F I G XVIII. INTERNATIONAL CONGRESS

TORONTO, CANADA, 1986

CONNECTION BETWEEN LOCAL AND REGIONAL: ADDITIONAL "INTELLIGENCE" NEEDED

M. EGENHOFER / A. FRANK

ABSTRACT

Local and regional are two examples of different levels of aggregation. We show how the rules for deriving regional data from local data can be formally expressed. This permits a general treatment of aggregation. The method lends itself to implementation in a PROLOG like language /FRANK/.

STRUCTURED LOCAL AND REGIONAL DATA

Classification, generalization and aggregation are common tools for data structuring and allow to choose different levels of abstraction. In land information systems we typically find local and regional as two levels depending on each other. We can distinguish 4 types of information

- local information achieved from local data.
- regional infomation achieved from independent regional data.
- local information influenced by regional data.
- regional information deduced from local data.

Independent local and regional information can be accessed by simple queries. The regionally influenced local information is a straight forward derivation, as one regional property transmits its value to one or several local properties. More complex is the higher level information that is derived from the lower level data.

CONNECTING LOCAL AND REGIONAL

If we connect local with regional data using tools of data structuring, we must describe which local and regional items belong together. How to derive the regional aggregated data values from the multiple local values? The method depends on the form of the relation that exists between the properties of the different levels. If we project one single value from the local level to one property of the regional level, we can do it directly by assigning the local value to the regional property. This method, however, can not be applied in order to project (aggregate) several local values to a single regional property. For this we need additional operations.

Let us look at a simple example: Settlements are the local places where people live in. The regional expression county aggregates the settlements so that each county consists of one or more settlements. We express the facts that Orono and Bangor (two settlements) have a certain population and that both of them belong to the county Penobscot by the following rules:

```
*p (Orono, Population, 10000).                          (1)
*p (Bangor, Population, 50000).                         (2)
*p (Penobscot, ConsistsOf, Orono).                      (3)
*p (Penobscot, ConsistsOf, Bangor).                     (4)
```

The population of the county is a regional information. It is related to the local settlement population, namely by summing up their single numbers. We say:

```
*passesOnTo (ConsistsOf, Population, County_Population,
             BySumming).                                (5)
```

meaning if something "consists of" other items, and these items have the property "Population", then they pass on their values for the population to the property "CountyPopulation" using the function "sum". Thus the first parameter describes the relation which the deduction is used on, the second is the lower level attribute, the third the higher level attribute the values are to be deduced from. The last parameter defines the operation how to aggregate data in order to derive the higher level value. If we formulate the meta-information for stepping from local to regional in the "passesOnTo"-form, we can deduce the regional values.

```
*p (regional_entity, regional_attribute, result) if
       passesonto (relation, local_attribute,
                   regional_attribute, operation),
       p (regional_entity, relation, local_entity),
       p (local_entity, local_attribute, value),
       operation (value, result).                       (6)
```

Using the above general rules (5) and (6) we are now able to answer the query how many people live in a certain county

```
p (Penobscot, County_Population, x).                    (7)
```

and we would get the result x = 60000.

DIFFERENT OPERATIONS FOR AGGREGATION

Simple operations for aggregation include the mathematical operations such as summing and product. Other operations are the similar functions for computing the maximum, minimum, average or composites of these.
The largest settlement of a county is determined by

- using the population of the settlement
- selecting the maximum value of the population
- and keeping the associated name of the settlement.

We describe these operations in a single rule:

```
*passesOnTo (ConsistsOf, Population, LargestTownName,
             NameOfMaximum).                            (8)
```

where NameOfMaximum is a function which selects the highest value and keeps its name. Note that this operation differs from determining only the maximum population of a county.

More complex operations combine the former mentioned ones, eg. in order to determine the population per square mile, the operation must sum up the population, calculate the area, and divide the sum by the area.

CONCLUSION

The step from the local to the regional information most often needs additonal meta-data which we can express as rules in predicate calculus. We found that an aggregation can be specified by stating

- which entities we have to derive from
- which entity we want to project to
- how the first entities are connected to the latter
- which aggregation operations have to be used to combine the local data to deliver the expected regional information.

Tools from the artificial intelligence like PROLOG systems /CLOCKSIN/ are very useful to express such connections easily and without a need to formal programming.

/FRANK/ A. Frank, Combining a Network Database with Logic Programming, Department of Civil Engineering, University of Maine at Orono, Report No. 45, 1984

/CLOCKSIN/ W.F. Clocksin, C.S. Mellish, Programming in PROLOG, Springer Verlag, 1981

Authors:
Dipl.-Ing. M.Egenhofer, Dr. A.Frank,
Department of Civil Engineering,
University of Maine at Orono,
Orono ME 04469