



Введение

В наши дни невозможно представить работу какой-либо организации или предприятия без информационной системы и базы данных. Все большую актуальность приобретают освоение принципов построения и применения программных продуктов и технологий, таких как СУБД, средства администрирования и защиты БД и CASE-систем автоматизации проектирования.

Ранее базы данных использовались в большей мере на больших серверах, а сейчас, когда персональный компьютер доступен каждому, высокоэффективная обработка данных также стала доступна каждому. В наше время базы данных облегчают обработку огромного потока информации, с которым мы сталкиваемся каждый день.

Базы данных необходимы, так как воспользовавшись ими, мы с лёгкостью находим доступную и точную информацию по своим интересам. Еще не менее важный фактор: постоянное обновление данных.

Таким образом, невозможно представить как нашу современную жизнь без информационных технологий и баз данных, так и отдельные ее сферы.

Основные понятия

База данных – это поименная совокупность структурированных данных, относящихся к определенной предметной области.

Система управления базами данных (СУБД) – это совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования базы данных многими пользователями.

Автоматизированная информационная система (АИС) – это система, реализующая автоматизированный сбор, обработку, манипулирование данными, функционирующими на основе ЭВМ и включающими соответствующее ПО и персонал.

Банк данных – это система, специальным образом организованных данных, баз данных, программных, технических, языковых, организационно-методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования данных.

Приложения – это отдельные программы или комплекс программ, реализующих автоматизацию решения прикладных задач обработки данных.

Факторы, предопределившие направление развития технологий БД

Факторы, предопределившие направление развития технологий БД:

1. Обеспечение надёжности хранения информации;
2. Снижение трудоемкости обработки информации;
3. Возможность независимости сбора и актуализации данных.

База данных, как и любой программный продукт, обладает жизненным циклом. База данных является фундаментальным компонентом информационной системы, поэтому жизненный цикл БД связан и жизненным циклом информационной системы организации.

Жизненный цикл БД – это период времени, в котором протекают процессы проектирования, реализации, поддержания системы базы данных, а также ее изъятие из эксплуатации.

Этапы проектирования БД

1-й этап – это этап формулирования и анализа требований. На этом этапе устанавливаются цели организации, определяются требования к БД. Все требования документируются в форме, доступной конечному пользователю и проектировщику БД.

2-й этап – это этап концептуального проектирования. На этом этапе происходит описание и синтез информационных требований пользователя к первоначальному проекту БД. Результат этого этапа – появление высокоуровневого представления (в виде таблиц) информационных требований пользователя.

3-й этап – этап логического проектирования. На этом этапе высокоуровневые представления данных преобразуются в структуру используемой СУБД. Основной целью этого этапа является устранение избыточности данных с использованием специальных правил нормализации. Сначала выбирается модель БД, затем с

помощью ЯОД (язык описания данных) создается структура БД, которая заполняется данными с помощью команд ЯМД (язык манипулирования данными), систем меню, экранных форм или в режиме просмотра таблиц БД. Здесь же обеспечивается защита и целостность данных с помощью СУБД.

4-й этап – этап физического проектирования. Здесь решаются вопросы, связанные с производительностью системы, а также определяются структуры хранения данных и виды доступа.

Этапы жизненного цикла БД

1. Предварительное планирование. В процессе планирования собирается информация по использованным прикладным программам и их функциям; по файлам, связанных с приложениями; Данная информация помогает определить будущие требования к системе БД.
2. Проверка осуществимости. На этом этапе определяется технологическая, экономическая и операционная осуществимость плана БД:

а) Технологическая осуществимость поясняет, имеется ли технология создания запланированной БД.

б) Экономическая осуществимость показывает издержки и выгоду от создания БД.

в) Операционная осуществимость показывает, имеются ли средства и эксперты, с помощью которых возможно успешное создание БД.

Определение требований. Этот этап включает в себя выбор цели БД, выяснение требований к оборудованию и ПО. Таким образом, на данном этапе создаётся общая информационная модель, которая выражается в следующих задачах:

а) Определяются цели системы путём анализа информационных потребностей. Здесь также обязательно указывается, какую именно БД следует создавать и какие коммуникационные средства необходимы. Выходной документ – комментарий, описывающий цели системы.

б) Определение пользовательских требований: документация в виде общей информации (комментарии и отчёты); фиксация функций системы и определение прикладных систем, которые будут выполнять эти требования. Данные представляются в виде соответствующих документов.

в) Определение общих требований к оборудованию и программному обеспечению, связанных с поддержанием желаемого уровня быстродействия. Данная информация используется для выбора типов компьютеров и СУБД, объема дисков, количества принтеров. Данные этого этапа излагаются в отчете, содержащем примерные конфигурации оборудования и ПО.

Модели жизненного цикла

Модель жизненного цикла – структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении всего жизненного цикла БД. Важным компонентом в проектировании БД является тестирование системы в процессе ее разработки. На такое способны не все модели жизненного цикла.

Базовые модели жизненного цикла:

- Модель каскад («водопад») Данная модель представляет линейную последовательность прохождения стадий создания программного продукта.

Достоинства данной модели:

- законченный набор проектной документации, отвечающий критериям согласованности;
- все стадии выполняются строго друг за другом, что позволяет разработчикам уверенную планировку сроков разработки и денежных средств;

Недостатки данной модели:

- жесткая схема процесса разработки;
- жизненный цикл строго очерчен и определен на стадиях разработки программного продукта;
- результат разработки программного продукта известен заказчику в самом конце проекта, что значительно влияет на соответствие требованиям программного продукта;
- Инкрементная модель. Данная модель представляет линейную структуру стадий, но с несколькими версиями, что улучшает программный продукт.

Достоинства и недостатки у данной модели такие же, как и у предыдущей, но в отличие от модели каскад, заказчик может увидеть результаты раньше, тем самым изменяя начальные требования, что повысит соответствие программного продукта требованиям заказчика. Данная модель предназначена для выполнения разработки сложных комплексных систем, и заказчик и разработчик должны четко представлять готовый программный продукт.

- Спиральная модель («эволюционная», разработана Барри Боэмом, 1988 г.,. Данная модель представляет разработку последовательных версий, при этом в самом начале указаны все требования.

Данная модель создана для разработки новаторских систем. В начале разработки у заказчика и разработчика нет четкого видения конечного программного продукта, в связи с рисками.

Недостатки модели:

- увеличивается неопределенность у разработчика в перспективах разработки продукта.
- трудности в планировании времени и ресурсов всего программного продукта. Чтобы устранить этот недостаток, достаточно ограничить во времени стадии жизненного цикла.

Достоинства модели:

- позволяет на ранних стадиях предоставить пользователю рабочий программный продукт для внесения возможных изменений и дополнений.
- допускается изменение начальных требований при разработке программы.
- позволяет полностью управлять проектом.
- позволяет разработать очень надёжную систему: по мере развития обнаруживаются и исправляются ошибки на каждом шаге.
- совершенствует процесс разработки: анализ, сделанный на каждом шаге, позволяет оценивать изменения в организации разработки, а также проводить улучшения по анализу.

- заказчик может завершить процесс разработки продукта без огромного ущерба для своих финансов.

Самая короткая модель разработки — это спиральная модель, а самая длинная модель разработки - каскадная. Каскадная модель применяется при разработке небольших проектов и при решении отдельных задач.

Теория нормализации

Проектирование реляционных баз данных связано с понятием нормализации. Эта теория основана на анализе зависимости функций между атрибутами отношений. Цель этой замечательной теории состоит в том, чтобы устранить избыточные данные из базы данных. Нормализация помогает значительно снизить количество хранимой информации и устранить неполадки в организации хранения данных.

Нормализация имеет различные уровни. В реляционной базе данных под основой понимается приведение модели к определенному уровню нормализации.

Нормализация достигается так: проверяют, соответствуют ли таблицы ряду условий, определенных в трех главных уровнях нормализации: первой, второй и третьей нормальных формах. Поговорим о них подробнее.

Первая нормальная форма

Эта форма требует, чтобы каждое поле таблицы было неделимым (содержащиеся в нем значения не должны делиться на более меньшие) и не содержало повторяющихся групп.

Вторая нормальная форма

Эта форма требует, чтобы все таблицы напрямую зависели от первичного ключа. Если в таблице имеется зависимость не ключевых полей от части первичного ключа, то следует вынести их в отдельную таблицу, причем сделать первичным ключом новой таблицы часть первичного ключа, от которой зависят поля и установить связь «один ко многим» от новой таблицы к старой.

Третья нормальная форма

Эта же форма требует, чтобы в таблицах не было транзитивных зависимостей между не ключевыми полями.

Результат нормализации – модель данных, не содержащая неопределенностей в данных и повторений данных, а также ее легко поддерживать.

Метод интеллектуального штурма

Этот метод преследует цель обнаружить ошибки проекта на ранних этапах жизненного цикла БД (чем раньше может быть выявлена ошибка, тем меньше придется платить за ее преодоление). Соответственно для проведения экспертизы формируются бригады экспертов, которые включают ведущих специалистов из групп: администраторов БД, разработки ПО, системных программистов, обслуживающего персонала, ведущих специалистов предметных областей создания БД и соответственно назначается руководитель бригады экспертов (как правило, администратор БД). Заседание экспертов целесообразно проводить не менее 4-х раз в течение жизненного цикла БД. Во-первых, после этапа концептуального проектирования, во-вторых, после реализации, в-третьих, перед началом эксплуатации системы, в-четвертых, в процессе эксплуатации системы когда обнаружены ошибки проекта.

Заключение

Таким образом, можно утверждать, что базы данных остаются динамично развивающимся теоретико-прикладным направлением в нашей жизни, при этом особенно прогрессирует прикладная сторона. В данной работе нами были рассмотрены основные понятия и свойства моделей жизненного цикла баз данных, их достоинства и недостатки.

Список источников

1. https://spravochnick.ru/bazy_dannyh/zhiznennyy_cikl_bd/#etapy-zhiznennogo-cikla-bd
2. https://mf.grsu.by/Kafedry/kaf001/academic_process/048/46
3. <https://studizba.com/lectures/10-informatika-i-programmirovanie/311-konspekt-lekciy-po-bazam-dannyh/4158-3-proektirovanie-baz-dannyh.html>
4. <https://pandia.ru/text/78/294/69801.php>