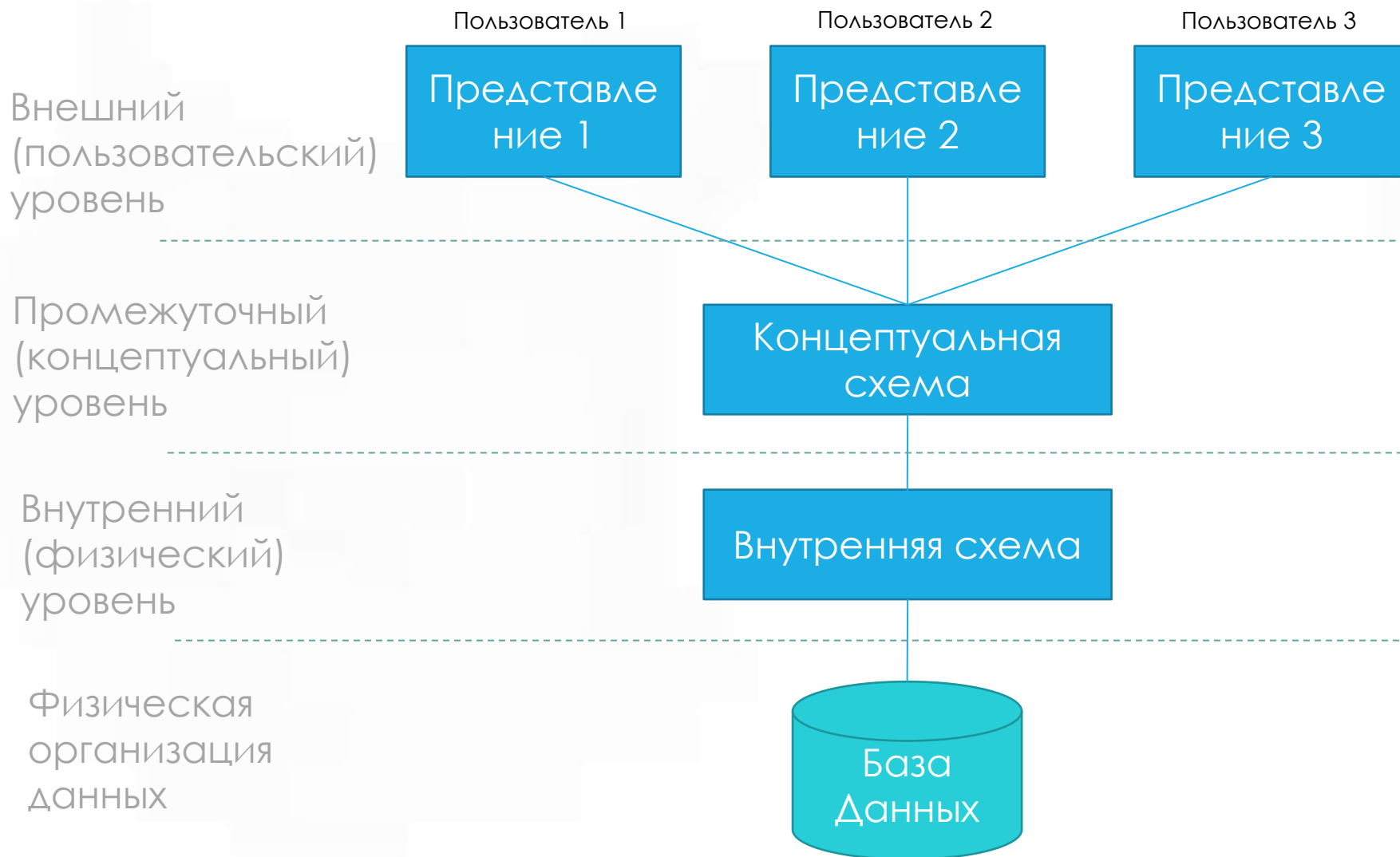


A light beige rectangular tab is positioned at the top center of the white text box.

ANSI-SPARC



# Архитектура ANSI-SPARC

ANSI – SPARC (American National Standards Institute – Standards Planning and Requirement Committee) - представляет собой 3 уровня описания элементов данных – внешний, концептуальный и внутренний. Цель трехуровневой архитектуры заключается в отделении пользовательского представления БД от ее физического представления.

# Внешний уровень

– отвечает за представление данных с точки зрения пользователей. Он состоит из нескольких различных внешних представлений БД.

Каждый пользователь имеет дело с представлением предметной области в наиболее удобной для него форме. Внешнее представление данного пользователя содержит только те сущности, атрибуты и связи, которые ему интересны. Он может ничего не знать о других сущностях, содержащихся в базе. Помимо этого, различные представления могут по-разному отображать одни и те же данные. Например, один пользователь может просматривать даты в формате (день, месяц, год), а другой - в формате (год, месяц, день). Некоторые представления могут содержать производные или вычисляемые данные, которые не хранятся в БД, а создаются по мере надобности.

# Концептуальный уровень

– содержит логическую структуру всей БД с точки зрения ее администратора. Описывает то, какие данные содержатся в БД, а также связи между ними.

На концептуальном уровне представлены следующие компоненты:

- все сущности, их атрибуты и связи;
- ограничения, накладываемые на данные;
- семантическая информация о данных;
- информация о мерах обеспечения безопасности и поддержки целостности данных

Концептуальный уровень поддерживает каждое внешнее представление, в том смысле, что любые доступные пользователю данные должны содержаться (или могут быть вычислены) на этом уровне. Однако этот уровень не содержит никаких сведений о методах хранения данных.

# Внутренний уровень

– отвечает за физическое представление БД в памяти компьютера. Этот уровень описывает, как информация хранится в БД, ее физическую реализацию и предназначен для достижения оптимальной производительности и обеспечения экономного использования дискового пространства.

Он содержит описание структур данных и организации отдельных файлов, используемых для хранения данных на запоминающих устройствах.

На внутреннем уровне хранится следующая информация:

- распределение дискового пространства для хранения данных и индексов.
- сведения о размещении записей.
- сведения о сжатии данных и выбранных методах их шифрования.

# Схемы базы данных

На самом высоком уровне имеется несколько внешних схем или подсхем, которые соответствуют разным представлениям данных.

На концептуальном уровне описание базы данных называют концептуальной схемой, которая описывает все элементы данных и связи между ними, с указанием необходимых ограничений поддержки целостности данных.

На самом низком уровне абстракции находится внутренняя схема, которая является полным описанием внутренней модели данных. Она содержит определения хранимых записей, методы представления, описания полей данных, сведения об индексах и выбранных схемах хеширования.

Система управления базами данных отвечает за установление соответствия между этими тремя типами схем, а также за проверку их непротиворечивости.

# Описание базы данных

– схема базы данных. Схема создается в процессе проектирования базы данных, причем предполагается, что она изменяется достаточно редко. Однако содержащаяся в базе данных информация может меняться часто - например, всякий раз при вставке о том или ином объекте. Совокупность информации, хранящейся в базе данных в любой определенный момент времени, называется состоянием базы данных. Следовательно, одной и той же схеме базы данных может соответствовать множество ее различных состояний.

Схема базы данных иногда называется содержанием базы данных, а ее состояние - детализацией.

# Независимость от данных

Логическая независимость от данных означает полную защищенность внешних схем от изменений, вносимых в концептуальную схему. Такие изменения концептуальной схемы, как добавление или удаление новых сущностей, атрибутов или связей, должны осуществляться без внесения изменений в уже существующие внешние схемы или переписывания прикладных программ.

Физическая независимость от данных означает защищенность концептуальной схемы от изменений, вносимых во внутреннюю схему. Такие изменения внутренней схемы, как использование различных файловых систем или структур хранения, разных устройств хранения, модификация индексов или хеширование, должны осуществляться без необходимости внесения изменений в концептуальную или внешнюю схемы. Пользователем могут быть замечены изменения только в общей производительности системы.