Appendix-A

List of publication in National and International Journals


## Appendix-B

### List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>Autonomic Computing</td>
</tr>
<tr>
<td>ACS</td>
<td>Autonomic Computing System</td>
</tr>
<tr>
<td>ADBMS</td>
<td>Autonomic DataBase Management Systems</td>
</tr>
<tr>
<td>ADDM</td>
<td>Automatic Database Diagnostic Monitor</td>
</tr>
<tr>
<td>ANS</td>
<td>Autonomic Nervous System</td>
</tr>
<tr>
<td>DARPA</td>
<td>Defense Advanced Research Projects Agency</td>
</tr>
<tr>
<td>DASADA</td>
<td>Dynamic Assembly for Systems Adaptability Dependability and Assurance</td>
</tr>
<tr>
<td>DBAs</td>
<td>DataBase Administrators</td>
</tr>
<tr>
<td>DBMS</td>
<td>DataBase Management Systems</td>
</tr>
<tr>
<td>DT</td>
<td>Decision Tree</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>HI</td>
<td>Human Intervention</td>
</tr>
<tr>
<td>ID3</td>
<td>Iterative Dichotomiser</td>
</tr>
<tr>
<td>IT</td>
<td>Information Technology</td>
</tr>
<tr>
<td>KDD</td>
<td>Knowledge Discovery and Data mining</td>
</tr>
<tr>
<td>MAPE-K</td>
<td>Map Analyze Plan Execute-Knowledge</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>OLAP</td>
<td>Online Analytical Processing</td>
</tr>
<tr>
<td>OLTP</td>
<td>Online Transaction Processing</td>
</tr>
<tr>
<td>QEP</td>
<td>Query Execution Plan</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>QP</td>
<td>Query Patroller</td>
</tr>
<tr>
<td>SAS</td>
<td>Situational Awareness System</td>
</tr>
<tr>
<td>SLA</td>
<td>Service Level Agreement</td>
</tr>
<tr>
<td>SQL</td>
<td>SeQueL</td>
</tr>
<tr>
<td>SUO</td>
<td>Small Units Operations</td>
</tr>
<tr>
<td>TPC</td>
<td>Transaction Processing Council</td>
</tr>
</tbody>
</table>
Appendix-C
TPC-H Benchmark

The TPC-H benchmark is a decision support benchmark developed by the Transaction Performance Council (TPC). It is used to evaluate the performance of decision support systems by virtue of executing a set of complex queries against a standard database (containing large volume of data) under controlled conditions in order to give answers to real-world business questions. It consists of a group of business oriented ad-hoc queries and concurrent data modifications. The queries and the data populating the database have been chosen to have broad industry-wide relevance while maintaining a sufficient degree of ease of implementation. These queries have been given a realistic context, portraying the activity of a wholesale supplier to help the reader relate intuitively to the components of the benchmark[90].

TPC benchmarks versions are released periodically. Version 2.1.6.0 was referred for experimentation here.

A TPC-H database contains eight base tables. The relationships between these tables are illustrated in the figure A.1 below. The arrows point in the direction of one-to-many relationships between tables. The parentheses following each table name defines the prefix of the column names for that table. The number below each table name represents the cardinality (number of rows) of the table. A TPC-H database with a scaling factor 1 is approximately 1 GB large in size.

The purpose of this benchmark is to reduce the diversity of operations found in an information analysis application, while retaining the application's essential performance characteristics, namely: the level of system utilization and the complexity of operations. A large number of queries of various types and complexities needs to be executed to completely manage a business analysis environment. Many of the queries are not of primary interest for performance analysis because of the length of time the queries run, the system resources they use and the frequency of their execution [90].
The queries that have been selected exhibit the following characteristics:

- They have a high degree of complexity;
- They use a variety of access;
- They are of an ad hoc nature;
- They examine a large percentage of the available data;
- They all differ from each other;
- They contain query parameters that change across query executions.

These selected queries provide answers to the following classes of business analysis:

- Pricing and promotions;
- Supply and demand management;
- Profit and revenue management;
- Customer satisfaction study;
- Market share study;
- Shipping management.
Study and analysis of autonomic components to improve the performance of some database applications

Figure A.1: TPC-H schema

TPC-H defines twenty-two decision support queries (Q1 to Q22). The queries used here for experimentation were Q10, Q11, Q16 and Q18.
The details of these queries are given below.

1. **Returned Item Reporting Query (Q10)**

The query identifies customers who might be having problems with the parts that are shipped to them.

*Business question:* The Returned Item Reporting Query finds the top 20 customers, in terms of their effect on lost revenue for a given quarter, who have returned parts. The query considers only parts that were ordered in the specified quarter. The query lists the customer's name, address, nation, phone number, account balance, comment information and revenue lost. The customers are listed in descending order of lost revenue. Revenue lost is defined as assume \((L_{extended\ price} \times (1 - l_{discount}))\) for all qualifying lineitems.

Select

\[
\begin{align*}
  &c_{custkey}, \\
  &c_{name}, \\
  &\text{sum}(l_{extendedprice} \times (1 - l_{discount})) \text{ as revenue,} \\
  &c_{acctbal}, \\
  &n_{name}, \\
  &c_{address}, \\
  &c_{phone}, \\
  &c_{comment}
\end{align*}
\]

from

\[
\text{customer,} \\
\text{orders,} \\
\text{lineitem,} \\
\text{nation}
\]

where

\[
\begin{align*}
  &c_{custkey} = o_{custkey} \\
  &l_{orderkey} = o_{orderkey} \\
  &o_{orderdate} >= 1993-10-01 \\
  &o_{orderdate} < 1993-10-01 \\
  &l_{returnflag} = 'R' \\
  &c_{nationkey} = n_{nationkey}
\end{align*}
\]

group by

\[
\begin{align*}
  &c_{custkey}, \\
  &c_{name}, \\
  &c_{acctbal}, \\
  &c_{phone}, \\
  &n_{name}, \\
  &c_{address}, \\
  &c_{comment}
\end{align*}
\]

order by
2. **Important Stock Identification Query (Q11)**

This query finds the most important subset of suppliers' stock in a given nation.

**Business Question:** The Important Stock Identification Query finds, from scanning the available stock of suppliers in a given nation, all the parts that represent a significant percentage of the total value of all available parts. The query displays the part number and the value of those parts in descending order of value.

```sql
select ps_partkey, 
    sum(ps_supplycost * ps_availqty) as value
from partsupp, supplier, nation
where ps_suppkey = s_suppkey 
    and s_nationkey = n_nationkey 
    and n_name = 'GERMANY'
group by ps_partkey
having sum(ps_supplycost * ps_availqty) > ( 
    select sum(ps_supplycost * ps_availqty) * 0.0001 
    from partsupp, supplier, nation
where ps_suppkey = s_suppkey 
    and s_nationkey = n_nationkey 
    and n_name = 'GERMANY'
)
order by value desc;
```
3. **Parts/Supplier relationship query (Q16)**

This query finds out how many suppliers can supply parts with given attributes. It might be used, for example, to determine whether there is a sufficient number of suppliers for heavily ordered parts.

**Business Question**: The Parts/Supplier Relationship Query counts the number of suppliers who can supply parts that satisfy a particular customer's requirements. The customer is interested in parts of eight different sizes as long as they are not of a given type, not of a given brand, and not from a supplier who has had complaints registered at the Better Business Bureau.

```sql
select p_brand, p_type, p_size, count(distinct ps_suppkey) as supplier_cnt
from partsupp, part
where p_partkey = ps_partkey
  and p_brand <> 'Brand#13'
  and p_type not like '[TYPE]%'
  and p_size in (1, 2, 3, 4, 5, 6, 7, 8)
  and ps_suppkey not in (select s_suppkey
                        from supplier
                        where s_comment like 'each slyly above the careful')
group by p_brand, p_type, p_size
order by supplier_cnt desc, p_brand, p_type, p_size;
```
4. Large volume customer Query (Q18)

The Large Volume Customer Query ranks customers based on their having placed a large quantity order.

Business Question: The large volume customer query finds a list of the top 100 customers who have ever placed large quantity orders. The query lists the customer name, customer key, the order key, date and total price and the quantity for the order.

```
select c_name,
c_custkey,
o_orderkey,
o_orderdate,
o_totalprice,
sum(l_quantity)
from customer,
orders,
lineitem
where o_orderkey in (select l_orderkey
from lineitem
group by l_orderkey having sum(l_quantity) > 300)
and c_custkey = o_custkey
and o_orderkey = l_orderkey
group by c_name,
c_custkey,
o_orderkey,
o_orderdate,
o_totalprice
order by o_totalprice desc,
o_orderdate;
```