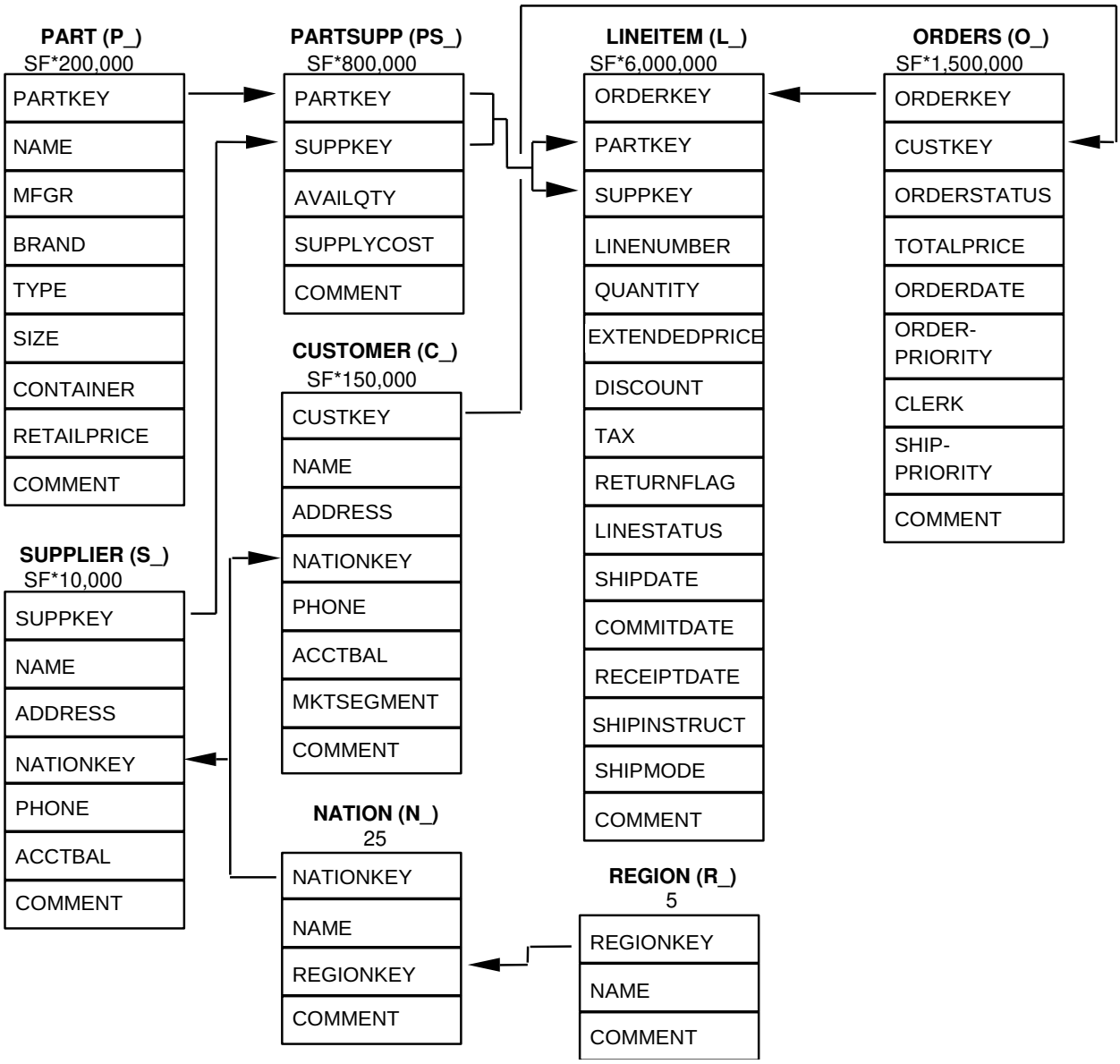
1.2 Database Entities, Relationships, and Characteristics

The components of the TPC-H database are defined to consist of eight separate and individual tables (the Base Tables). The relationships between columns of these tables are illustrated in Figure 2: The TPC-H Schema.

Figure 2: The TPC-H Schema



Legend:

- The parentheses following each table name contain the prefix of the column names for that table;
- The arrows point in the direction of the one-to-many relationships between tables;
- The number/formula below each table name represents the cardinality (number of rows) of the table. Some are factored by SF, the Scale Factor, to obtain the chosen database size. The cardinality for the LINEITEM table is approximate (see Clause 4.2.5).

2.4 Query Definitions

For each query a single example output row is shown (even though queries often produce multiple rows) along with the column headers. This is for illustration only. See Appendix F: for the precise validation output for each query.

2.4.1 Pricing Summary Report Query (Q1)

This query reports the amount of business that was billed, shipped, and returned.

2.4.1.1 Business Question

The Pricing Summary Report Query provides a summary pricing report for all lineitems shipped as of a given date. The date is within 60 - 120 days of the greatest ship date contained in the database. The query lists totals for extended price, discounted extended price, discounted extended price plus tax, average quantity, average extended price, and average discount. These aggregates are grouped by RETURNFLAG and LINESTATUS, and listed in ascending order of RETURNFLAG and LINESTATUS. A count of the number of lineitems in each group is included.

2.4.1.2 Functional Query Definition

```
select
        l_returnflag,
        l_linestatus,
        sum(l_quantity) as sum_qty,
        sum(l_extendedprice) as sum_base_price,
        sum(l_extendedprice*(1-l_discount)) as sum_disc_price,
        sum(l_extendedprice*(1-l_discount)*(1+l_tax)) as sum_charge,
        avg(l_quantity) as avg_qty,
        avg(l_extendedprice) as avg_price,
        avg(l_discount) as avg_disc,
        count(*) as count_order
from
        lineitem
where
        l_shipdate <= date '1998-12-01' - interval '[DELTA]' day (3)
group by
        l_returnflag,
        l_linestatus
order by
        l_returnflag,
        l_linestatus;
```

2.4.1.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. DELTA is randomly selected within [60. 120].

Comment: 1998-12-01 is the highest possible ship date as defined in the database population. (This is ENDDATE - 30). The query will include all lineitems shipped before this date minus DELTA days. The intent is to choose DELTA so that between 95% and 97% of the rows in the table are scanned.

2.4.1.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. DELTA = 90.

2.4.1.5 Sample Output

L_RETURNFLAG	L_LINESTATUS	SUM_QTY	SUM_BASE_PRICE	SUM_DISC_PRICE
A	F	37734107.00	56586554400.73	53758257134.87

SUM_CHARGE	AVG_QTY	AVG_PRICE	AVG_DISC	COUNT_ORDER
55909065222.83	25.52	38273.13	.05	1478493

2.4.2 Minimum Cost Supplier Query (Q2)

This query finds which supplier should be selected to place an order for a given part in a given region.

2.4.2.1 Business Question

The Minimum Cost Supplier Query finds, in a given region, for each part of a certain type and size, the supplier who can supply it at minimum cost. If several suppliers in that region offer the desired part type and size at the same (minimum) cost, the query lists the parts from suppliers with the 100 highest account balances. For each supplier, the query lists the supplier's account balance, name and nation; the part's number and manufacturer; the supplier's address, phone number and comment information.

2.4.2.2 Functional Query Definition

Return the first 100 selected rows

```
select
        s_acctbal,
        s_name,
        n_name,
        p_partkey,
        p_mfgr,
        s_address,
        s_phone,
        s_comment
from
        part,
        supplier,
        partsupp,
        nation,
        region
where
        p_partkey = ps_partkey
        and s_suppkey = ps_suppkey
        and p_size = [SIZE]
        and p_type like '%[TYPE]'
        and s_nationkey = n_nationkey
        and n_regionkey = r_regionkey
        and r_name = '[REGION]'
        and ps_supplycost = (
                select
```

```
                min(ps_supplycost)
        from
                partsupp, supplier,
                nation, region
        where
                p_partkey = ps_partkey
                and s_suppkey = ps_suppkey
                and s_nationkey = n_nationkey
                and n_regionkey = r_regionkey
                and r_name = '[REGION]'
        )
order by
        s_acctbal desc,
        n_name,
        s_name,
        p_partkey;
```

2.4.2.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. SIZE is randomly selected within [1. 50];
2. TYPE is randomly selected within the list Syllable 3 defined for Types in Clause 4.2.2.13;
3. REGION is randomly selected within the list of values defined for R_NAME in 4.2.3.

2.4.2.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. SIZE = 15;
2. TYPE = BRASS;
3. REGION = EUROPE.

2.4.2.5 Sample Output

S_ACCTBAL	S_NAME	N_NAME	P_PARTKEY	P_MFGR
9938.53	Supplier#000005359	UNITED KINGDOM	185358	Manufacturer#4

S_ADDRESS	S_PHONE	S_COMMENT
QKuHYh,vZGiwu2FW EJoLDx04	33-429-790-6131	uriously regular requests hag

2.4.3 Shipping Priority Query (Q3)

This query retrieves the 10 unshipped orders with the highest value.

2.4.3.1 Business Question

The Shipping Priority Query retrieves the shipping priority and potential revenue, defined as the sum of l_extendedprice * (1-l_discount), of the orders having the largest revenue among those that had not been shipped as of a given date. Orders are listed in decreasing order of revenue. If more than 10 unshipped orders exist, only the 10 orders with the largest revenue are listed.

2.4.3.2 Functional Query Definition

Return the first 10 selected rows

```
select
        l_orderkey,
        sum(l_extendedprice*(1-l_discount)) as revenue,
        o_orderdate,
        o_shippriority
from
        customer,
        orders,
        lineitem
where
        c_mktsegment = '[SEGMENT]'
        and c_custkey = o_custkey
        and l_orderkey = o_orderkey
        and o_orderdate < date '[DATE]'
        and l_shipdate > date '[DATE]'
group by
        l_orderkey,
        o_orderdate,
        o_shippriority
order by
        revenue desc,
        o_orderdate;
```

2.4.3.3 Substitution Parameters

Values for the following substitution parameters must be generated and used to build the executable query text:

1. SEGMENT is randomly selected within the list of values defined for Segments in Clause 4.2.2.13;
2. DATE is a randomly selected day within [1995-03-01 .. 1995-03-31].

2.4.3.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. SEGMENT = BUILDING;
2. DATE = 1995-03-15.

2.4.3.5 Sample Output

L_ORDERKEY	REVENUE	O_ORDERDATE	O_SHIPPRIORITY
2456423	406181.01	1995-03-05	0

2.4.4 Order Priority Checking Query (Q4)

This query determines how well the order priority system is working and gives an assessment of customer satisfaction.

2.4.4.1 Business Question

The Order Priority Checking Query counts the number of orders ordered in a given quarter of a given year in which at least one lineitem was received by the customer later than its committed date. The query lists the count of such orders for each order priority sorted in ascending priority order.

2.4.4.2 Functional Query Definition

```
select
        o_orderpriority,
        count(*) as order_count
from
        orders
where
        o_orderdate >= date '[DATE]'
        and o_orderdate < date '[DATE]' + interval '3' month
        and exists (
                select
                        *
                from
                        lineitem
                where
                        l_orderkey = o_orderkey
                        and l_commitdate < l_receiptdate
        )
group by
        o_orderpriority
order by
        o_orderpriority;
```

2.4.4.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. DATE is the first day of a randomly selected month between the first month of 1993 and the 10th month of 1997.

2.4.4.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. DATE = 1993-07-01.

2.4.4.5 Sample Output

O_ORDERPRIORITY	ORDER_COUNT
1-URGENT	10594

2.4.5 Local Supplier Volume Query (Q5)

This query lists the revenue volume done through local suppliers.

2.4.5.1 Business Question

The Local Supplier Volume Query lists for each nation in a region the revenue volume that resulted from lineitem transactions in which the customer ordering parts and the supplier filling them were both within that nation. The query is run in order to determine whether to institute local distribution centers in a given region. The query considers only parts ordered in a given year. The query displays the nations and revenue volume in descending order by revenue. Revenue volume for all qualifying lineitems in a particular nation is defined as sum(l_extendedprice * (1 - l_discount)).

2.4.5.2 Functional Query Definition

```
select
        n_name,
        sum(l_extendedprice * (1 - l_discount)) as revenue
from
        customer,
        orders,
        lineitem,
        supplier,
        nation,
        region
where
        c_custkey = o_custkey
        and l_orderkey = o_orderkey
        and l_suppkey = s_suppkey
        and c_nationkey = s_nationkey
        and s_nationkey = n_nationkey
        and n_regionkey = r_regionkey
        and r_name = '[REGION]'
        and o_orderdate >= date '[DATE]'
        and o_orderdate < date '[DATE]' + interval '1' year
group by
        n_name
order by
        revenue desc;
```

2.4.5.3 Substitution Parameters

Values for the following substitution parameters must be generated and used to build the executable query text:

1. REGION is randomly selected within the list of values defined for R_NAME in C;aise 4.2.3;
2. DATE is the first of January of a randomly selected year within [1993 .. 1997].

2.4.5.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:

Values for substitution parameters:

1. REGION = ASIA;
2. DATE = 1994-01-01.

2.4.5.5 Sample Output

N_NAME	REVENUE
INDONESIA	55502041.17

2.4.6 Forecasting Revenue Change Query (Q6)

This query quantifies the amount of revenue increase that would have resulted from eliminating certain company-wide discounts in a given percentage range in a given year. Asking this type of "what if" query can be used to look for ways to increase revenues.

2.4.6.1 Business Question

The Forecasting Revenue Change Query considers all the lineitems shipped in a given year with discounts between DISCOUNT-0.01 and DISCOUNT+0.01. The query lists the amount by which the total revenue would have increased if these discounts had been eliminated for lineitems with l_quantity less than quantity. Note that the potential revenue increase is equal to the sum of [l_extendedprice * l_discount] for all lineitems with discounts and quantities in the qualifying range.

2.4.6.2 Functional Query Definition

```
select
        sum(l_extendedprice*l_discount) as revenue
from
        lineitem
where
        l_shipdate >= date '[DATE]'
        and l_shipdate < date '[DATE]' + interval '1' year
        and l_discount between [DISCOUNT] - 0.01 and [DISCOUNT] + 0.01
        and l_quantity < [QUANTITY];
```

2.4.6.3 Substitution Parameters

Values for the following substitution parameters must be generated and used to build the executable query text:

1. DATE is the first of January of a randomly selected year within [1993 .. 1997];
2. DISCOUNT is randomly selected within [0.02 .. 0.09];
3. QUANTITY is randomly selected within [24 .. 25].

2.4.6.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. DATE = 1994-01-01;
2. DISCOUNT = 0.06;
3. QUANTITY = 24.

2.4.6.5 Sample Output

REVENUE
123141078.23

2.4.7 Volume Shipping Query (Q7)

This query determines the value of goods shipped between certain nations to help in the re-negotiation of shipping contracts.

2.4.7.1 Business Question

The Volume Shipping Query finds, for two given nations, the gross discounted revenues derived from lineitems in which parts were shipped from a supplier in either nation to a customer in the other nation during 1995 and 1996. The query lists the supplier nation, the customer nation, the year, and the revenue from shipments that took place in that year. The query orders the answer by Supplier nation, Customer nation, and year (all ascending).

2.4.7.2 Functional Query Definition

```
select
        supp_nation,
        cust_nation,
        l_year, sum(volume) as revenue
from (
        select
                n1.n_name as supp_nation,
                n2.n_name as cust_nation,
                extract(year from l_shipdate) as l_year,
                l_extendedprice * (1 - l_discount) as volume
        from
                supplier,
                lineitem,
                orders,
                customer,
                nation n1,
                nation n2
        where
                s_suppkey = l_suppkey
                and o_orderkey = l_orderkey
                and c_custkey = o_custkey
                and s_nationkey = n1.n_nationkey
                and c_nationkey = n2.n_nationkey
                and (
                        (n1.n_name = '[NATION1]' and n2.n_name = '[NATION2]')
                        or (n1.n_name = '[NATION2]' and n2.n_name = '[NATION1]')
                )
                and l_shipdate between date '1995-01-01' and date '1996-12-31'
        ) as shipping
group by
        supp_nation,
        cust_nation,
        l_year
order by
        supp_nation,
        cust_nation,
        l_year;
```

2.4.7.3 Substitution Parameters

Values for the following substitution parameters must be generated and used to build the executable query text:

1. NATION1 is randomly selected within the list of values defined for N_NAME in Clause 4.2.3;
2. NATION2 is randomly selected within the list of values defined for N_NAME in Clause 4.2.3 and must be different from the value selected for NATION1 in item 1 above.

2.4.7.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:

Values for substitution parameters:

1. NATION1 = FRANCE;
2. NATION2 = GERMANY.

2.4.7.5 Sample Output

SUPP_NATION	CUST_NATION	YEAR	REVENUE
FRANCE	GERMANY	1995	54639732.73

2.4.8 National Market Share Query (Q8)

This query determines how the market share of a given nation within a given region has changed over two years for a given part type.

2.4.8.1 Business Question

The market share for a given nation within a given region is defined as the fraction of the revenue, the sum of [l_extendedprice * (1-l_discount)], from the products of a specified type in that region that was supplied by suppliers from the given nation. The query determines this for the years 1995 and 1996 presented in this order.

2.4.8.2 Functional Query Definition

```
select
        o_year,
        sum(case
                when nation = '[NATION]'
                then volume
                else 0
        end) / sum(volume) as mkt_share
from (
        select
                extract(year from o_orderdate) as o_year,
                l_extendedprice * (1-l_discount) as volume,
                n2.n_name as nation
        from
                part,
                supplier,
                lineitem,
                orders,
                customer,
                nation n1,
                nation n2,
                region
        where
                p_partkey = l_partkey
                and s_suppkey = l_suppkey
                and l_orderkey = o_orderkey
                and o_custkey = c_custkey
                and c_nationkey = n1.n_nationkey
                and n1.n_regionkey = r_regionkey
                and r_name = '[REGION]'
                and s_nationkey = n2.n_nationkey
                and o_orderdate between date '1995-01-01' and date '1996-12-31'
                and p_type = '[TYPE]'
        ) as all_nations
group by
        o_year
order by
        o_year;
```

2.4.8.3 Substitution Parameters

Values for the following substitution parameters must be generated and used to build the executable query text:

1. NATION is randomly selected within the list of values defined for N_NAME in Clause 4.2.3;
2. REGION is the value defined in Clause 4.2.3 for R_NAME where R_REGIONKEY corresponds to N_REGIONKEY for the selected NATION in item 1 above;
3. TYPE is randomly selected within the list of 3-syllable strings defined for Types in Clause 4.2.2.13.

2.4.8.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. NATION = BRAZIL;
2. REGION = AMERICA;
3. TYPE = ECONOMY ANODIZED STEEL.

2.4.8.5 Sample Output

YEAR	MKT_SHARE
1995	.03

2.4.9 Product Type Profit Measure Query (Q9)

This query determines how much profit is made on a given line of parts, broken out by supplier nation and year.

2.4.9.1 Business Question

The Product Type Profit Measure Query finds, for each nation and each year, the profit for all parts ordered in that year that contain a specified substring in their names and that were filled by a supplier in that nation. The profit is defined as the sum of [(l_extendedprice*(1-l_discount)) - (ps_supplycost * l_quantity)] for all lineitems describing parts in the specified line. The query lists the nations in ascending alphabetical order and, for each nation, the year and profit in descending order by year (most recent first).

2.4.9.2 Functional Query Definition

```
select
        nation,
        o_year,
        sum(amount) as sum_profit
from (
        select
                n_name as nation,
                extract(year from o_orderdate) as o_year,
                l_extendedprice * (1 - l_discount) - ps_supplycost * l_quantity as amount
        from
                part,
                supplier,
                lineitem,
                partsupp,
                orders,
                nation
        where
                s_suppkey = l_suppkey
                and ps_suppkey = l_suppkey
                and ps_partkey = l_partkey
                and p_partkey = l_partkey
                and o_orderkey = l_orderkey
                and s_nationkey = n_nationkey
                and p_name like '%[COLOR]%'
        ) as profit
group by
        nation,
        o_year
order by
        nation,
        o_year desc;
```

2.4.9.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. COLOR is randomly selected within the list of values defined for the generation of P_NAME in Clause 4.2.3.

2.4.9.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. COLOR = green.

2.4.9.5 Sample Output

NATION	YEAR	SUM_PROFIT
ALGERIA	1998	31342867.24

2.4.10 Returned Item Reporting Query (Q10)

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

2.4.10.1 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 sum(l_extendedprice*(1-l_discount)) for all qualifying lineitems.

2.4.10.2 Functional Query Definition

Return the first 20 selected rows

```
select
        c_custkey,
        c_name,
        sum(l_extendedprice * (1 - l_discount)) as revenue,
        c_acctbal,
        n_name,
        c_address,
        c_phone,
        c_comment
from
        customer,
        orders,
        lineitem,
        nation
where
        c_custkey = o_custkey
        and l_orderkey = o_orderkey
        and o_orderdate >= date '[DATE]'
        and o_orderdate < date '[DATE]' + interval '3' month
        and l_returnflag = 'R'
        and c_nationkey = n_nationkey
group by
        c_custkey,
        c_name,
        c_acctbal,
        c_phone,
        n_name,
        c_address,
        c_comment
order by
        revenue desc;
```

2.4.10.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. DATE is the first day of a randomly selected month from the second month of 1993 to the first month of 1995.

2.4.10.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. DATE = 1993-10-01.

2.4.10.5 Sample Output

C_CUSTKEY	C_NAME	REVENUE	C_ACCTBAL	N_NAME
57040	Customer#000057040	734235.24	632.87	JAPAN

C_ADDRESS	C_PHONE	C_COMMENT
Eioyzjf4pp	22-895-641-3466	sits. slyly regular requests sleep alongside of the regular inst

2.4.11 Important Stock Identification Query (Q11)

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

2.4.11.1 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.

2.4.11.2 Functional Query Definition

```
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 = '[NATION]'
group by
        ps_partkey having
                sum(ps_supplycost * ps_availqty) > (
                        select
                                sum(ps_supplycost * ps_availqty) * [FRACTION]
                        from
                                partsupp,
                                supplier,
                                nation
                        where
                                ps_suppkey = s_suppkey
                                and s_nationkey = n_nationkey
                                and n_name = '[NATION]'
                )
order by
        value desc;
```

2.4.11.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. NATION is randomly selected within the list of values defined for N_NAME in Clause 4.2.3;
2. FRACTION is chosen as 0.0001 / SF.

2.4.11.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:

Values for substitution parameters:

1. NATION = GERMANY;
2. FRACTION = 0.0001.

2.4.11.5 Sample Output

PS_PARTKEY	VALUE
129760	17538456.86

2.4.12 Shipping Modes and Order Priority Query (Q12)

This query determines whether selecting less expensive modes of shipping is negatively affecting the critical-priority orders by causing more parts to be received by customers after the committed date.

2.4.12.1 Business Question

The Shipping Modes and Order Priority Query counts, by ship mode, for lineitems actually received by customers in a given year, the number of lineitems belonging to orders for which the l_receiptdate exceeds the l_commitdate for two different specified ship modes. Only lineitems that were actually shipped before the l_commitdate are considered. The late lineitems are partitioned into two groups, those with priority URGENT or HIGH, and those with a priority other than URGENT or HIGH.

2.4.12.2 Functional Query Definition

```
select
        l_shipmode,
        sum(case
                when o_orderpriority ='1-URGENT'
                        or o_orderpriority ='2-HIGH'
                then 1
                else 0
        end) as high_line_count,
        sum(case
                when o_orderpriority <> '1-URGENT'
                        and o_orderpriority <> '2-HIGH'
                then 1
                else 0
        end) as low_line_count
from
        orders,
        lineitem
where
        o_orderkey = l_orderkey
        and l_shipmode in ('[SHIPMODE1]', '[SHIPMODE2]')
        and l_commitdate < l_receiptdate
        and l_shipdate < l_commitdate
        and l_receiptdate >= date '[DATE]'
        and l_receiptdate < date '[DATE]' + interval '1' year
group by
        l_shipmode
order by
        l_shipmode;
```

2.4.12.3 Substitution Parameters

Values for the following substitution parameters must be generated and used to build the executable query text:

1. SHIPMODE1 is randomly selected within the list of values defined for Modes in Clause 4.2.2.13;
2. SHIPMODE2 is randomly selected within the list of values defined for Modes in Clause 4.2.2.13 and must be different from the value selected for SHIPMODE1 in item 1;
3. DATE is the first of January of a randomly selected year within [1993 .. 1997].

2.4.12.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. SHIPMODE1 = MAIL;

2. SHIPMODE2 = SHIP;
3. DATE = 1994-01-01.

2.4.12.5 Sample Output

L_SHIPMODE	HIGH_LINE_COUNT	LOW_LINE_COUNT
MAIL	6202	9324

2.4.13 Customer Distribution Query (Q13)

This query seeks relationships between customers and the size of their orders.

2.4.13.1 Business Question

This query determines the distribution of customers by the number of orders they have made, including customers who have no record of orders, past or present. It counts and reports how many customers have no orders, how many have 1, 2, 3, etc. A check is made to ensure that the orders counted do not fall into one of several special categories of orders. Special categories are identified in the order comment column by looking for a particular pattern.

2.4.13.2 Functional Query Definition

```
select
        c_count, count(*) as custdist
from (
        select
                c_custkey,
                count(o_orderkey)
        from
                customer left outer join orders on
                        c_custkey = o_custkey
                        and o_comment not like '%[WORD1]%[WORD2]%'
        group by
                c_custkey
        )as c_orders (c_custkey, c_count)
group by
        c_count
order by
        custdist desc,
        c_count desc;
```

2.4.13.3 Substitution Parameters

1. WORD1 is randomly selected from 4 possible values: special, pending, unusual, express.
2. WORD2 is randomly selected from 4 possible values: packages, requests, accounts, deposits.

2.4.13.4 Query Validation

For validation against the qualification database the query must be executed using the following substitution parameters and must produce the following output data:

Values for substitution parameters:

1. WORD1 = special.
2. WORD2 = requests.

2.4.13.5 Sample Output

C_COUNT	CUSTDIST
9	6641

2.4.14 Promotion Effect Query (Q14)

This query monitors the market response to a promotion such as TV advertisements or a special campaign.

2.4.14.1 Business Question

The Promotion Effect Query determines what percentage of the revenue in a given year and month was derived from promotional parts. The query considers only parts actually shipped in that month and gives the percentage. Revenue is defined as (l_extendedprice * (1-l_discount)).

2.4.14.2 Functional Query Definition

```
select
        100.00 * sum(case
                when p_type like 'PROMO%'
                then l_extendedprice*(1-l_discount)
                else 0
        end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
from
        lineitem,
        part
where
        l_partkey = p_partkey
        and l_shipdate >= date '[DATE]'
        and l_shipdate < date '[DATE]' + interval '1' month;
```

2.4.14.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. DATE is the first day of a month randomly selected from a random year within [1993 .. 1997].

2.4.14.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. DATE = 1995-09-01.

2.4.14.5 Sample Output

PROMO_REVENUE
16.38

2.4.15 Top Supplier Query (Q15)

This query determines the top supplier so it can be rewarded, given more business, or identified for special recognition.

2.4.15.1 Business Question

The Top Supplier Query finds the supplier who contributed the most to the overall revenue for parts shipped during a given quarter of a given year. In case of a tie, the query lists all suppliers whose contribution was equal to the maximum, presented in supplier number order.

2.4.15.2 Functional Query Definition

```
create view revenue[STREAM_ID] (supplier_no, total_revenue) as
        select
                l_suppkey,
                sum(l_extendedprice * (1 - l_discount))
        from
                lineitem
        where
                l_shipdate >= date '[DATE]'
                and l_shipdate < date '[DATE]' + interval '3' month
        group by
                l_suppkey;

select
        s_suppkey,
        s_name,
        s_address,
        s_phone,
        total_revenue
from
        supplier,
        revenue[STREAM_ID]
where
        s_suppkey = supplier_no
        and total_revenue = (
                select
                        max(total_revenue)
                from
                        revenue[STREAM_ID]
        )
order by
        s_suppkey;

drop view revenue[STREAM_ID];
```

2.4.15.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. DATE is the first day of a randomly selected month between the first month of 1993 and the 10th month of 1997.

2.4.15.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. DATE = 1996-01-01.

2.4.15.5 Sample Output

S_SUPPKEY	S_NAME	S_ADDRESS	S_PHONE	TOTAL_REVENUE
8449	Supplier#000008449	Wp34zim9qYFbVctdW	20-469-856-8873	1772627.21

2.4.16 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.

2.4.16.1 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. Results must be presented in descending count and ascending brand, type, and size.

2.4.16.2 Functional Query Definition

```
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]'
        and p_type not like '[TYPE]%'
        and p_size in ([SIZE1], [SIZE2], [SIZE3], [SIZE4], [SIZE5], [SIZE6], [SIZE7], [SIZE8])
        and ps_suppkey not in (
                select
                        s_suppkey
                from
                        supplier
                where
                        s_comment like '%Customer%Complaints%'
        )
group by
        p_brand,
        p_type,
        p_size
order by
        supplier_cnt desc,
        p_brand,
        p_type,
        p_size;
```

2.4.16.3 Substitution Parameters

Values for the following substitution parameters must be generated and used to build the executable query text:

1. BRAND = Brand#MN where M and N are two single character strings representing two numbers randomly and independently selected within [1 .. 5];
2. TYPE is made of the first 2 syllables of a string randomly selected within the list of 3-syllable strings defined for Types in Clause 4.2.2.13;
3. SIZE1 is randomly selected as a set of eight different values within [1 .. 50];
4. SIZE2 is randomly selected as a set of eight different values within [1 .. 50];
5. SIZE3 is randomly selected as a set of eight different values within [1 .. 50];
6. SIZE4 is randomly selected as a set of eight different values within [1 .. 50];

7. SIZE5 is randomly selected as a set of eight different values within [1 .. 50];
8. SIZE6 is randomly selected as a set of eight different values within [1 .. 50];
9. SIZE7 is randomly selected as a set of eight different values within [1 .. 50];
10. SIZE8 is randomly selected as a set of eight different values within [1 .. 50].

2.4.16.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:

Values for substitution parameters:

1. BRAND = Brand#45.
2. TYPE = MEDIUM POLISHED .
3. SIZE1 = 49
4. SIZE2 = 14
5. SIZE3 = 23
6. SIZE4 = 45
7. SIZE5 = 19
8. SIZE6 = 3
9. SIZE7 = 36
10. SIZE8 = 9.

2.4.16.5 Sample Output

P_BRAND	P_TYPE	P_SIZE	SUPPLIER_CNT
Brand#41	MEDIUM BRUSHED TIN	3	28

2.4.17 Small-Quantity-Order Revenue Query (Q17)

This query determines how much average yearly revenue would be lost if orders were no longer filled for small quantities of certain parts. This may reduce overhead expenses by concentrating sales on larger shipments.

2.4.17.1 Business Question

The Small-Quantity-Order Revenue Query considers parts of a given brand and with a given container type and determines the average lineitem quantity of such parts ordered for all orders (past and pending) in the 7-year database. What would be the average yearly gross (undiscounted) loss in revenue if orders for these parts with a quantity of less than 20% of this average were no longer taken?

2.4.17.2 Functional Query Definition

```
select
        sum(l_extendedprice) / 7.0 as avg_yearly
from
        lineitem,
        part
where
        p_partkey = l_partkey
        and p_brand = '[BRAND]'
        and p_container = '[CONTAINER]'
        and l_quantity < (
                select
                        0.2 * avg(l_quantity)
                from
                        lineitem
                where
                        l_partkey = p_partkey
        );
```

2.4.17.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. BRAND = 'Brand#MN' where MN is a two character string representing two numbers randomly and independently selected within [1 .. 5];
2. CONTAINER is randomly selected within the list of 2-syllable strings defined for Containers in Clause 4.2.2.13.

2.4.17.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. BRAND = Brand#23;
2. CONTAINER = MED BOX.

2.4.17.5 Sample Output

AVG_YEARLY
348406.05

2.4.18 Large Volume Customer Query (Q18)

The Large Volume Customer Query ranks customers based on their having placed a large quantity order. Large quantity orders are defined as those orders whose total quantity is above a certain level.

2.4.18.1 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.

2.4.18.2 Functional Query Definition

Return the first 100 selected rows

```
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) > [QUANTITY]
        )
        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;
```

2.4.18.3 Substitution Parameters

Values for the following substitution parameter must be generated and used to build the executable query text:

1. QUANTITY is randomly selected within [312..315].

2.4.18.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. QUANTITY = 300

2.4.18.5 Sample Output

C_NAME	C_CUSTKEY	O_ORDERKEY	O_ORDERDATE	O_TOTALPRICE	Sum(L_QUANTITY)
Customer#000128120	128120	4722021	1994-04-07	544089.09	323.00

2.4.19 Discounted Revenue Query (Q19)

The Discounted Revenue Query reports the gross discounted revenue attributed to the sale of selected parts handled in a particular manner. This query is an example of code such as might be produced programmatically by a data mining tool.

2.4.19.1 Business Question

The Discounted Revenue query finds the gross discounted revenue for all orders for three different types of parts that were shipped by air and delivered in person. Parts are selected based on the combination of specific brands, a list of containers, and a range of sizes.

2.4.19.2 Functional Query Definition

```
select
        sum(l_extendedprice * (1 - l_discount) ) as revenue
from
        lineitem,
        part
where
        (
                p_partkey = l_partkey
                and p_brand = '[BRAND1]'
                and p_container in ( 'SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
                and l_quantity >= [QUANTITY1] and l_quantity <= [QUANTITY1] + 10
                and p_size between 1 and 5
                and l_shipmode in ('AIR', 'AIR REG')
                and l_shipinstruct = 'DELIVER IN PERSON'
        )
        or
        (
                p_partkey = l_partkey
                and p_brand = '[BRAND2]'
                and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
                and l_quantity >= [QUANTITY2] and l_quantity <= [QUANTITY2] + 10
                and p_size between 1 and 10
                and l_shipmode in ('AIR', 'AIR REG')
                and l_shipinstruct = 'DELIVER IN PERSON'
        )
        or
                (
                p_partkey = l_partkey
                and p_brand = '[BRAND3]'
                and p_container in ( 'LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
                and l_quantity >= [QUANTITY3] and l_quantity <= [QUANTITY3] + 10
                and p_size between 1 and 15
                and l_shipmode in ('AIR', 'AIR REG')
                and l_shipinstruct = 'DELIVER IN PERSON'
        );
```

2.4.19.3 Substitution Parameters

1. QUANTITY1 is randomly selected within [1..10].
2. QUANTITY2 is randomly selected within [10..20].
3. QUANTITY3 is randomly selected within [20..30].
4. BRAND1, BRAND2, BRAND3 = 'Brand#MN' where each MN is a two character string representing two numbers randomly and independently selected within [1 .. 5]

2.4.19.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. QUANTITY1 = 1.
2. QUANTITY2 = 10.
3. QUANTITY3 = 20.
4. BRAND1 = Brand#12.
5. BRAND2 = Brand#23.
6. BRAND3 = Brand#34.

2.4.19.5 Sample Output

REVENUE
3083843.05

Comment: The TPC recognizes that the predicates on l_shipmode include the non-existing shipmode “AIR REG”.

2.4.20 Potential Part Promotion Query (Q20)

The Potential Part Promotion Query identifies suppliers in a particular nation having selected parts that may be candidates for a promotional offer.

2.4.20.1 Business Question

The Potential Part Promotion query identifies suppliers who have an excess of a given part available; an excess is defined to be more than 50% of the parts like the given part that the supplier shipped in a given year for a given nation. Only parts whose names share a certain naming convention are considered.

2.4.20.2 Functional Query Definition

```
select
        s_name,
        s_address
from
        supplier, nation
where
        s_suppkey in (
                select
                        ps_suppkey
                from
                        partsupp
                where
                        ps_partkey in (
                                select
                                        p_partkey
                                from
                                        part
                                where
                                        p_name like '[COLOR]%'
                        )
                and ps_availqty > (
                        select
                                0.5 * sum(l_quantity)
                        from
                                lineitem
                        where
                                l_partkey = ps_partkey
                                and l_suppkey = ps_suppkey
                                and l_shipdate >= date('[DATE]')
                                and l_shipdate < date('[DATE]') + interval '1' year
                )
        )
        and s_nationkey = n_nationkey
        and n_name = '[NATION]'
order by
        s_name;
```

2.4.20.3 Substitution Parameters

1. COLOR is randomly selected within the list of values defined for the generation of P_NAME.
2. DATE is the first of January of a randomly selected year within 1993..1997.
3. NATION is randomly selected within the list of values defined for N_NAME in Clause 4.2.3.

2.4.20.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:

Values for substitution parameters:

1. COLOR = forest.
2. DATE = 1994-01-01.
3. NATION = CANADA.

2.4.20.5 Sample Output

S_NAME	S_ADDRESS
Supplier#000000020	iybAE,RmTymrZVYaFZva2SH,j

2.4.21 Suppliers Who Kept Orders Waiting Query (Q21)

This query identifies certain suppliers who were not able to ship required parts in a timely manner.

2.4.21.1 Business Question

The Suppliers Who Kept Orders Waiting query identifies suppliers, for a given nation, whose product was part of a multi-supplier order (with current status of 'F') where they were the only supplier who failed to meet the committed delivery date.

2.4.21.2 Functional Query Definition

Return the first 100 selected rows.

```
select
        s_name,
        count(*) as numwait
from
        supplier,
        lineitem l1,
        orders,
        nation
where
        s_suppkey = l1.l_suppkey
        and o_orderkey = l1.l_orderkey
        and o_orderstatus = 'F'
        and l1.l_receiptdate > l1.l_commitdate
        and exists (
                select
                        *
                from
                        lineitem l2
                where
                        l2.l_orderkey = l1.l_orderkey
                        and l2.l_suppkey <> l1.l_suppkey
        )
        and not exists (
                select
                        *
                from
                        lineitem l3
                where
                        l3.l_orderkey = l1.l_orderkey
                        and l3.l_suppkey <> l1.l_suppkey
                        and l3.l_receiptdate > l3.l_commitdate
        )
        and s_nationkey = n_nationkey
        and n_name = '[NATION]'
group by
        s_name
order by
        numwait desc,
        s_name;
```

2.4.21.3 Substitution Parameters

1. NATION is randomly selected within the list of values defined for N_NAME in Clause 4.2.3.

2.4.21.4 Query Validation

For validation against the qualification database the query must be executed using the following values for substitution parameters and must produce the following output data:
Values for substitution parameters:

1. NATION = SAUDI ARABIA.

2.4.21.5 Sample Output

S_NAME	NUMWAIT
Supplier#000002829	20

2.4.22 Global Sales Opportunity Query (Q22)

The Global Sales Opportunity Query identifies geographies where there are customers who may be likely to make a purchase.

2.4.22.1 Business Question

This query counts how many customers within a specific range of country codes have not placed orders for 7 years but who have a greater than average “positive” account balance. It also reflects the magnitude of that balance. Country code is defined as the first two characters of c_phone.

2.4.22.2 Functional Query Definition

```
select
        cntrycode,
        count(*) as numcust,
        sum(c_acctbal) as totacctbal
from (
        select
                substring(c_phone from 1 for 2) as cntrycode,
                c_acctbal
        from
                customer
        where
                substring(c_phone from 1 for 2) in
                        ('[I1]','[I2]','[I3]','[I4]','[I5]','[I6]','[I7]')
                and c_acctbal > (
                        select
                                avg(c_acctbal)
                        from
                                customer
                        where
                                c_acctbal > 0.00
                                and substring (c_phone from 1 for 2) in
                                        ('[I1]','[I2]','[I3]','[I4]','[I5]','[I6]','[I7]')
                )
                and not exists (
                        select
                                *
                        from
                                orders
                        where
                                o_custkey = c_custkey
                )
        ) as custsale
group by
        cntrycode
order by
        cntrycode;
```

2.4.22.3 Substitution Parameters

1. I1 … I7 are randomly selected without repetition from the possible values for Country code as defined in Clause 4.2.2.9.

2.4.22.4 Query Validation

For validation against the qualification database the query must be executed using the following substitution parameters and must produce the following output data:

1. I1 = 13.
2. I2 = 31.
3. I3 = 23.
4. I4 = 29.
5. I5 = 30.
6. I6 = 18.
7. I7 = 17.

2.4.22.5 Sample Output

CNTRYCODE	NUMCUST	TOTACCTBAL
13	888	6737713.99