

1985 Version 5
1988 Version 6SYBASE
1987 Version 1PostgreSQL
1988 PrototypeMicrosoft
SQL Server
1989MySQL
1994 First release

由于年代较久，DBMS 不能重构它们的架构。因为这会要求客户迁移大量的数据。

Mid 1980s looking like in the 1980s.

VAX 8600 (high-end)

32-bit architecture

Processor, ~10 to 20MHz clock

4 to 256 Mb of memory
Keep in mind that the big DBMS products were designed for this.

I/Os ~10 to 30 Mb/s

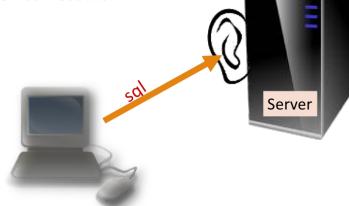
digital

Beginning of multi-processor computers



A 4 MB memory board for the VAX 8600

SQL queries will be directly sent to this server. The listener is only here for new connections.



In many cases (HTTP server, application server) the end user isn't directly talking to the DBMS server.



It doesn't make any difference for the DBMS.

```
select m.title, m.year_released
from movies m
inner join credits c
on c.movieid = m.movieid
inner join people p
on p.peopleid = c.peopleid
where p.first_name = 'Tim'
and p.surname = 'Burton'
and c.credited_as = 'D'
```

Syntax ✓
Do tables exist? ✓
Right to access? ✓
Do columns exist? ✓
Indexes we can use? Best way
to access data? ✓

② Another crucial phase is the one when

One way to improve efficiency is to keep the optimizer tries to determine the data dictionary information (meta-data) most efficient way to access data. in a shared cache to avoid additional queries.

Kept in memory

QUERY OPTIMIZER

- Logical transforms
- Indexes Volumes Storage
- Hardware performance
- System load
- Settings

The optimizer has to (or can) take into account a lot of factors.

① 保存 data dictionary 信息 (metadata) 在共享缓存中，以避免额外查询。

② 另一个关键阶段是优化器试图确定访问 data 最有效的方式。

PARSING takes time

解析需要时间，
因此不要对相同
查询解析多次。

Keep parsed queries in memory



As most applications run exactly the same SQL statements again and again, a DBMS will cache a parsed query for reuse. For MySQL, it will be cached for a session.

Query cache management
LRU

Least Recently Used

但是不可能将解析查询
保持在 cache 中长时间。
因此要管理缓存，用新查询
替换一段时间没有执行的
查询。

```
select m.title, m.year_released
from movies m
inner join credits c
on c.movieid = m.movieid
inner join people p
on p.peopleid = c.peopleid
where p.first_name = 'Tim'
and p.surname = 'Burton'
and c.credited_as = 'D'
```

Checksum

We primarily recognize identical queries by computing a text checksum.

+ check tables are same
and context identical

Scaling