**Лекция №6**

**Методология и технология разработки информационных систем**

**Стандарты и методики**

Одним из важных условий эффективного использования информационных технологий является внедрение корпоративных стандартов. Корпоративные стандарты представляют собой соглашение о единых правилах организации технологии или управления. При этом за основу корпоративных стандартов могут приниматься отраслевые, национальные и даже международные стандарты.  
Однако динамика развития информационных технологий приводит к быстрому устареванию существующих стандартов и методик разработки информационных систем. Так, в связи со значительным прогрессом в области программного обеспечения и средств вычислительной техники наблюдается рост размеров и сложности информационных систем. При этом существенно меняются требования как к основным функциям и сервисным возможностям систем, так и к динамике изменения этих функций. В этих условиях применение классических способов разработки и обеспечения качества информационных систем становится малоэффективным и не приводит к уровню качества, адекватному реальным требованиям.  
Полезны в этом отношении стандарты открытых систем (в первую очередь, стандарты на интерфейсы различных видов, включая лингвистические, и на протоколы взаимодействия). Однако разработка систем в новых условиях требует также новых методов проектирования и новой организации проектных работ. Проектирование и методическая поддержка разработки информационных систем, включая программное обеспечение и базы данных, традиционно поддерживаются многими стандартами и фирменными методиками. Вместе с тем известно, что требуется адаптивное планирование разработки, в том числе в динамике процесса ее выполнения. Одним из способов адаптивного проектирования является разработка и применение профилей жизненного цикла информационных систем и программного обеспечения. Корпоративные стандарты образуют целостную систему, которая включает три вида стандартов:

* на продукты и услуги;
* на процессы и технологии;
* на формы коллективной деятельности, или управленческие стандарты.

**Виды стандартов**

Существующие на сегодняшний день стандарты можно условно разделить на несколько групп.

* *По предмету стандартизации.*К этой группе можно отнести функциональные стандарты (стандарты на языки программирования, интерфейсы, протоколы) и стандарты на организацию жизненного цикла создания и использования информационных систем (ИС) и программного обеспечения (ПО).
* *По утверждающей организации.*Здесь можно выделить официальные международные, официальные национальные или ведомственные национальные стандарты (например, ГОСТ, ANSI, IDEFO/1), стандарты международных консорциумов и комитетов по стандартизации (например, OMG), стандарты де-факто — официально никем не утвержденные, но фактически действующие (например, стандартом де-факто долгое время были язык взаимодействия с реляционными базами данных SQL и язык программирования С), фирменные стандарты (например, Microsoft ODBC).
* *По методическому источнику.*К этой группе относятся различного рода методические материалы ведущих фирм-разработчиков программного обеспечения, фирм-консультантов, научных центров, консорциумов по стандартизации.

Вкратце рассмотрим методику CDM (Custom Development Method) фирмы Oracle по разработке прикладных ИС под заказ и Международный стандарт ISO/IEC 12207:1995-08-01 01 на организацию жизненного цикла продуктов программного обеспечения.

### Методика CDM фирмы Oracle

Одним из уже сложившихся направлений деятельности фирмы Oracle стали раз­работка методологических основ и производство инструментальных средств для автоматизации процессов разработки сложных прикладных систем, ориентированных на интенсивное использование баз данных. Методика CDM является развитием давно разработанной методики CASE-Method фирмы Oracle, применяемой в CASE-средстве Oracle CASE (в новых версиях — Designer/2000).  
Перечислим основные составляющие CASE-технологии и инструментальной среды фирмы Oracle.

* Методология структурного нисходящего проектирования, при которой разработка прикладной системы представляется в виде последовательности четко определенных этапов.
* Поддержка всех этапов жизненного цикла прикладной системы, начиная с самых общих описаний предметной области до получения и сопровождения готового программного продукта.
* Ориентация на реализацию приложений в архитектуре клиент-сервер с использованием всех особенностей современных серверов баз данных, включая декла­ративные ограничения целостности, хранимые процедуры, триггеры баз данных, с поддержкой в клиентской части всех современных стандартов и требований к графическому интерфейсу конечного пользователя.
* Наличие централизованной базы данных — репозитария. Репозитарий предназначен для хранения спецификаций проекта прикладной системы на всех этапах ее разработки. Он представляет собой базу данных специальной структуры, работающую под управлением СУБД Oracle.
* Возможность одновременной работы с репозитарием многих пользователей. Такой многопользовательский режим почти автоматически обеспечивается стандартными средствами СУБД Oracle. Централизованное хранение проекта системы и управление одновременным доступом к нему всех участников разработки поддерживают согласованность действий разработчиков и не допускают ситуаций, в которых каждый проектировщик или программист работает со своей версией проекта и модифицирует ее независимо от других.
* Автоматизация последовательного перехода от одного этапа разработки к следующему. Для этого предусмотрены специальные утилиты, с помощью которых можно по спецификациям концептуального уровня (модели предметной области) автоматически получать первоначальный вариант спецификации уровня проектирования (описание структуры базы данных и состава программных модулей), чтобы на его основе после всех необходимых уточнений и дополнений автоматически генерировать готовые к выполнению программы.
* Автоматизация различных стандартных действий по проектированию и разработке приложения. Предусматривается генерация многочисленных отчетов по содержимому репозитория, обеспечивающих полное документирование текущей версии системы на всех этапах ее разработки; с помощью специальных процедур предоставляется возможность проверки спецификаций на полноту и не­противоречивость.

#### Общая структура

Жизненный цикл формируется из определенных этапов (фаз) проекта и процессов, каждый из которых выполняется в течение нескольких этапов.  
Методика CDM определяет следующие фазы ЖЦ ИС:  
стратегию;

* анализ (формулирование детальных требований к прикладной системе);
* проектирование (преобразование требований в детальные спецификации системы);
* реализацию (написание и тестирование приложений);
* внедрение (установка новой прикладной системы, подготовка к началу эксплуатации);
* эксплуатацию (поддержка и сопровождение приложения, планирование будущих функциональных расширений).

Первый этап связан с моделированием и анализом процессов, описывающих деятельность организации, технологические особенности работы. Целью является построение моделей существующих процессов, выявление их недостатков и возможных источников совершенствования. Этот этап не является обязательным в случае, когда существующие технология и организационные структуры четко определены, хорошо понятны и не требуют дополнительного изучения и реорганизации.  
На втором этапе разрабатываются детальные концептуальные модели предметной области, описывающие информационные потребности организации, особенности функционирования и т.п. Результатом являются модели двух типов:

* информационные, отражающие структуру и общие закономерности предметной области;
* функциональные, описывающие особенности решаемых задач.

На третьей стадии (этапе проектирования) на основании концептуальных моделей вырабатываются технические спецификации будущей прикладной системы – определяются структура и состав базы данных, специфицируется набор программных модулей. Первоначальный вариант проектных спецификаций может быть получен автоматически с помощью специальных утилит на основании данных концептуальных моделей.  
На этапе реализации создаются программы, отвечающие всем требованиям проектных спецификаций.  
Методика СDМ выделяет следующие процессы, протекающие на протяжении ЖЦ ИС:

* определение производственных требований;
* исследование существующих систем;
* определение технической архитектуры;
* проектирование и построение базы данных;
* проектирование и реализацию модулей;
* конвертирование данных;
* документирование;
* тестирование;
* обучение;
* переход к новой системе;
* поддержку и сопровождение.

#### Особенности методики СDМ

Отметим основные особенности методики CDM, определяющие область ее применения и присущие ей ограничения.

* Степень адаптивности CDM ограничивается тремя моделями жизненного цикла:
* *классическая*модель предусматривает все этапы;
* *быстрая разработка*ориентирована на использование инструментов моделирования и программирования Oracle;
* *облегченный подход*рекомендуется в случае малых проектов и возможности быстро прототипировать приложения.
* Методика не предусматривает включение дополнительных задач, которые не оговорены в CDM, и их привязку к остальным. Также исключено удаление задачи (и порождаемых ею документов), не предусмотренное ни одной из трех моделей жизненного цикла, и изменение предложенной последовательности выполнения задач.
* Все модели жизненного цикла являются по сути каскадными. Даже «облегченный подход», несмотря на итерационность действий по прототипированию, сохраняет общий последовательный и детерминированный порядок выполнения задач.
* Методика не является обязательной, но может считаться фирменным стандартом. При формальном применении степень обязательности полностью соответствует ограничениям возможностей адаптации.
* Прикладная система рассматривается в основном как программно-техническая система, например, возможность выполнения организационно-структурных преобразований, практически всегда происходящих при переходе к новой информационной системе, в этой методике отсутствует.
* CDM теснейшим образом опирается на инструментарий Oracle, несмотря на утверждения о простоте адаптации CDM к проектам, в которых используется другой комплект инструментальных средств.
* Методика CDM представляет собой вполне конкретный материал, детализированный до уровня заготовок проектных документов, рассчитанных на прямое использование в проектах информационных систем с опорой на инструментальные средства и СУБД фирмы Oracle.

### Международный стандарт ISO/IEC 12207: 1995-08-01

Первая редакция ISO Г2207 была подготовлена в 1995 г. подкомитетом SC7 (Проектирование программного обеспечения) объединенного технического комитета JTC1 (Информационные технологии) ISO/IEC.  
По определению, ISO 12207 — базовый стандарт процессов жизненного цикла ПО, ориентированный на различные виды ПО и типы проектов автоматизированных систем, в которых ПО является одной из составных частей. Стандарт определяет стратегию и общий порядок создания и эксплуатации ПО, он охватывает жизненный цикл от концептуализации идей до завершения проекта.  
Целесообразность совместного использования стандартов на ИС и на ПО обусловливается одним из положений ISO 12207, согласно которому процессы, протекающие во время жизненного цикла ПО, должны быть совместимы с процессами, протекающими во время жизненного цикла автоматизированной системы.  
Согласно ISO 12207, система — это объединение одного или нескольких процессов, аппаратных средств, программного обеспечения, оборудования и людей для удовлетворения определенным потребностям или целям.

#### Общая структура

В стандарте ISO 12207 не предусмотрено каких-либо этапов (фаз или стадий) ЖЦ ИС. Данный стандарт определяет лишь ряд процессов, причем по сравнению с CDM стандарт ISO 12207 состоит из гораздо более крупных обобщенных процессов (приобретение, поставка, разработка и т.п.). Несколько утрируя, можно сказать, что один процесс ISO 12207 сопоставим со всеми процессами CDM вместе взятыми.  
Согласно ISO 12207, каждый процесс подразделяется на ряд действий, а каждое действие — на ряд задач.  
Очень важной особенностью ISO 12207 по сравнению с CDM является то, что каждый процесс, действие или задача инициируется и выполняется другим процессом по мере необходимости, причем нет заранее определенных последовательностей (естественно, при сохранении логики связей по исходным сведениям задач и т.п.).

#### Основные и вспомогательные процессы ЖЦ

В стандарте ISO 12207 описаны пять основных процессов ЖЦ программного обеспечения.

* *Процесс приобретения*определяет действия предприятия-покупателя, которое приобретает информационную систему, программный продукт или службу.
* *Процесс поставки*определяет действия предприятия-поставщика, которое снабжает покупателя системой, программным продуктом или службой.
* *Процесс разработки*определяет действия предприятия-разработчика, которое разрабатывает принцип построения программного изделия и программный продукт.
* *Процесс функционирования*определяет действия предприятия-оператора, которое обеспечивает обслуживание системы в целом (а не только программного обеспечения) в процессе ее функционирования в интересах пользователей. В отличие от действий, перечисленных разработчиком в инструкциях по эксплуатации (эта деятельность разработчика предусмотрена во всех трех рассматриваемых стандартах), определяются действия оператора по консультированию пользователей, получению обратной связи и др., которые он планирует сам и берет на себя соответствующие обязанности.
* *Процесс сопровождения*определяет действия персонала, обеспечивающего сопровождение программного продукта, то есть управление модификациями программного продукта, поддержку его текущего состояния и функциональной пригодности; сюда же относятся установка программного изделия на вычислительной системе и его удаление.

Помимо основных, стандарт ISO 12207 оговаривает 8 вспомогательных процессов, которые являются неотъемлемой частью всего ЖЦ программного изделия и обеспечивают должное качество проекта ПО. К вспомогательным процессам относятся:

* решения проблем;
* документирование;
* управление конфигурацией;
* обеспечение качества;
* верификация;
* аттестация;
* совместная оценка;
* аудит.

В стандарте ISO 12207 также определяются четыре организационных процесса:  
управление;  
создание инфраструктуры;  
усовершенствование;  
обучение.  
[spoiler=Примечание**]**  
Под процессом усовершенствования в стандарте ISO 12207 понимается не усовершенствование информационной системы или программного обеспечения, а улучшение самих процессов приобретения, разработки, обеспечения качества и т.д., реально осуществляемых в организации.  
И, наконец, в стандарте ISO 12207 определен один особый процесс, называемый процессом адаптации, который определяет основные действия, необходимые для адаптации этого стандарта к условиям конкретного проекта.[/spoiler]

#### Особенности стандарта ISO 12207

Все сказанное выше позволяет сформулировать некоторые особенности стандарта ISO 12207.

* Стандарт ISO 12207 имеет динамический характер, обусловленный способом определения последовательности выполнения процессов и решения задач, при котором один процесс при необходимости вызывает другой или его часть. Такой характер позволяет реализовать любую модель жизненного цикла.
* Стандарт ISO 12207 обеспечивает максимальную степень адаптивности. Множество процессов и задач сконструировано так, что возможна их адаптация в соответствии с конкретными проектами ИС. Адаптация сводится к исключению процессов, видов деятельности и задач, неприменимых в конкретном проекте.
* Стандарт принципиально не содержит описания конкретных методов действий, а тем более заготовок решений или документации. Он лишь описывает архитектуру процессов ЖЦ ПО, но не конкретизирует в деталях, как предоставлять услуги или решать задачи, включенные в процессы. Данный стандарт не предписывает имена, форматы или точное содержание получаемой документации. Решения такого типа принимаются сторонами, использующими стандарт.

Качество обеспечивается с помощью различных процессов, выполняемых с разной степенью независимости контролирующей деятельности, вплоть до обязательных требований к полной независимости проверяющего персонала от какой-либо прямой ответственности за проверяемые объекты.

* Степень обязательности рассматриваемого стандарта следующая: после решения организации о соответствии торговых отношений стандарту ISO 12207 в качестве условия оговаривается ее ответственность за минимальный набор процессов и задач, которые обеспечивают согласованность с этим стандартом.
* Стандарт содержит предельно мало описаний, направленных на проектирование базы данных. Это можно считать оправданным, так как разные системы и разные прикладные комплексы ПО могут не только использовать весьма специфические типы баз данных, но и вообще обходиться без них.

Ценность стандарта ISO 12207 в том, что в нем представлены наборы задач, характеристики качества, критерии оценки и т.п., обеспечивающие всесторонний охват проектных ситуаций. Например, при анализе требований к системе предусматривается, что:

* рассматривается область применения системы для определения требований, предъявляемых к системе;
* спецификация требований системы должна описывать функции и возможности системы, области применения системы, организационные требования и требования пользователя, безопасность, защищенность, человеческие факторы, эргономику, связи, операции и требования сопровождения; проектные ограничения и квалификационные требования.

При анализе требований к ПО предусмотрено 11 характеристик качества, позволяющих обеспечить заданный уровень качества. При этом разработчик должен установить и документировать в виде требований к ПО следующие спецификации и характеристики:

* функциональные и возможные спецификации, включая исполнение, физические характеристики и условия среды эксплуатации, при которых единица ПО должна быть выполнена;
* внешние связи (интерфейсы) с единицей ПО;
* квалификационные требования;
* спецификации надежности, включая спецификации, связанные с методами функционирования и сопровождения, воздействия окружающей среды и вероятностью травмы персонала;
* спецификации защищенности, включая спецификации, связанные с компрометацией точности информации;
* человеческие факторы спецификаций по инженерной психологии (эргономике), включая связанные с ручным управлением, взаимодействием человека и оборудования, ограничениями на персонал и областями, нуждающимися в концентрированном человеческом внимании, которые являются чувствительными к ошибкам человека и обучению;
* определение данных и требований к базе данных;

§установочные и приемочные требования поставляемого программного продукта в местах функционирования и сопровождения (эксплуатации);

* документацию пользователя;
* требования к интерфейсу пользователя.

[http://mirea.kremlina.ru/templates/Default/dleimages/spoiler-plus.gif](javascript:ShowOrHide('sp4e34697d9319917bedef34bf75cb3a09')) [Примечание](javascript:ShowOrHide('sp4e34697d9319917bedef34bf75cb3a09'))

Хотя стандарт не предписывает конкретной модели ЖЦ или метода разработки, он определяет, что стороны-участники при использовании стандарта ответственны:

* за выбор модели ЖЦ для разрабатываемого проекта;
* за адаптацию процессов и задач стандарта к этой модели;
* за выбор и применение методов разработки ПО;
* за выполнение действий и решение задач, подходящих для проекта ПО.