1.Does 3 level architecture solve all the problems?

A.) **NO.** It's implementation is a much difficult compared to 2-tier architecture. There are few considerable problems with the 3-tier architecture:-

i) The changes to the data structure are difficult to make, considering the physical data independence that should be maintained at view level and conceptual level.

ii) Application code becomes complex since it must deal with details.

iii) Implementation of new features to address the scalability and security problems becomes difficult at physical level.

iv) The design of schema at conceptual level considering the different constraints for integrity and maintenance of schema without any information of physical level makes it difficult.

2.How can 3 level architecture be implemented? What transformations and data structures are needed?

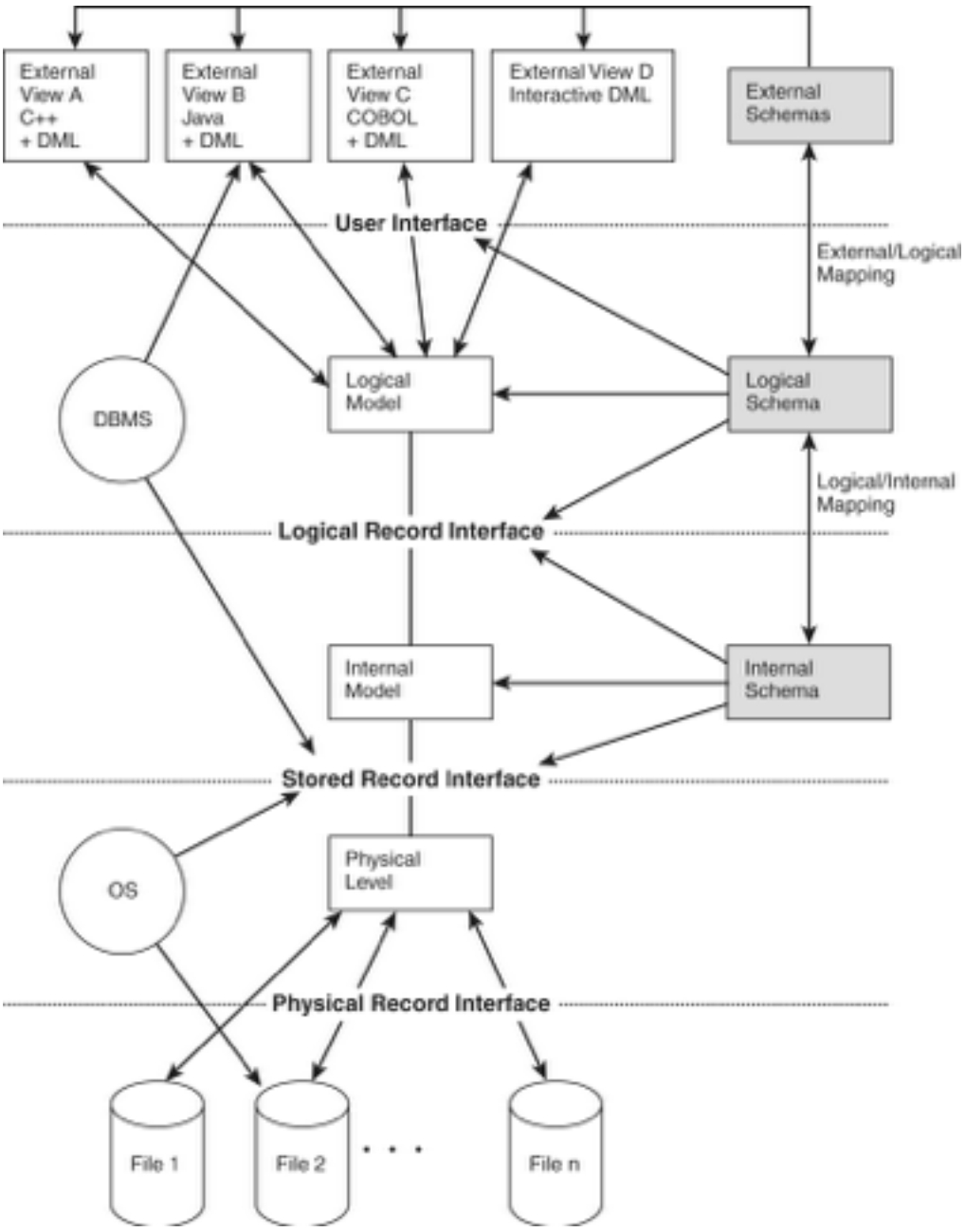
A.) **PHYSICAL LEVEL:-** The physical level of data deals with how the data is stored in physical storage. It deals with files, indices, record storage and placement. It also includes data compression and encryption.

The data structures used for efficient retrieval and storage are B-Trees and B+ Trees. Earlier at this level they used to deal with minimising physical distances between related data and organising the data structures within the file (blocked records, linked lists of blocks, etc.)

The advantage in 3-level architecture is that we can change the entire physical implementation of data owing to scalability and efficiency issues without affecting the other two levels.

CONCEPTUAL LEVEL:- The conceptual level deals with the logical structure of data including the dependencies, schema, constraints on data, etc. It can be implemented in any query language or even programming language. We can define the structure of data (schema) at the conceptual level.

EXTERNAL LEVEL:- This is the view of the database which limits the users or groups of users to view only specific parts of schema, i.e. subschema.



3.Does 3 level architecture support more than 1 programming language?

A.) The entire 3-level architecture can be implemented in any high level languages like C or C++. Yes of course the 3-level architecture do support wide range of languages.

The physical level, its B+ trees, indices and all such data structures are implemented in C or any other high level language.

The conceptual model including the DDL and DML can be implemented in C or C++ and the external level can be implemented in any language.

But the advantage of 3 level architecture can be implemented in many different languages, i.e. each level can be implemented in different language because 3-level architecture supports data independence.

4. How is data independence achieved?

The ability to modify a schema definition in one level without affecting a schema definition in a higher level is called data independence.

1. There are two kinds:

Logical data independence

i) The ability to modify the conceptual schema without causing application programs to be rewritten.

ii) Immunity of external schemas to changes in the conceptual schema.

iii) Usually done when logical structure of database is altered

Physical data independence

i) The ability to modify the internal scheme without having to change the conceptual or external schemas.

ii) Modifications at this level are usually to improve performance.