COMPUTATIONAL ECONOMICS

Databases in Access

David A. Kendrick
Ruben P. Mercado
Hans M. Amman

1 Introduction

Database systems have had a very substantial impact on the way that both businesses and government agencies manage production, sales, inventory, and personnel. Curiously though, they have as yet had relatively little effect on the way economists develop and maintain the data that are used to measure the pulse of economic activity in both macro- and microeconomic settings. It seems likely that this will change as a new generation of economists who have cut their teeth on Mac's and PC's arrive on the scene.

This chapter provides an introduction to the use of relational database systems using the Access software. An example database developed by Kendrick (1982b) in Access is used to illustrate the potential for relational database systems in economics.

At present most economic data are organized as sets of unrelated time series that are maintained by different agencies. Thus to find the consumption data for the U.S. economy one might go Bill Goffe's *Resources for Economists* page on the Internet [see Goffe (2004)] and from there track down the macroeconomic databases and pull out the consumption time series. From an econometrician's point of view this may be a very serviceable system: To estimate a consumption function the user might download the time series for consumption, income, taxes, and interest rates into a spreadsheet such as Excel or an estimation package. Then disposable income would be calculated from the income and taxes series. Finally consumption would be regressed on disposable income and interest rates.

However, there are many other uses of economic data apart from as input to regression packages. Frequently the user does not want to run a regression, but rather wants to address a query that depends upon the relationships among the data. If the data are stored and organized not as a set of unrelated time series but as a relational database, such queries can be answered quickly and easily. Moreover, once the data are organized in relational forms it would also serve the econometricians very well by permitting easy control over aggregation and disaggregation and the development of samples for use in estimation packages.

This chapter begins with an introduction to the terminology of database systems. The sample database for the U.S. economy is outlined along with the specification of these data in the Access software. Then the procedure for developing and using queries of the database is discussed.

2 Domains, Relationships, and Joins

The relationship is the key concept in database methodology. Yet it is as simple as a table. For example, consider a table that shows the locations and production levels of

Table 1: Production Relationship

plant	city	commodity	output
Inland	Gary	Steel	4
ARCO	Houston	Oil	73
ALCOA	Rockdale	Aluminum	125

Table 2: City Location Relationship

city	state
Gary	Indiana
Houston	Texas
Rockdale	Texas

a set of plants as in Table 5.1: the Inland Steel plant at Gary, StateIndiana, produced 4 million tons of steel, the ARCO refinery at Houston, StateTexas, processed 73,000 barrels of oil, and the ALCOA aluminum smelter at Rockdale, StateTexas, produced 125,000 tons of aluminum. In the language of database systems this table would be called a relationship of the form

@eq:Production (plant, city, commodity, output)

@normal:and the domains of the relationship are the sets of plants, cities, commodities, and output levels used in the database.

The foregoing production relationship has three elements and each element is a four-tuple, that is,

```
Production = { (Inland, Gary, Steel, 4)
             (ARCO, Houston, Oil, 73)
             (ALCOA, Rockdale, Aluminum, 125) }
```

So, in summary, the Production relationship is a set with elements, each of which is a tuple.

Another key concept in relationship databases is that of the *join*. In order to illustrate a join we introduce two more relationships, City Location and State Location as shown in Tables 5.2 and 5.3.

The City Location and State Location relationships have a common *domain*, that is, state. So one can join these two relationships to create a new relationship that shows

Table 3: State Location Relationship

state	region
Indiana	Mid-West
Texas	Gulf Coast

Table 4: City Location Relationship

city	region
Gary	Mid-West
Houston	Gulf Coast
Rockdale	Gulf Coast

the region of each city, as shown in Table 2

Furthermore, one can then do an additional join of the Production and the City Location relationships using the common city domain to obtain a Regional Production relationship, shown in Table 2, which could then be printed without the plant domain to produce the desired result, as shown in Table 2

All of this may seem like a lot of work to obtain a simple table. However, note that the State Location relationship can be modified independently of the others. Thus an economist would be free to create his or her own regional aggregation scheme and develop queries based on that scheme.

3 An Example Database

An example database for the U.S. economy from Kendrick (1982b) is provided in full in Appendix A at the end of this chapter. The Access file is available on the book web site.

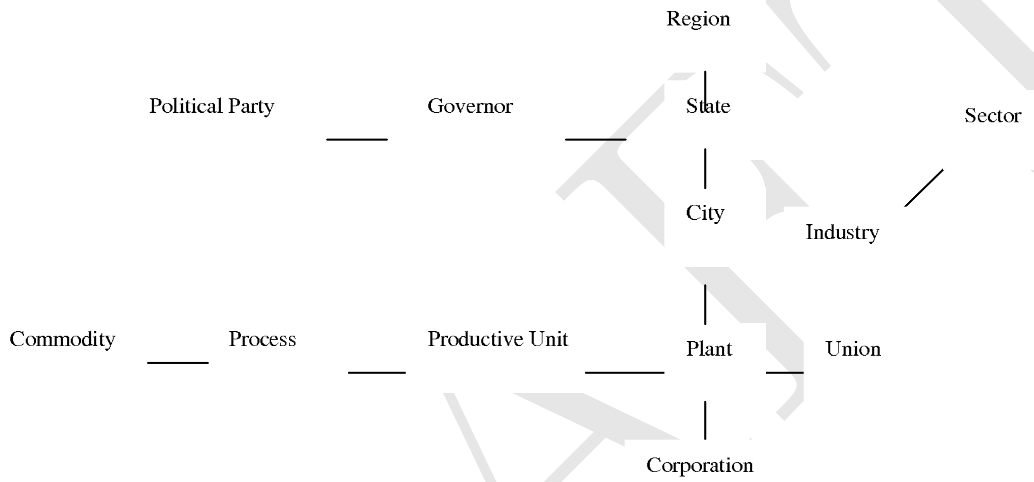
Table 5: Regional Production Relationship

region	commodity	plant	output
Mid-West	Steel	Inland	4
Gulf Coast	Oil	ARCO	73
Gulf Coast	Aluminum	ALCOA	125

Table 6: Regional Production

region	commodity	output
Mid-West	Steel	4
Gulf Coast	Oil	73
Gulf Coast	Aluminum	125

Figure 1: Links between the domains in the example database



That database was created as a simple illustration of how a relational database might be constructed with data from the U.S. economy. The purpose was not to be comprehensive, but rather to illustrate how one might fruitfully link production, ownership, labor relations, location, and even politics together in a single database.

The U.S example database is a set of fourteen relationships that link commodities, productive units, plants, unions, companies, industries, sectors, cities, states, regions, governors, and political parties in the fashion outlined in Figure 5.1

The story that can be told about how these domains are linked is as follows:

plants contain *productive units* in which *processes* are used to produce *commodities* *plants* belong to *corporations* *plants* have workers who belong to *unions* *plants* belong to *industries* that belong to *sectors* *plants* are located in *cities* that are in *states* that are in *regions* *states* have *governors* who belong to *political parties*

The same story can be told using the relationships instead of the domains. The inputs to and outputs from production processes are described by

```
Input-Output (commodity, process, input-output coefficient)
```

and the level of production of each commodity by each process is given by

`Production (commodity, process, year, production level)`

The productive units that are used by each process are indicated in

`Capacity Use (process, productive unit, capacity coefficient)`

and the capacity of those productive units in each plant are shown in

`Capacity (productive unit, plant, year, capacity level)`

The increase in this capacity in a given year at each plant is displayed in

`Increment to Capacity (productive unit, plant, year, incremental capacity)`

The plants are owned by corporations:

`Ownership (plant, corporation)`

The plants also have employees who belong to unions:

`Plant Employees (plant, union, number of employees)`

Moreover the plants belong to industries:

`Industry Composition (plant, industry)`

and the industries to sectors in the economy:

`Sector Composition (industry, sector)`

The plants are located in cities:

`Plant Location (plant, city)`

which are located in states:

`City Location (city, state)`

which are in turn located in regions of the country:

`State Location (state, region)`

The states have governors:

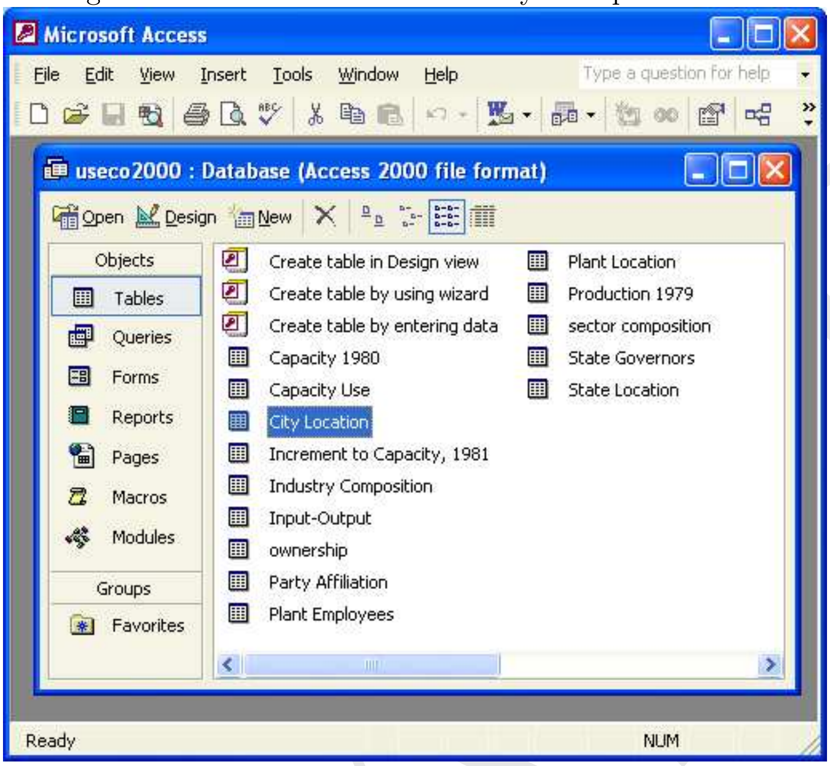
`State Governors (state, governor)`

who are affiliated with political parties:

`Party Affiliation (governor, party)`

One would not be able to tell such a simple story for a full and comprehensive database of the U.S. economy, but this simple story and small database serve our purpose of introducing the use of relational databases in economics.

Figure 2: Tables in the U.S. economy example database



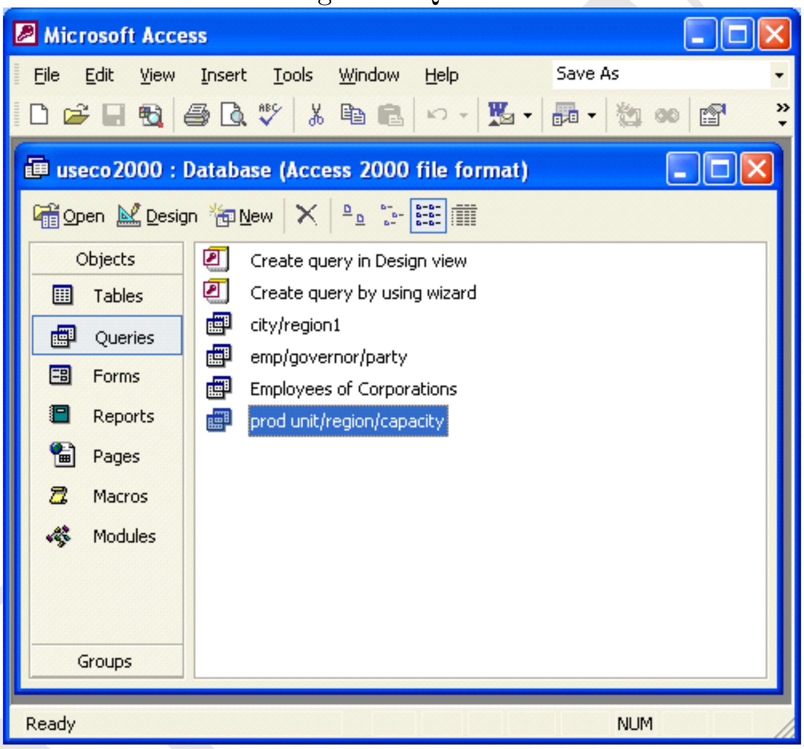
4 Representation of the Example Database in Access

We turn now to the way that this relational database is represented and used in Access. The first window that you see when you open it contains a list of the tables (relationships) that make up the database as is shown in Figure 5.2, which shows the fourteen relationships discussed earlier. Unfortunately they are arranged alphabetically rather than in the conceptual order used in that discussion. You can examine the tables one by one by double clicking on them and then comparing them to the corresponding relationship in Appendix A. In particular take a look at the Ownership and the Plant Employees tables since we use both of them in the explanation of the query that follows.

The principal use of databases is to answer queries, that is, questions about the data in the database. First we take a look at a couple of existing queries and how they are specified. Select the Queries option in the objects bar on the useco2000 : Database window and you see the display shown in Figure 5.3.

This window shows that four queries have already been developed. After you see how they work you are in a position to develop queries of your own. Consider first the Employees of Corporations query. Select it but be careful to single, rather than double, click on it . This query answers the question of how many employees of each corporation are members of each union. One cannot answer this question directly by looking at the individual tables; however, the question can be answered by combining the information

Figure 3: Queries



in the two relationships:

```
Ownership (plant, corporation)
Plant Employees (plant, union, number of employees)
```

The first tells us which corporation owns each plant and the second tells us how many employees in each plant belong to each union. Thus if we combine the two we have a new relationship that we call Employees of Corporations, which has the domains

```
Employees of Corporations (union, corporation, number of employees)
```

This is a join of the type we discussed at the beginning of the chapter since we are joining two relationships together by using the common domain *plant*.

Look back to Figure 5.3 and note that there is a Design button in the toolbar at the top of the Queries window. Click that button and a window opens that shows the design of the query as in Figure 5.4. The top part of this window includes the two relationships that are used in the query, namely Plant Employees and Ownership. The short zigzag line that connects the plant domain in the two relationships indicates that the join is to be performed over this domain. You can move the Plant Employees and the Ownership table around by clicking on the label at the top and dragging the table, which is a capability that comes in handy when you begin designing your own queries.

The bottom half of the window in Figure 5.4 contains a table that lists the domains in the new relationship created by the query. There are also check marks that allow you to suppress the display of any of the domains. We need all of them for our query so leave all the check marks for the moment and close the query design window by clicking on the $\times$ in the upper right hand corner.

Now you are back at the Query window. Be sure that the Employees of Corporations query is still selected and then click on the Open button in the toolbar at the top of the dialog box. The window shown in Figure 5.5 appears.

Note here that the answer to a query is itself a relationship. You can quickly see from the table that U. S. Steel has employees who belong to the machinists (IAM), teamsters (IBT), and steelworkers (USA) unions. Note that there are also two lines in the table that are almost identical:

```
USA  ALCOA  0.5

USA  ALCOA  0.7
```

This happens because ALCOA owns two plants in the database, namely Rockdale and Point Comfort, and members of the United Steel Workers of America (USA) are employed at both plants. From this you can see that it is sometimes necessary to aggregate after a query is run before you have the answer in exactly the form you want.

Figure 4: Design of the employees of corporations query

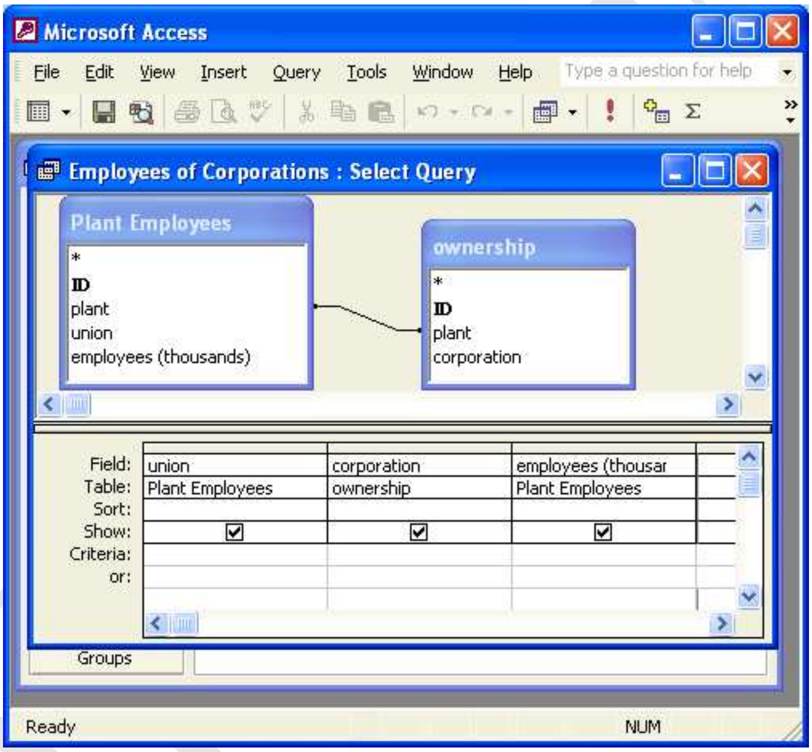
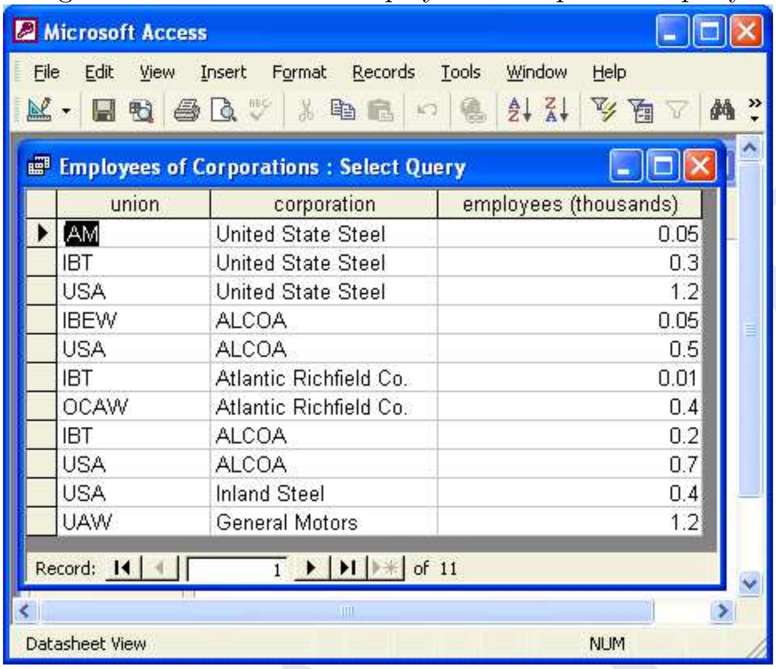


Figure 5: Answer to the employees of corporation query



union	corporation	employees (thousands)
AM	United State Steel	0.05
IBT	United State Steel	0.3
USA	United State Steel	1.2
IBEW	ALCOA	0.05
USA	ALCOA	0.5
IBT	Atlantic Richfield Co.	0.01
OCAW	Atlantic Richfield Co.	0.4
IBT	ALCOA	0.2
USA	ALCOA	0.7
USA	Inland Steel	0.4
UAW	General Motors	1.2

Next try designing a query of your own. We use one of the queries that is already available in the database so that you can see how it should come out in case yours does not work out as it should. We take the simplest case. There are relationships for the location of cities by state and for the location of states by region so use these two to create a relationship that shows the locations of cities by region. Thus we use the two relationships;

```
City Location (city, state)
State Location (state, region)
```

to create a new relationship that we call

```
City/Region2 (city, region)
```

The name City/Region2 is used to distinguish the query from the one already in the database called City/Region1. Here you see the principles of designing your own query. Begin by clicking on the Query object in the objects bar and without selecting any of the existing queries click on the New button in the toolbar at the top of the window. When you do this the dialog box shown in Figure 5.6 appears.

Though you may find it useful to use one of the Wizards later let us do it by hand here. So select the Design View option and click on OK. The Show Table dialog box

Figure 6: New query dialog box

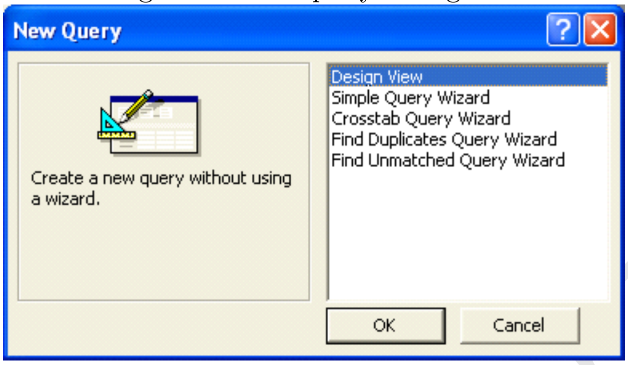
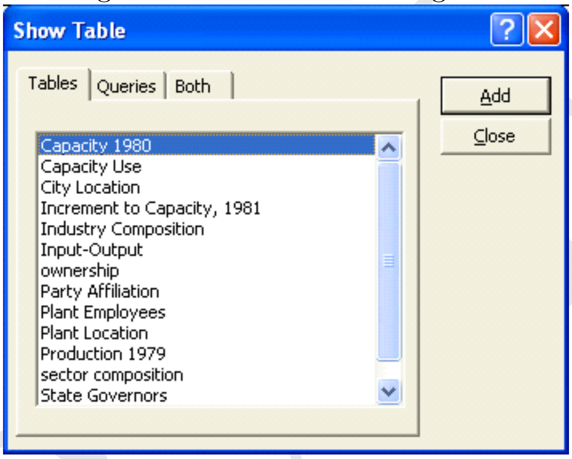


Figure 7: Shown Table dialog box



appears, as in Figure 5.7.

One of the tables we want to use in the query is City Location, so click on it and then click on the Add push button in the upper-right-hand corner of the dialog box. The City Location table appears in the top half of the Query Design dialog box. Select the State Location table from the Show Table dialog box (scroll down to find it if necessary), click on the Add push button in the upper-right-hand corner of the box, and then close it. Once you have done this the window should look something like the one shown in Figure 5.8, with the City Location and State Location relationships displayed in the top half of the Query Design window.

Before we go further change the name of the query from the default to the choice discussed previously: City/Region2. To do this select the File menu and the Save option. A small dialog box appears that allows you to rename the query. After you have done so and closed that small dialog box, you are back at Figure 5.8.

The next step is to establish the join. In Figure 5.8 there is already a join between

Figure 8: Designing a query

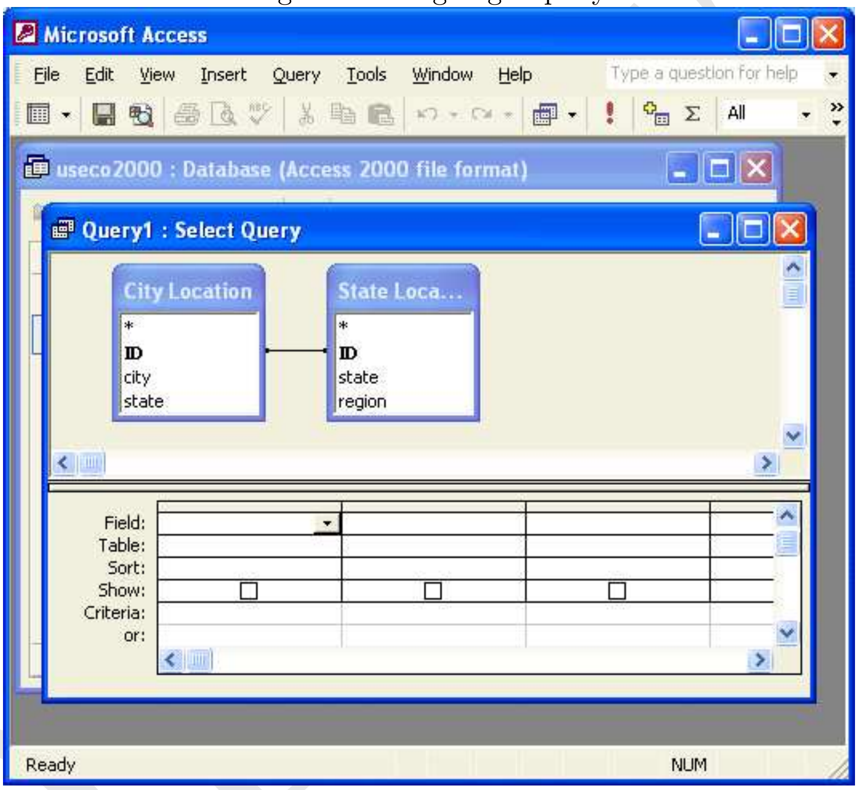
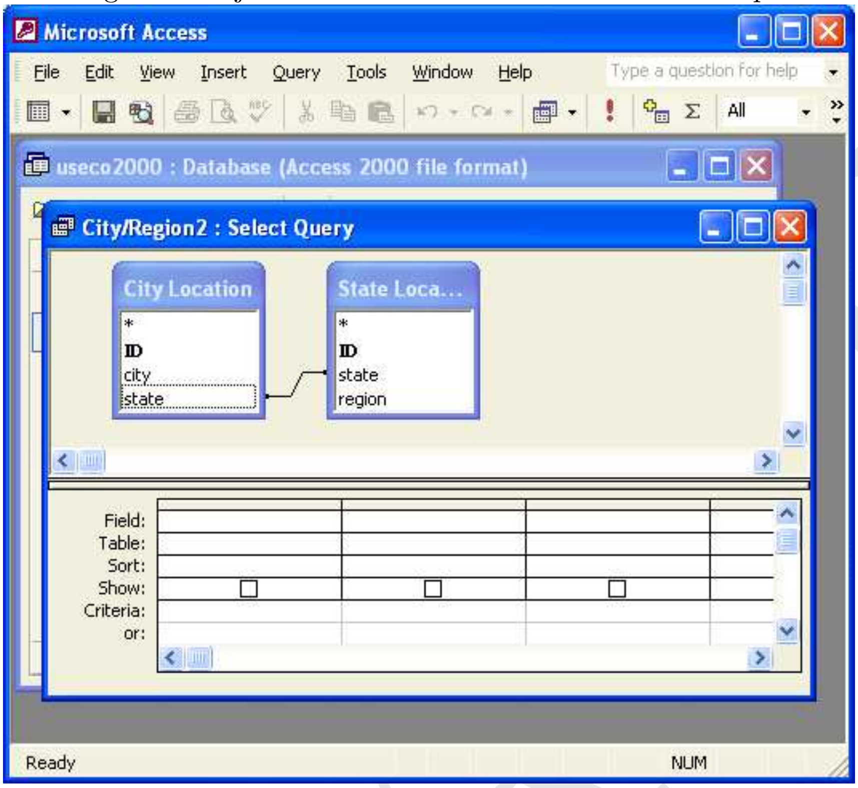


Figure 9: A join between State in the two relationships



the IDs in the two tables but that is not what we want. Click on the line that connects the two relationships and strike the Delete key so that the line disappears. You may have trouble with this at first, but keep trying until the line becomes slightly darker when you click on it to indicate that it has been selected. Then you should be able to delete it by striking the Delete key.

Next create a join between the *state* domain in the two relationships by clicking on state in one of the tables and dragging to state in the other table. Once you have done this, the window should appear as shown in Figure 9. The zigzag line between the state domains in the two relationships indicates that a join has been established. This completes our work on the top part of the design window and we can now turn our attention to the bottom part.

Begin by clicking in the first column in the Field row. When you do so the cursor appears there along with a small arrow in the right-hand side of the box. Click on this box to cause a drop-down window to appear, as shown in Figure 5.10.
bb

Since we want the city domain in this first field select the line City Location.city and the domain city appears in the box. Repeat this process for the Field row in the next

Figure 10: Drop-down window for filling in the field

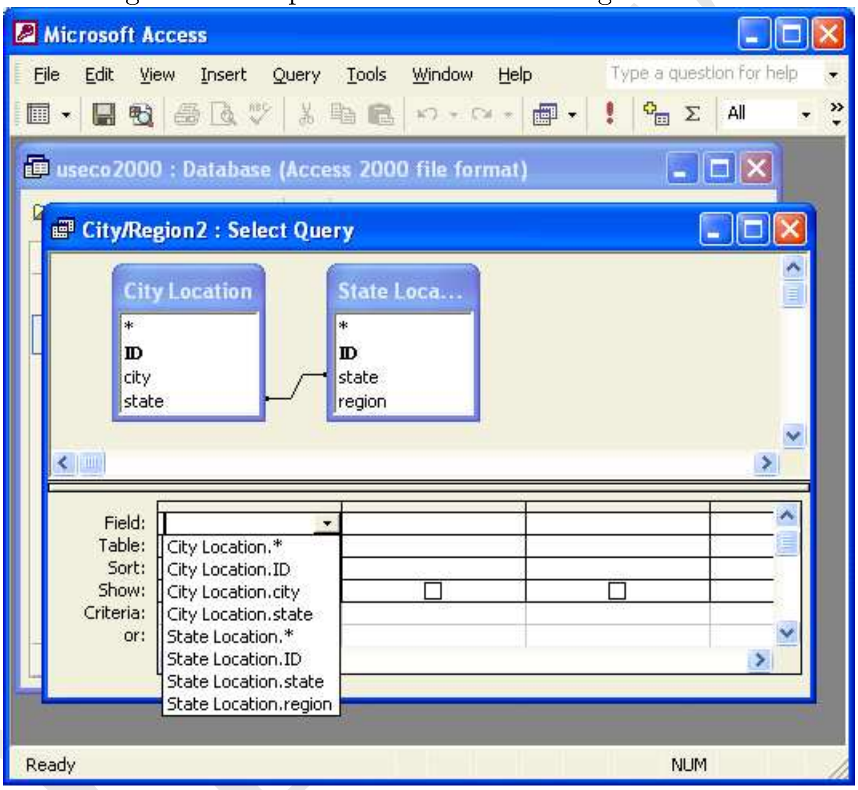
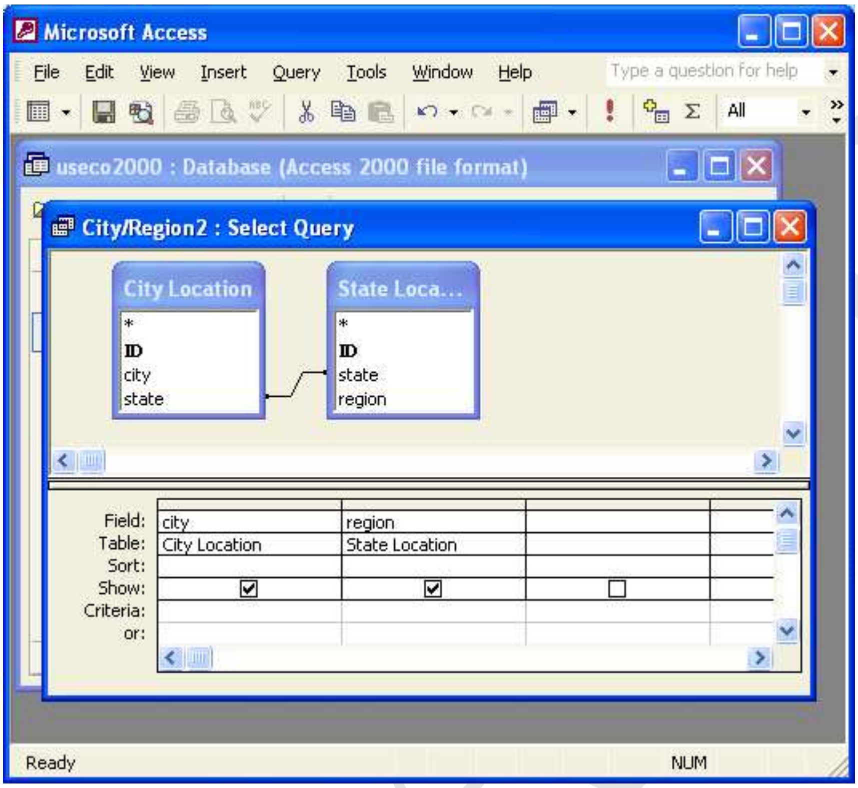


Figure 11: Complete query design



column but this time select State Location.region from the drop-down window. Once you have done this the Query Design window should appear as shown in Figure 5.11. The query design process is now completed. This undoubtedly seems like a long and complicated process but it goes very quickly once you have the hang of it.

Now we are ready to make use of the query that we have constructed. To do this close the Query Design window by clicking on the × in the upper-right-hand corner. When you do this the Query dialog box should appear as shown in Figure 5.12. Now the Query dialog box contains the new query that we have created, City/Region2. As mentioned earlier there was already a city/region1 query that performs the same function for comparison if you have had difficulties in some of the foregoing steps. However, for now ignore this and try the one we have created by clicking on City/Region2 and then on the Open push box in the upper-left-hand corner of the window. When you do so the result of the query should appear as shown in Figure 5.13, which gives the desired result—a table that shows the region in which each city is located.

As we noted earlier, this seems like too much work just to find out that Houston

Figure 12: Query dialog box showing the new query City-Region2

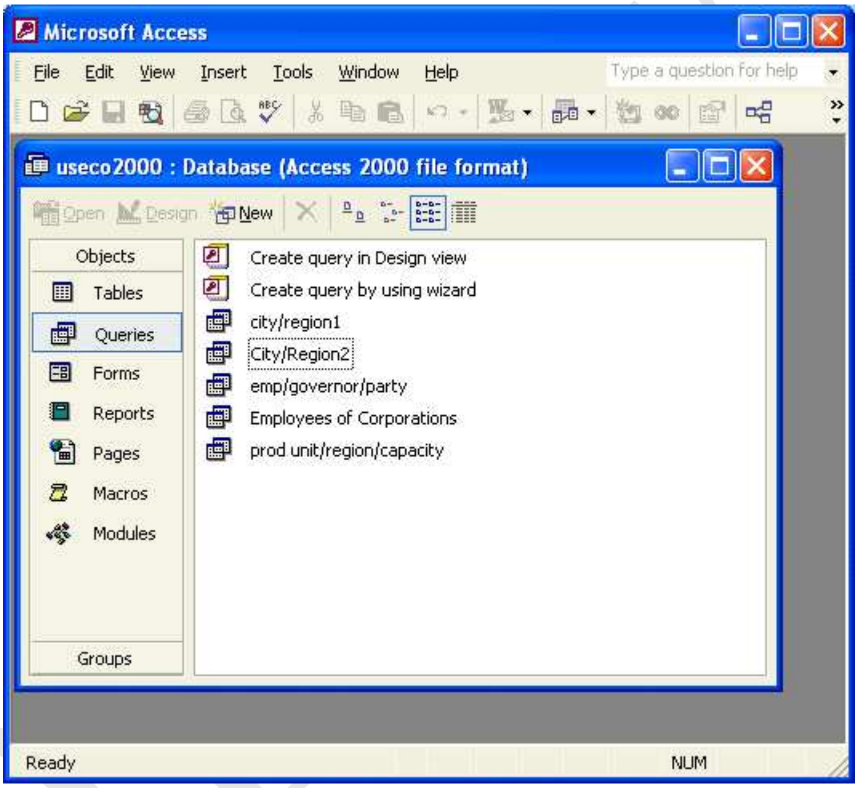
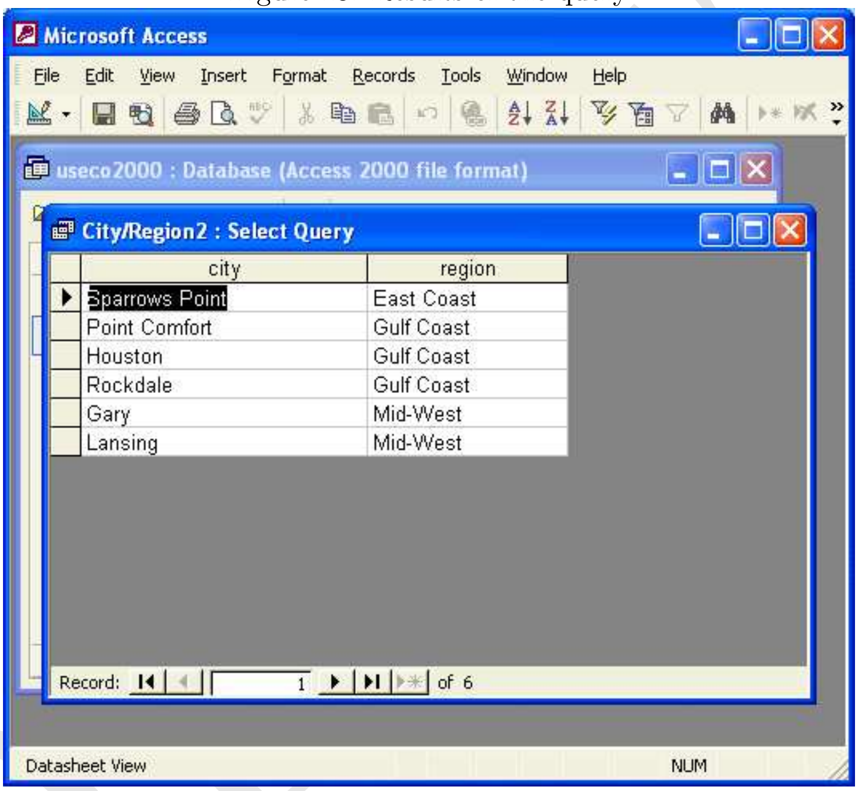


Figure 13: Results of the query



is in the Gulf Coast region. However, once you have gained some facility with Access the point and click nature of the interface makes it an efficient way to develop queries. Moreover, when you have a large database with many relationships and many data the power of the methods becomes apparent.

5 Examples

There are a number of examples of the use of this database given in Kendrick (1982b). Here it suffices to describe a couple of them.

The first example is from the field of energy economics. Imagine that as the hurricane season approaches emergency management officials want to know the amount of refining capacity on the placeAtlantic and placeGulf of Mexico coasts. This roughly corresponds to the East Coast and Gulf Coast regions in the database so we clearly need to begin from the relationship

```
State Location (state, region)
```

and from there we work backward with the

```
City Location (city, state)
```

relationship to the

```
Plant Location (plant, city)
```

relationship. Then we cap this off by using the

```
Capacity (productive unit, plant, year, capacity level)
```

relationship. If we do a join from plant to plant in the last two of these relationships and from city to city just above and then from state to state in the previous pair we obtain a new relationship that we call

```
Regional Capacity (productive unit, region, year, capacity level)
```

that contains the information shown in Table 5.7, from which we can quickly see that that the primary still capacity (which is the best indicator of the refinery capacity) is 0.2 million barrels a day in the Gulf Coast region.

You can look at the implementation of this query in the database by going to the Query tab and then selecting the prod unit/region/capacity query. In particular, it is useful to single click this query and then click on the Design button so that you can see how the four relationships are used in the query and how they are linked by plant, city, and state to create the desired relationship.

Table 7: Regional Capacity Relationship

Productive unit		Region		
	East Coast	Gulf Coast	Mid-West	Units
Blast furnace	2.0		2.5	mty
Steel shop	2.35		2.8	mty
Rolling mill	1.9		2.4	mty
Alumina plant		0.8		mty
Aluminum plant		1.1		mty
Primary still		0.2		mbd
Catalytic cracker		0.23		mbd
Auto stamping			0.6	muy
Auto assembly line			0.6	muy

A second example comes from politics. From time to time presidential politics in the United States are affected by difficulties in a particular industry. One example is the pressure that the U.S. auto industry has felt from imports at times. Suppose that in a campaign a presidential candidate asks for a list of Democratic governors whose states have more than 10,000 people employed in the automobile and steel industries. Clearly for this query we need to work back from

```
Party Affiliation (governor, party)
```

to

```
State Governors (governor, state)
```

to

```
City Location (city, state)
```

to

```
Plant Location (plant, city)
```

We then make use of the

```
Plant Employees (plant, union, number of employees)
```

and the

`Industry Composition (plant, industry)`

relationships.

If we do all the required joins properly we should obtain a relationship that we call

`Employees by Governor and Party (industry, governor, party, number of employees)`

From this table we can assemble the data required to answer the query.

These two examples provide an indication of how a relational database of the economy might be used to provide quick answers to a wide variety of questions. In most cases the answers can be provided in tables. In other cases the results of the queries are time series or cross sections of data that would then subjected to further econometric analysis.

6 Experiments

A beginning experiment might be to implement the last of the two foregoing examples; however, it is probably wise to begin with something simpler such as the location of cities by region as described earlier but to do it without following the steps in the book.

Once you gain some confidence in doing simple joins you are encouraged to develop you own queries. If the existing relationships in the database are not sufficient, you may want to add some others in order to be able to answer richer and more interesting queries.

Finally, you might want to develop your own database with data from financial markets, labor relations, or environmental economics as suits your interests. If you have had a summer job or an internship in a business or governmental agency you have likely made use of some databases and might want to try your hand at developing a similar database in Access or some other relational database software.

7 Further Reading

The classic book on relational database systems is Date (1977). To learn more about Access 2000 see Andersen (n.d.).

References

Andersen, V.: n.d., *Access 2000: The Complete Reference*, The McGraw-Hill Companies, New York, USA.

Date, C. J.: 1977, *An Introduction to Database Systems*, 2nd edn, Addison-Wesley, Reading, Massachusetts.

Appendices

A An Example U.S. Economy Database

This appendix contains the relationships in the example U.S. economy database from Kendrick (1982b).

$$\textit{test} \tag{A-1}$$

Table 1: Input-Output

commodity		process		
	Pig iron production	Steel production pig iron intensive	Steel production scrap intensive	Rolling flat steel products
Iron ore	–1.6			
Pig iron	1.0	–0.9	–0.7	
Scrap iron		–0.2	–0.4	
Liquid steel		1.0	1.0	–1.2
Scrap				0.2
Flat steel				1.0

Table 2: Regional Production

	Alumina production	Aluminum production	Primary distillation	Catalytic cracking
Bauxite	–1.4			
Alumina	1.0	–1.2		
Aluminum		1.0		
Crude oil			–1.0	
Distillate			0.2	–1.0
Gasoline			0.3	0.6
Jet fuel			0.1	0.2

In the Input-Output relationship negative values indicate inputs and positive values outputs. Thus in the pig iron production process 1.6 tons of iron ore are used to produce

Table 3: Regional Production

	Auto body stamping	Auto assembly
Flat steel	–1.2	
Aluminum	–0.2	
Auto bodies	1.0	–1.0
Automobiles		1.0

Table 4: Production

commodity		process		
	Pig iron production	Steel production pig iron intensive	Steel production scrap intensive	Rolling flat steel products
Pig iron (mty)	86.8			
Liquid steel (mty)		55.5	53.0	
Scrap (mty)				18.0
Flat steel (mty)				90.0

1.0 tons of pig iron. In the next column 0.9 ton of that pig iron is used along with 0.2 ton of scrap to produce 1 ton of liquid steel. The activity analysis vectors here follow in the tradition of Tjalling Koopmans (1951). Also for an introduction to use of activity analysis in economics see Kendrick (1996).

Thus a process is akin to a cook's recipe in that it provides a list of ingredients and how much is required of each as well as an indication of the final product or products. However, unlike the usual recipe for a cake, a process may have a single input and multiple outputs as, for example, the foregoing process for primary distillation in an oil refinery where crude oil input is transformed into distillate, gasoline, and jet fuel.Kendrick/Ch 5 35 06/29/08 8:57 PM

Note: mty = million tons per year; tby = trillion barrels per year; muy = million units per year.

Table 5: Regional Production

	Alumina production	Aluminum production	Primary distillation	Catalytic cracking
Alumina (mty)	20.0			
Aluminum (mty)		16.0		
Distillate (tby)			1.46	
Gasoline (tby)			2.19	2.43
Jet fuel (tby)			0.73	0.73

Table 6: Regional Production

	Auto body stamping	Auto assembly
Auto bodies (muy)	9.5	
Automobiles (muy)		9.35

@T:Table 5A.3 Capacity Use

The Capacity Use relationship simply indicates the productive unit in which each process runs. Note that substitute processes such as the two for steel production both use the same productive unit, namely the steel shop.

Note: mty = million tons per year; mbd = million barrels per day; muy = million units per year.

Table 5A.7 is really about investment, but it differs from the usual notion of investment, which is a certain number of dollars spent on a new plant or equipment. Rather the investment in the table is defined as an increment to capacity and is measured in units of the principal input or output of the productive unit. Thus the blast furnace capacity at Sparrows Point is increased by 0.5 million tons per year (an output) and the primary still at ARCO-Houston is increased by 0.1 million barrels per day (an input).

Note: OCAW, Oil, Chemical and Atomic Workers; UAW, United Auto Workers; USA, United Steel Workers of America; IBEW, International Brotherhood of Electrical Workers; IBT, International Brotherhood of Teamsters; IAM, International Association of Machinists.

Which relationships share a common domain?

Plant Location and City Location are linked by *city*

City Location and State Location and

State Governor are linked by *state*

State Governor and Party Affiliation are linked by *governor*

Table 7: Regional Production

productive unit		process		
	Pig iron production	Steel production pig iron intensive	Steel production scrap intensive	Rolling flat steel products
Blast furnace	1			
Steel shop		1	1	
Rolling mill				1

Table 8: Regional Production

	Alumina production	Aluminum production	Primary distillation	Catalytic cracking
Alumina plant	1			
Aluminum plant		1		
Primary still			1	
Catalytic cracker				1

Table 9: Regional Production

	Auto body stamping	Auto assembly
Auto stamping plant	1	
Auto assembly plant		1

Table 10: Industry Composition

plant	industry
Sparrows Point	Steel
Rockdale	Aluminum
ARCO-Houston	Oil
Point Comfort	Aluminum
Inland-Gary	Steel
Lansing	Automobile

Table 11: Sector Composition

industry	sector
Steel	Primary metal
Aluminum	Primary metal
Oil	Petroleum and coal
Automobile	Transportation equipment

Table 12: Capacity 1980

productive unit	plant	capacity level	units
Blast furnace	Sparrows Point	2.0	mty
Blast furnace	Inland-Gary	2.5	mty
Steel shop	Sparrows Point	2.35	mty
Steel shop	Inland-Gary	2.8	mty
Rolling mill	Sparrows Point	1.9	mty
Rolling mill	Inland-Gary	2.4	mty
Alumina plant	Point Comfort	0.8	mty
Aluminum plant	Point Comfort	0.6	mty
Aluminum plant	Rockdale	0.5	mty
Primary still	ARCO-Houston	0.2	mbd
Catalytic cracker	ARCO-Houston	0.23	mbd
Auto stamping plant	Lansing	0.6	muy
Auto assembly line	Lansing	0.6	muy

Table 13: Increment to Capacity 1981

productive unit	plant	increment to capacity	units
Alumina plant	Point Comfort	0.5	mty
Aluminum plant	Point Comfort	0.4	mty
Auto assembly line	Lansing	0.0	muy
Auto stamping plant	Lansing	0.0	muy
Blast furnace	Sparrows Point	0.5	mty
Blast furnace	Inland-Gary	0.0	mty
Catalytic cracker	ARCO-Houston	0.12	mbd
Primary still	ARCO-Houston	0.1	mbd
Rolling mill	Sparrows Point	0.4	mty
Steel shop	Sparrows Point	0.5	mty
Aluminum plant	Rockdale	0.0	mty
Steel shop	Inland-Gary	0.0	mty
Rolling mill	Inland-Gary	0.0	mty

Table 14: Ownership

plant	corporation
Sparrows Point	United States Steel
Rockdale	ALCOA
ARCO-Houston	Atlantic Richfield Co.
Point Comfort	ALCOA
Inland-Gary	Inland Steel
Lansing	General Motors

Table 15: Plant Employees (in thousands of employees

plant			union			
	OCAW	UAW	USA	IBEW	IBT	IAM
Sparrows Point			1.2		0.3	0.05
Rockdale			0.5	0.05		
ARCO-Houston	0.4				0.01	
Point Comfort			0.7		0.2	
Inland-Gary			0.4			
Lansing		1.2				

Table 16: Plant Location

plant	city
Sparrows Point	Sparrows Point
Rockdale	Rockdale
ARCO-Houston	Houston
Point Comfort	Point Comfort
Inland-Gary	Gary
Lansing	Lansing

Table 17: City Location

city	state
Sparrows Point	Maryland
Rockdale	Texas
Houston	Texas
Point Comfort	Texas
Gary	Indiana
Lansing	Michigan

Table 18: State Location

state	region
Maryland	East Coast
Texas	Gulf Coast
Indiana	Mid-West
Michigan	Mid-West

Table 19: State Governors

state	governor
Maryland	Harry Hughes
Texas	William P. Clements, Jr.
Indiana	Otis R. Bowen
Michigan	William G. Milliken

Table 20: Party Affiliation

governor	party
Harry Hughes	Democrat
William P. Clements, Jr.	Republican
Otis R. Bowen	Republican
William G. Milliken	Republican

Ownership and Plant are linked by *corporation*
Industry Composition and
Sector Composition are linked by *industry*
Input-Output and Production are linked by *process* and *commodity*
Production and Capacity Use are linked by *process*
Capacity Use and Capacity are linked by *productive unit*
Increment to Capacity and Capacity are linked by *productive unit* and *plant*
Capacity and Plant Employees are linked by *plant*
Increment to Capacity and
Plant Employees are linked by *plant*
Industry Composition and
Plant Employees are linked by *plant*