

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ПРИМОРСКОГО КРАЯ

краевое государственное автономное профессиональное образовательное учреждение

«Лесозаводский индустриальный колледж»

по МДК 05.02 специальность 09.02.07 «Информационные системы и
программирование»

Контакты преподавателя: maya_tok@mail.ru

Задание: составить конспект лекции, письменно ответить на вопросы к лекциям, ответы прислать на почту 28.10 до 22.00

Лекция №3-4_ Структура CASE-средства. Структура среды разработки.

Основные возможности.

Структура CASE-средства

С темой о выборе CASE – средства для проектирования ИС мы уже знакомы (см.лекции по МДК 05.01).

Сегодня мы подробнее остановимся на том, из каких элементов состоит любое CASE – средство.

CASE-технология (Computer – Aided Software Engineering, т.е. автоматизированная разработка ПО с помощью компьютерных технологий).

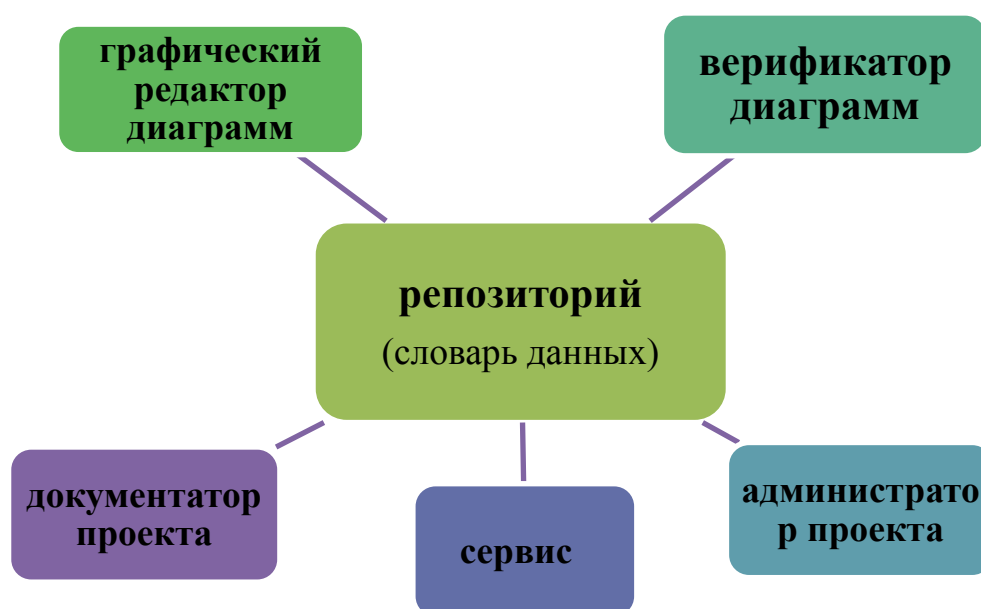
В настоящее время под термином CASE-средства понимаются программные средства, поддерживающие процессы создания и сопровождения ИС, включая анализ и формулировку требований, проектирование прикладного ПО (приложений) и баз данных, генерацию кода, тестирование, документирование, обеспечение качества, конфигурационное управление и управление проектом, а также другие процессы.

В CASE-модели фаза прототипирования заменяет традиционную фазу системного анализа. Необходимо отметить, что наиболее автоматизируемыми фазами являются фазы контроля проекта и кодогенерации.

CASE-средства вместе с системным ПО и техническими средствами образуют полную среду разработки ИС.

Интегрированное CASE-средство (или комплекс средств, поддерживающих полный ЖЦ ПО) содержит следующие компоненты:

- **репозиторий**, являющийся основой CASE-средства. Он должен обеспечивать хранение версий проекта и его отдельных компонентов, синхронизацию поступления информации от различных разработчиков при групповой разработке, контроль метаданных на полноту и непротиворечивость;
- **графические средства анализа и проектирования**, обеспечивающие создание и редактирование иерархически связанных диаграмм (DFD, ERD и др.), образующих модели ИС;
- **средства разработки приложений, включая языки 4GL и генераторы кодов**;
- **средства конфигурационного управления**;
- **средства документирования**;
- **средства тестирования**;
- **средства управления проектом**;
- **средства реинжиниринга**.



Интегрированная среда разработки. Основные возможности и назначение.

Интегрированная среда разработки, ИСР (англ. IDE, Integrated Development Environment или Integrated Debugging Environment) — система программных средств, используемая программистами для разработки программного обеспечения (ПО).

Обычно среда разработки **включает в себя:**

- текстовый редактор;
- компилятор и / или интерпретатор;
- средства автоматизации сборки;
- отладчик.

Основным окном является текстовый редактор, который используется для ввода исходного кода в ИСР и ориентирован на работу с последовательностью символов в текстовых файлах. Такие редакторы обеспечивают расширенную функциональность — подсветку синтаксиса, сортировку строк, шаблоны, конвертацию кодировок, показ кодов символов и т. п. Иногда их называют редакторами кода, так как основное их предназначение — написание исходных кодов компьютерных программ.

Подсветка синтаксиса — выделение синтаксических конструкций текста с использованием различных цветов, шрифтов и начертаний. Обычно применяется в текстовых редакторах для облегчения чтения исходного текста, улучшения визуального восприятия. Часто применяется при публикации исходных кодов в Интернет.

Иногда содержит также средства для интеграции с системами управления версиями и разнообразные инструменты для упрощения конструирования графического интерфейса пользователя. Многие современные среды разработки также включают браузер классов, инспектор объектов и диаграмму иерархии классов — для использования при объектно-ориентированной разработке ПО.

Хотя и существуют ИСР, предназначенные для нескольких языков программирования — такие, как Eclipse, NetBeans, Embarcadero RAD Studio, Qt Creator или Microsoft Visual Studio, но обычно ИСР предназначается для одного определённого языка программирования - как, например, Visual Basic, PureBasic, Delphi, Dev-C++.

Частный случай ИСР, их эволюционное развитие — среды визуальной разработки, которые включают в себя возможность визуального редактирования интерфейса программы.

Основной единицей управления исходным кодом, используемой в различных IDE является проект. Как правило, также поддерживается некоторый способ объединения родственных проектов в группы.

Интегрированные среды разработки также часто поддерживают пометки в комментариях в исходном тексте программ, отмечающие места, требующие дальнейшего внимания или предполагающие внесение изменений, такие как TODO. В дальнейшем эти пометки могут выделяться редакторами (напр. vim, emacs, встроенный редактор Visual Studio) или использоваться для организации совместной работы с построением тегов и задач (например, в IntelliJ). Использование комментариев с TODO так же является стандартом оформления кода на Object Pascal, Delphi.

Обычно интегрированная среда разработки - это совокупность программных средств, поддерживающая все этапы разработки программного обеспечения от написания исходного текста программы до ее компиляции и отладки, и обеспечивающая простое и быстрое взаимодействие с другими инструментальными средствами (программным отладчиком-симулятором, внутрисхемным эмулятором, эмулятором ПЗУ и программатором).

Строго говоря, интегрированные среды разработки не относятся к числу средств отладки. Отладка — лишь одно из свойств интегрированных сред, которые представляют собой основу любой визуальной среды разработки или RAD-среды.

При традиционном подходе, начальный этап написания программы строится следующим образом:

1. Исходный текст набирается при помощи какого-либо текстового редактора.
2. По завершении набора, работа с текстовым редактором прекращается и запускается кросс компилятор.
3. Как правило, вновь написанная программа содержит синтаксические ошибки, и компилятор сообщает о них на консоль оператора.

4. Вновь запускается текстовый редактор, и оператор должен найти и устранить выявленные ошибки, при этом сообщения о характере ошибок выведенные компилятором уже не видны, так как экран занят текстовым редактором.

И этот цикл может повторяться не один раз. Если программа имеет большой объем, собирается из различных частей, и подвергается длительному редактированию или модернизации, то даже этот начальный этап может потребовать много сил и времени. После этого наступает этап отладки программы и к редактору с компилятором добавляется эмулятор или симулятор, за работой которого хотелось бы следить прямо по тексту программы в текстовом редакторе.

Интегрированные среды (оболочки) разработки (Integrated Development Environment, IDE) позволяют избежать большого объема однообразных действий и тем самым существенно повысить эффективность процесса разработки и отладки позволяют, то есть они являются RAD-средами различной степени автоматизации процесса программирования.

Работа в интегрированной среде дает программисту:

- возможность использования встроенного многофайлового текстового редактора, специально ориентированного на работу с исходными текстами программ;
- иметь автоматическую диагностику выявленных при компиляции ошибок, когда исходный текст программы, доступный редактированию, выводится одновременно с диагностикой в многооконном режиме;
- возможность параллельной работы над несколькими проектами. Менеджер проектов позволяет использовать любой проект в качестве шаблона для вновь создаваемого проекта;
- минимум перекомпиляции. Ей подвергаются только редактировавшиеся модули;
- возможность загрузки отлаживаемой программы в имеющиеся средства отладки, и возможность работы с ними без выхода из оболочки;
- возможность подключения к оболочке практически любых программных средств.

В последнее время, функции интегрированных сред разработки становятся стандартной принадлежностью программных интерфейсов эмуляторов и отладчиков-симуляторов.

Подобные функциональные возможности, в сочетании с дружелюбным интерфейсом, в состоянии существенно увеличить скорость разработки программ, особенно для микроконтроллеров и процессоров цифровой обработки сигналов, являющихся очень трудоемкими и труднообозримыми процессами.

Задания по теме:

1. Дать определение понятия транслятора, компилятора, отладчика, интерпретатора.
2. В чем заключается основной процесс отладки? (дать развернутый ответ).
3. Дайте определение системы управления версиями проекта. Приведите несколько основных терминов с определениями.
4. Опишите следующие ИСР: Eclipse, KDevelop, Microsoft Visual Studio по плану:
 - а) кто разработал, в каком году
 - б) что лежит в основе исл
 - с) на каком языке программирования написана
 - д) основные функции
 - е) отличия и особенности
5. Приведите 5 примеров ИСР (кроме тех, которые упоминались в лекции) и кратко их опишите.