Week 8 Distributed Database Design Tutorial

Fragmentation and Allocation

Consider the database schema

PRODUCTION (BatchNumber, PartType, Model, Quantity, Machine)

SALESPERSON (Name, City, Address, Phone)

DISPATCH (BatchNumber, Lot, Client, SalesPerson, Amount)

CLIENT (Name, City, Address, Phone)

Assume four **production centres** located in Dublin, San José, Zurich and Taiwan. Each production centre is responsible for one type of part. The parts are CPU (Dublin), Keyboard (San José), Screen (Zurich) and Cable (Taiwan).

There are three **sales points**, located in San José, Zurich and Taiwan. The sales are distributed by geographic location; thus, Taiwan clients are served only by salespeople in Taiwan. Any sales point can sell all parts. The City of the CLIENT relation represents the sales point used by the client, except for Dublin clients, who are served by salespeople in Zurich. There are no clients in cities other than Dublin, San José, Zurich and Taiwan.

The description of a part (PartType and Model) seldom changes, and production centres often check descriptions of batches manufactured at other production centres. The quantity and machine change daily for parts. A day's production is called a batch.

New clients are added frequently, but updates of client details are less frequent. Queries on clients from other cities are not common.

The sales point in Zurich is responsible for payroll for all salespeople.

The company wishes to distribute its database to nodes at each of the 4 production centres. Design the fragmentation (horizontal, vertical, and mixed) of the tables PRODUCTION, DISPATCH, CLIENT and SALESPERSON. You need to consider the potential frequency of modifications against the frequency of access.

Allocate your fragments to appropriate cities. You can replicate your fragments if you consider it will be more efficient.