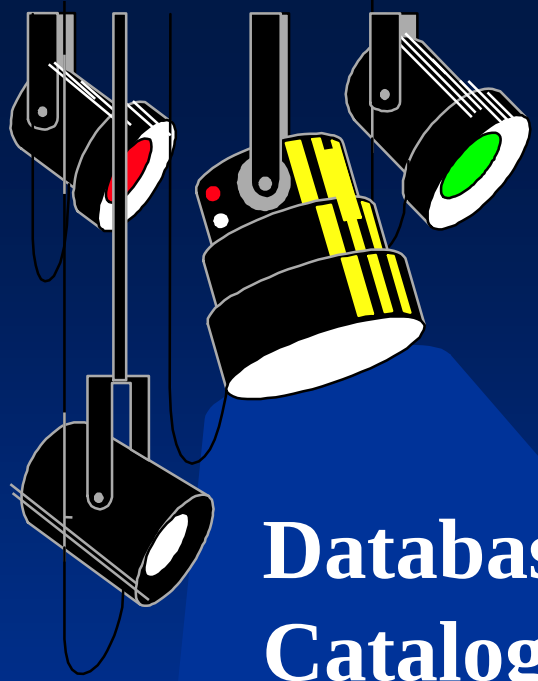


Database System Architecture and System Catalog

Outline (Ch. 17, 3rd ed. – Ch. 2, 4th ed., 5th ed., 6th ed., 7th ed.)

- Database System Architectures
- System Catalog
- System Catalog in Oracle

Database System Architectures

- Centralized DBMS

  Mainframe computer

  - DBMS functionality
  - Application program
  - User interfaces

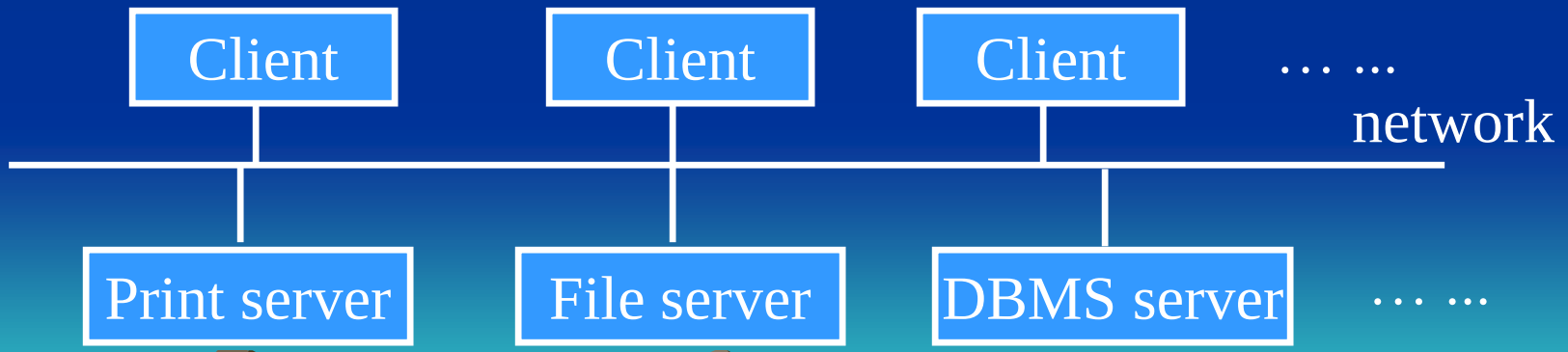
  Computer terminals

  - Input
  - Output

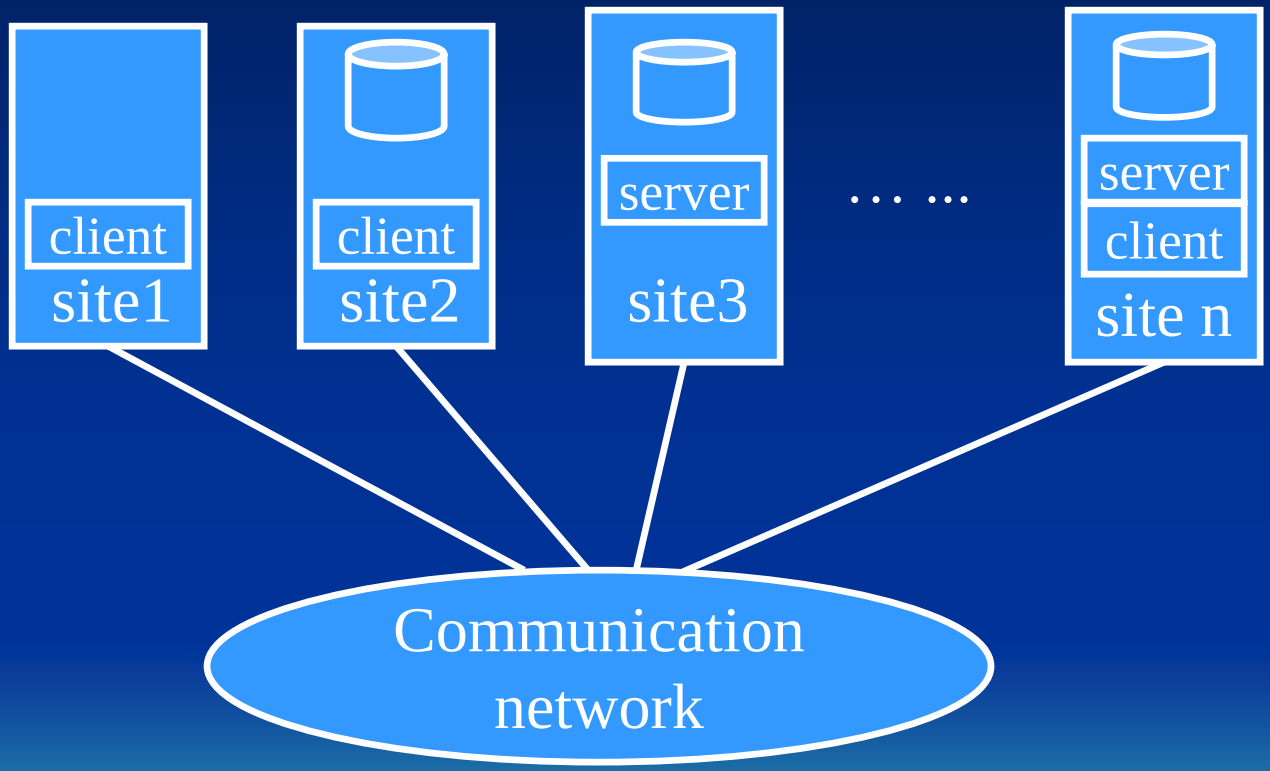
Database System Architectures

- Client-Server Computer Architecture
  - Terminals are replaced with PCs and workstations
  - Mainframe computer is replaced with specialized servers (with specific functionalities).

    File server, DBMS server, mail server, print server, …



Database System Architectures



Database System Architectures

- Client-Server Database Architecture
  - database client

    user interface, application programs
  - database server

    SQL language, transaction management
  - database connection

    ODBC - open database connectivity

    API - application programming interface

Database System Architectures

- Client-Server Architecture in DBMSs
  - database client

    user interface, data dictionary functions, DBMS interaction with programming language compiler, global query optimization, structuring of complex objects from the data in the buffers, ...
  - database server

    data storage on disk, local concurrency control and recovery, buffering and caching of disk storage, ...

Illustration for DBMS interaction with programming language compiler:

```
EXEC SQL DECLARE C1 CURSOR FOR
SELECT au_fname, au_lname FROM authors FOR BROWSE;
EXEC SQL OPEN C1;
while (SQLCODE == 0)
{
EXEC SQL FETCH C1 INTO :fname, :lname;
}
```

Catalog for Relational DBMSs

- Catalog - meta data for a relational schema
  - relation names, attribute names, attribute domains (data types)
  - description of constraints

    primary keys, secondary keys, foreign keys, NULL/NON-NULL, cardinality constraints, participation constraints, ...
  - views, storage structure, indexes
  - security, authorization, owner of each relation

Catalog for Relational DBMSs

- Catalog is stored as relations.

  (It can then be queried, updated and managed using DBMS software - SQL.)

REL_AND_ATTR_CATALOG

REL_NAME	ATTR_NAME	ATTR_TYPE	MEMBER_OF_PK	MEMBER_OF_FK	FK_RELATION
EMPLOYEE	FNAME	VSTR15	no	no	
…	…				
EMPLOYEE	SUPERSSN	STR9	no	yes	EMPLOYEE
EMPLOYEE	DNO	INTEGER	no	yes	DEPARTMENT
…	…				

EMPLOYEE

ssn	FNAME	LNAME	SUPERSSN	DNO	…. ….

DATABASE

System Catalog

Catalog for Relational DBMSs

- Catalog is stored as relations.

  (It can then be queried, updated and managed using DBMS software - SQL.)

RELATION_KEYS

REL_NAME	KEY_NUM	MEMBER_ATTR

RELATION_INDEXES

REL_NAME	INDEX_NAME	MEMBER_ATTR	INDEX_TYPE	ATTR_NO	ASC_DESC

VIEW_QUERIES

VIEW_NAME	QUERY

VIEW_ATTRIBUTES

VIEW_NAME	ATTR_NAME	ATTR_NUM

Works_on

ssn	Pno	hours
123456789	1	30
… …		

Employee

ssn	Dno	… …
123456789	1	… …
… …		

RELATION_KEY

REL_NAME	KEY_NUM.	MEMBER_ATTR
Works_on	1	ssn
Works_on	2	Pno
Employee	1	ssn
… …		

RELATION_INDEXES

REL_NAME	INDEX_NAME	MEMBER_ATTR	INDEX_TYPE	ATTR_NO	ASC_DESC
Works_on	I1	SSN	Primary	1	ASC
Works_on	I1	Pno	Primary	2	ASC
Works_on	I2	SSN	Clustering	1	ASC

Primary index:

Index file: I1
(<k(i), p(i)> entries)

Data file: Works_on

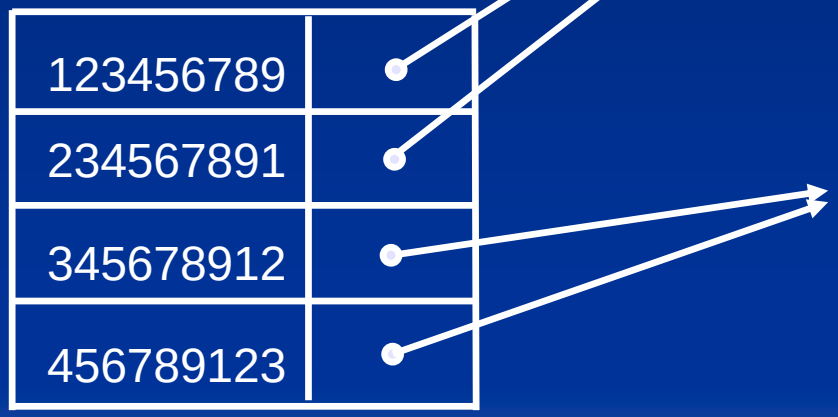
SSN	Pno	hours
123456789	1	…
123456789	2	
123456789	3	
234567891	**1**	
234567891	2	
345678912	2	
345678912	3	
456789123	1	
… …		

Query:

Is there a record with key = 234567891, 1?

Clustering index:

Index file: I2
(<k(i), p(i)> entries)



Data file: Works_on

SSN	Pno	hours
123456789	1	…
123456789	2	
123456789	3	
234567891	1	
234567891	2	
345678912	2	
345678912	3	
456789123	1	

… …

Create View Works_on1

AS Select FNAME, LNAME, PNAME, hours

From EMPLOYEE, PROJECT, WORKS_ON

Where ssn = essn and

Pno. = PNUMBER

VIEW_QUERIES

VIEW_NAME	QUERY

Works_on1 Select FNAME, LNAME, PNAME, hour

… …

VIEW_ATTRIBUTES

VIEW_NAME	ATTR_NAME	ATTR_NUM
Works_on1	FNAME	1
Works_on1	LNAME	2
Works_on1	PNAME	3
Works_on1	hours	4

Select FNAME, LNAME, PNAME
From Works_on1
Where FNAME = ‘David’ and LNAME = ‘Shepperd’

Select FNAME, LNAME, PNAME
From Works_on1
Where FNAME = ‘David’ and LNAME = ‘Shepperd’

Select FNAME, LNAME, PNAME
From EMPLOYEE, PROJECT, WORKS_ON

Where ssn = essn and

Pno. = PNUMBER and

FNAME = ‘David’ and LNAME = ‘Shepperd’

System Catalog in ORACLE

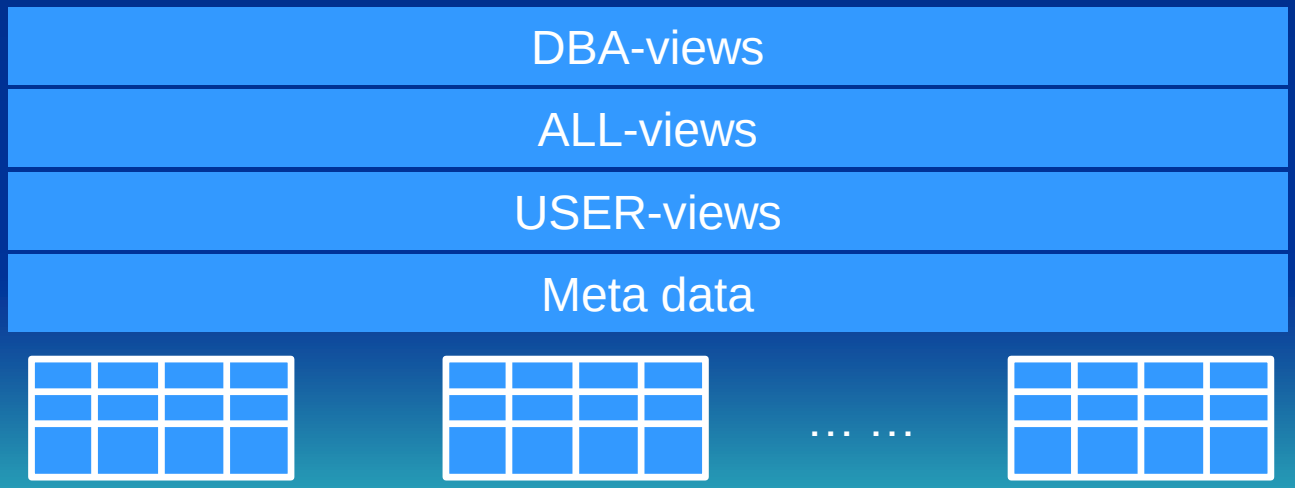
- Meta data - data dictionary:

  Information about schema objects: tables, indexes, views, triggers, ...

- Meta data are divided into three levels:
  - information for objects owned by a user
  - information for objects owned by a user as well as the objects that the user has been granted access to
  - information about all database objects

System Catalog in ORACLE

- Meta data are divided into three levels - three kinds of views:
  - view name prefixed with USER
  - view name prefixed with ALL
  - view name prefixed with DBA



•Example

```
SELECT *
FROM ALL_CATALOG
WHERE OWNER = 'SMITH'
```

Owner	TABLE	TABLE_TYPE
SMITH	ACCOUNT	TABLE
SMITH	CUSTOMERS	TABLE
SMITH	CUSTORDER	VIEW
SMITH	ORDERS	TABLE

System Catalog in ORACLE

- Example

  SELECT COLUMN_NAME, DATA_TYPE, DATA_LENGTH,
  NUM_DISTINCT, LOW_VALUE, HIGH_VALUE
  FROM ***USER_TAB_COLUMNS***
  WHERE TABLE_NAME = 'ORDERS'

COLUMN_NAME	DATA_TYPE	DATA_LENGTH	NUM_DISTINCT	LOW_VALUE	HIGH_VALUE
ORDERNO	NUMBER	22	4	C102	C105
CUSTNO	NUMBER	22	3	C102	C106
ORDERDATE	DATE	7	4	...	...

Change 'database statistics':
ANALYZE TABLE ORDERS
COMPUTE STATISTICS

System Catalog in ORACLE

- Example

  SELECT PCT_FREE, INITIAL_EXTENT, NUM_ROWS, BLOCK,
  EMPTY_BLOCKS, AVG_ROW_LENGTH
  FROM ***USER_TABLES***
  WHERE TABLE_NAME = 'ORDERS'

PCT_FREE	INITIAL_EXTENT	NUM_ROWS	BLOCK	EMPTY_BLOCK	AVG_ROW_LENGTH
10	10240	4	1	3	17

PCT_free: percertage of a block, which is left free
Initial_extent: initial allocation of space for a new table
A block: 2560 bytes

System Catalog in ORACLE

- Example

  SELECT INDEX_NAME, UNIQUENESS, BLEVEL, LEAF_BLOCKS, DISTINCT_KEYS, AVG_LEAF_BLOCKS_PER_KEY, AVG_DATA_BLOCKS_PER_KEY
  FROM ***USER_INDEXES***
  WHERE TABLE_NAME = 'ORDERS'

INDEX_ NAME	UNIQUENESS	BLEVEL	LEAF_ BLOCK	DISTINCT_ KEYS	AVG_LEAF_ BLOCKS_ PER_KEY	AVG_DATA_ BLOCK_ PER_KEY
ORD_ CUSTNO	NONUNIQUE	0	1	3	1	1

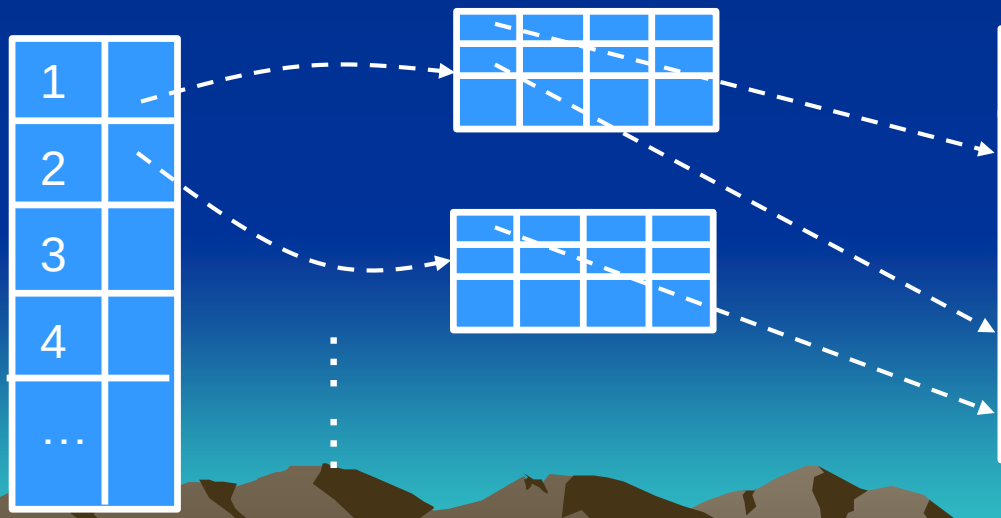
Index: ORD-CUSTNO

CustNo	Record-pointer
1	1, 3
2	4
3	2
…	

ORDERS

OrderNo	CustNo	OrderDate
1	1	… …
2	3	… …
3	1	… …
4	2	… …

Multi-level index:



ORDERS

OrderNo	CustNo	OrderDate
1	1	… …
2	3	… …
3	1	… …
4	2	… …

- AVG_LEAF_BLOCKS_PER_KEY: Average number of leaf blocks in which each distinct value in the index appears, rounded up to the nearest integer.

- AVG_DATA_BLOCKS_PER_KEY: Average number of data blocks (in the data file), in which each distinct value (in the index) appears, rounded up to the nearest integer.
  - Let say that the value of AVG_DATA_BLOCKS_PER_KEY is 3. So we have to visit 3 different data blocks in order to get all data which belong to a desirable index value on average.

System Catalog in ORACLE

- Example

  SELECT *
  FROM ***USER_VIEWS***
  WHERE OWNER = ‘SMITH

VIEW_NAME	TEXT_LENGTH	TEXT
CUSTORDER	101	select custname, city, orderno, orderdate from customers, orders where customers.custno = orders.custno

System Catalog in ORACLE

- Example

  SELECT COLUMN_NAME, DATA_TYPE, DATA_LENGTH
  FROM ***USER_TAB_COLUMN***
  WHERE TABLE_NAME = ‘CUSTORDER’

COLUMN_NAME	DATA_TYPE	DATA_LENGTH
CITY	CHAR	20
ORDERNO	NUMBER	22
ORDERDATE	DATE	7
CUSTNAME	CHAR	20

System Catalog in ORACLE

- DBMS software modules accessing the meta data
  1. DDL (SDL) compilers
     These DBMS modules process and check the specification of a database schema in the data definition language (DDL) and the specification in the storage definition language (SDL), and store these descriptions in the catalog.
  2. Query and DML parser and verifier
     These modules parse queries, DML retrieval statements, and database update statements; they also check the catalog to verify whether all the schema names referenced in these statements are valid.

System Catalog in ORACLE

- DBMS software modules accessing the meta data

  3. Query and DML compilers

     These compilers convert high-level queries and DML commands into low-level file access commands. The mapping between the conceptual schema and the internal schema file structures is accessed from the catalog during this process.

  4. Query and DML optimizer

     The query optimizer accesses the catalog for access path, implementation information, and data statistics to determine the best way to execute a query or a DML command.

System Catalog in ORACLE

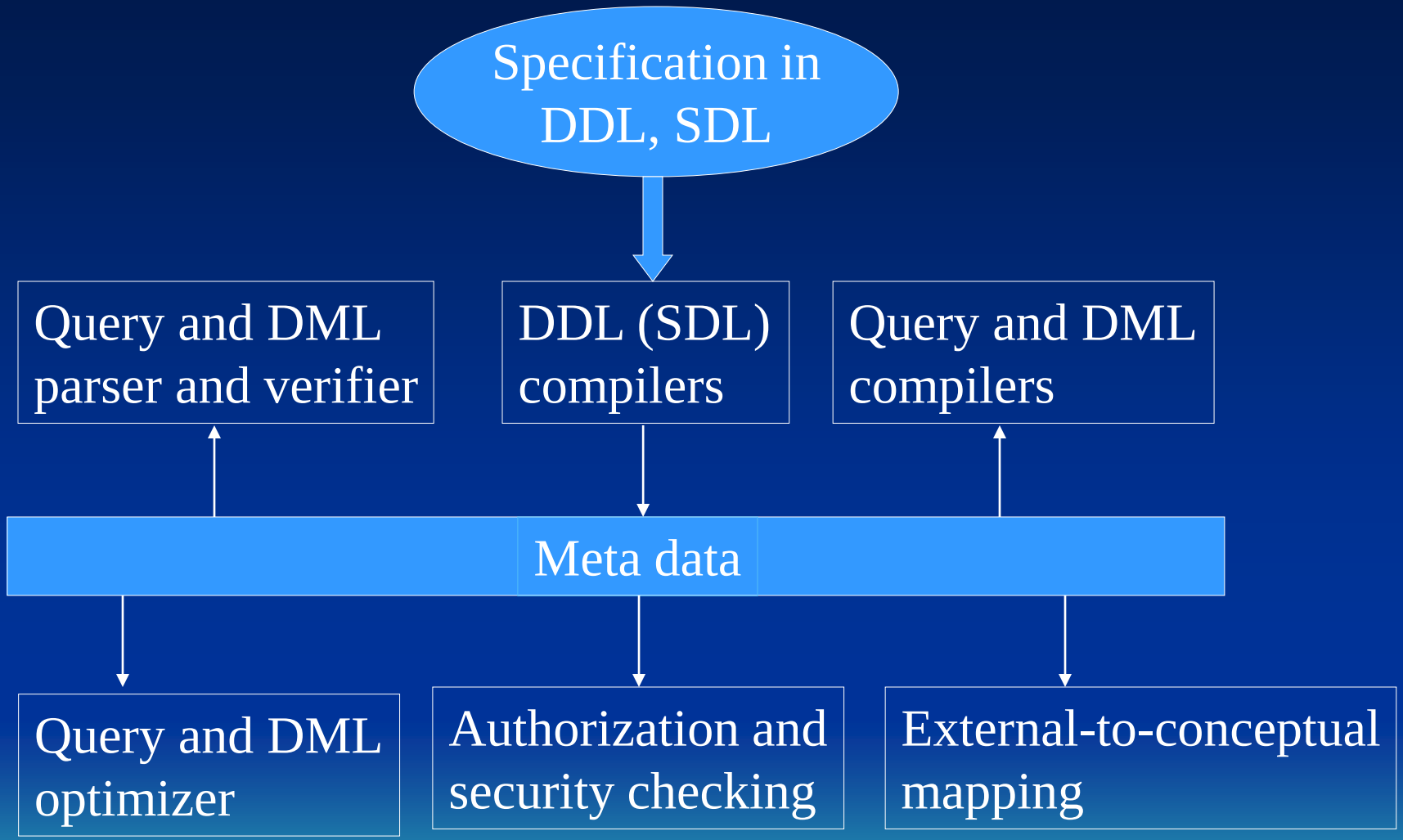
- DBMS software modules accessing the meta data

  5. Authorization and security checking

     The DBA has privileged commands to update the authorization and security portion of the catalog. All access by a user to a relation is checked by the DBMS for proper authorization by accessing the catalog.

  6. External-to-conceptual mapping of queries and DML commands

     Queries and DML commands specified with reference to an external view or schema must be transformed to refer to the conceptual schema before they can be accessed by the DBMS. It needs to access the catalog description of the view.


Specification in DDL, SDL
Query and DML parser and verifier
DDL (SDL) compilers
Query and DML compilers
Meta data
Query and DML optimizer
Authorization and security checking
External-to-conceptual mapping

Create View Works_on1

AS Select FNAME, LNAME, PNAME, hours

From EMPLOYEE, PROJECT, WORKS_ON

Where ssn = essn and

Pno. = PNUMBER

Select FNAME, LNAME, PNAME
From Works_on1
Where FNAME = ‘David’ and LNAME = ‘Shepperd’

Select FNAME, LNAME, PNAME
From EMPLOYEE, PROJECT, WORKS_ON

Where ssn = essn and

Pno. = PNUMBER and

FNAME = ‘David’ and LNAME = ‘Shepperd’