rules and not on PeerDirect.

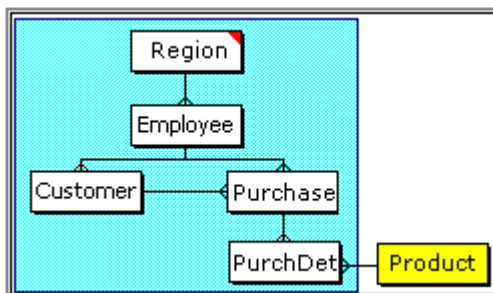
Using the PeerDirect [sample database](#) as an example, the tables that every site should always have the latest copy of are the *Product* and *Region* tables. These will be [global](#) tables. The *Customer* table, for example should not be global, since if you are located in the region of "Eastern United States", you do not need to see the customers in "South America".

PeerDirect Sample Database Schema:



Conversely, a [local](#) table is one that is never replicated. Like a global table, a local table is not part of any work set. In our sample database, there are no local tables.

We are now left with tables that will be included in work sets. Essentially a work set is a collection of related tables that you wish to replicate together. If you [subscribe](#) to a work set slice, you receive all of the data from the various tables in the work set (excluding data in nested work sets which must be subscribed to separately). For example, let's make the *Region* table a base table, and all other tables part of that work set (dependent tables). If there are two entries in the *Region* table, 'Canada' and 'United States', each site would have the choice of subscribing to those [slices](#). If a user in Toronto subscribes only to the region 'Canada', then when they replicate with another site, they will only receive updates, inserts or deletions from employees, customers, purchases and purchase details that are part of the 'Canada' region. If a New York site creates a new employee and associates them with the region of 'United States', then the Toronto site would not receive that information.



This seems to make sense, so we will make the *Region* table a [base table](#). Are there any other tables which would logically comprise a work set? What about the *Purchase* table? Yes, and no. For example, if you are subscribed to the 'Canada' region, you will receive all purchases made by customers in 'Canada'. From there, you can subscribe or unsubscribe purchases you don't want to see. This is a bit too much like micro-management, so not the best choice for a base table.

What about *Customer* or *Employee* as a base table? This is something that depends on what you want from your application. In our scenario, assume that employees can take over the customer accounts of other employees. Therefore, if the *Employee* table is made into a base table, then you would have to subscribe to a specific employee to see their customers. In our case, it makes more sense to make *Customer* a base table. In fact, we make *Customer* a nested base table under *Region*. This means that you can subscribe to any *Customers* in the *Region* slices to which you are subscribed. For example, if you are subscribed to the 'United States' region, then you can subscribe to any *Customers* that belong in that region.

Here is the final structure regarding work sets:

Region work set:

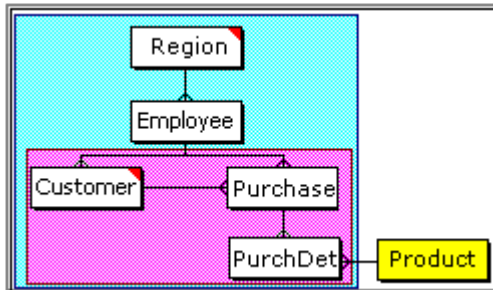
Region is the base table, Employee is a dependent table

Customer work set (Nested):

Customer is the base table, Purchase and PurchDet are dependent tables

Product:

global table



How does this breakdown in terms of data?

Region table:

This is a base table. It is unique such that every site receives all of the records. Because a site needs to know if it wants to subscribe to records in this table, each site must see all the records. Therefore, every site will have the complete contents of this table, and be able to subscribe to each record. Note that a *compact site* is one that stores only base table records to which it has been subscribed, rather than all records.

Employee table:

This is a nested table in the Region work set. Each site will receive only the Employee records that are related to subscribed records in the Region table (by foreign key relationships).

Customer table:

This is known as a nested base table. As with a region, you can subscribe to a Customer. However, unlike the Region table, you do not see all customers. You will only be able to subscribe to a customer in the regions to which you are subscribed. That is, if you are subscribed to the "Europe" region, you will have the choice of subscribing to any customers in that region only. Note again that a *compact site* is one that stores only base table records to which it has been subscribed, rather than all records.

Purchase table:

This is a dependent table in the Customer work set. You will see all Purchase information for the Customers to which you are subscribed. For example, if you are subscribed to the "Europe" slice, and the customer "PeerDirect UK", then you will see Purchase data from that customer. When you replicate, your site will exchange "PeerDirect UK" purchase data with the other sites that have subscribed to the same slice.

Purchase Detail table:

This is a nested table in the Customer work set. Similar situation to the Purchase table.

Product table:

This is a global table. Every site will contain all information, depending on the frequency of replication. It is not in any work set. If site A inserts a record the record will be replicated to all other sites.

Nested work sets

If a user of this application is interested in a particular region, they are likely to need customer information for that region. Since Customer is itself a work set, this is called a nested work set. PeerDirect uses nested work sets to solve this problem. If the Customer work set is made a nested work set under Region, then all of the tables in the Customer work set are automatically replicated along with the Region tables.

Replication of the Database

With the work sets defined for the sample application, an employee in the Boston region can subscribe to specific

customers in which he is interested. The information about these customers (a Slice) will then be replicated to his workstation. He has now received "all and only" the customer information that he wants.

Initially the user will have all of the regions replicated to his workstation. Once he subscribes to a specific region he will receive all of the employees in that region. Since Customer is a nested work set of Region however, he will also receive all customer information for the region. Once he subscribes to a specific customer, he receives that customer's purchases and purchase details.

If you unsubscribe from a customer, all of the related purchases and purchase details for that customer will be removed from the local database.

Support for 'many-to-many' database table relationships

The examples used on this page all involve work sets that include 'one-to-many' database table relationships. PDRE also supports the more complex situation of replication between work sets with 'many-to-many' database table relationships. Please refer to [Work sets and many-to-many relationships](#) for more information.