



МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННЫХ СИСТЕМ

ДИСЦИПЛИНА: ИНТЕГРИРОВАННАЯ
ИНФОРМАЦИОННАЯ ПОДДЕРЖКА ЖИЗНЕННОГО
ЦИКЛА ПРОДУКЦИИ

ТЕМА 2

ВВЕДЕНИЕ

Научоемкие производства - группа производств с высокими абсолютными и относительными затратами на научно-исследовательские и опытно-конструкторские работы



ПОНЯТИЕ ЖИЗНЕННОГО ЦИКЛА

Жизненный цикл продукции - это совокупность процессов, выполняемых в течение реального времени от момента выявления потребности общества в определенной продукции до момента удовлетворения этих потребностей и утилизации продукции (ГОСТ Р ИСО 9004)

Жизненный цикл ИС - ряд событий, происходящих с системой в процессе ее создания и использования

Модель жизненного цикла отражает различные состояния системы, начиная с момента возникновения необходимости в данной ИС и заканчивая моментом ее полного выхода из употребления.

Модель жизненного цикла - структура, содержащая процессы, действия и задачи, которые осуществляются в ходе разработки, функционирования и сопровождения программного продукта в течение всей жизни системы, от определения требований до завершения ее использования.

ЖИЗНЕННЫЙ ЦИКЛ ПРОДУКЦИИ В МЕНЕДЖМЕНТЕ КАЧЕСТВА



ПОДДЕРЖКА ЖИЗНЕННОГО ЦИКЛА

Жизненный цикл продукции



Управление проектами

Управление качеством

Управление развитием персонала

Экономическое обеспечение деятельности

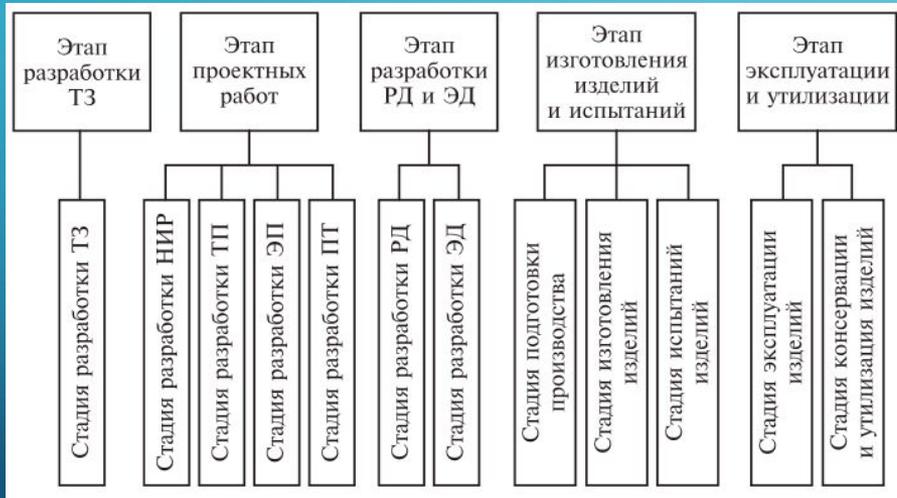
Математическое моделирование и принятие решений

Управление бизнес-процессами

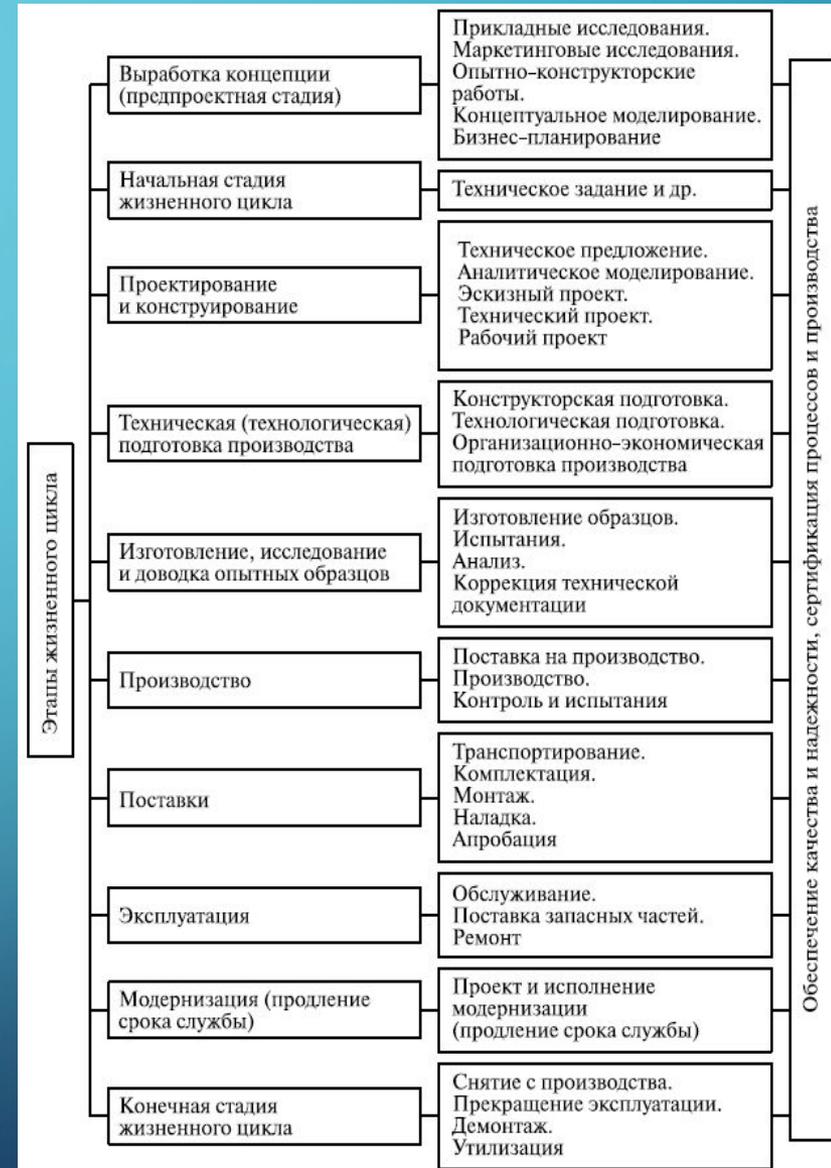
Информационные технологии

Поддержка жизненного цикла

ОСНОВНЫЕ СТАДИИ ЖИЗНЕННОГО ЦИКЛА ПРОДУКЦИИ



ТЗ – техническое задание, РД – рабочая документация, ЭД – электронная документация, ТП – техническое предложение, ЭП – эскизный проект, ПТ – проект технический



МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Каскадная модель

- **Каскадная модель** предусматривает последовательное выполнение всех этапов проекта в строго фиксированном порядке. Переход на следующий этап означает полное завершение работ на предыдущем этапе.



- Данная модель применяется при разработке информационных систем, для которых в самом начале разработки можно достаточно точно и полно сформулировать все требования.

МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Каскадная модель

Достоинства каскадной модели:

- на каждой стадии формируется законченный набор документации, программного и аппаратного обеспечения, отвечающий критериям полноты и согласованности;
- выполняемые в четкой последовательности стадии позволяют уверенно планировать сроки выполнения работ и соответствующие ресурсы (денежные, материальные и людские).

Недостатки модели:

- реальный процесс разработки информационной системы редко полностью укладывается в такую жесткую схему. Особенно это относится к разработке нетиповых и новаторских систем;
- жизненный цикл основан на точной формулировке исходных требований к информационной системе. Реально в начале проекта требования заказчика определены лишь частично;
- основной недостаток – результаты разработки доступны заказчику только в конце проекта. В случае неточного изложения требований или их изменения в течение длительного периода создания ИС заказчик получает систему, не удовлетворяющую его потребностям.

МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Итерационная модель

- Итерационная модель - поэтапная модель с промежуточным контролем и циклами обратной связи.



- Разработка ИС ведется итерациями с циклами обратной связи между этапами. Межэтапные корректировки позволяют учитывать реально существующее взаимовлияние результатов разработки на различных этапах; время жизни каждого из этапов растягивается на весь период разработки

МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Итерационная модель

Достоинства итерационной модели:

- поэтапные корректировки, которые обеспечивают меньшую трудоемкость по сравнению с каскадной.

Недостатки модели:

- время жизни каждого из этапов рассчитывается на весь период разработки

МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Инкрементная модель

- Инкрементная модель - поэтапная модель с промежуточным контролем и циклами обратной связи.



- В начале работы над проектом определяются все основные требования к системе, после чего выполняется ее разработка в виде последовательности версий. При этом каждая версия является законченным и работоспособным продуктом. Первая версия реализует часть запланированных возможностей, следующая версия реализует дополнительные возможности и т. д., пока не будет получена полная система.

МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Инкрементная модель

Разработка версиями ведется в силу разного рода причин:

- отсутствия у заказчика возможности **сразу профинансировать** весь дорогостоящий проект;
- **отсутствия** у разработчика необходимых **ресурсов** для реализации сложного проекта в сжатые сроки;
- требований **поэтапного внедрения** и **освоения** продукта конечными пользователями.

Достоинства и недостатки этой стратегии такие же, как и у каскадной.

Но в отличие от классической стратегии заказчик может раньше увидеть результаты. Уже по результатам разработки и внедрения первой версии он может незначительно изменить требования к разработке, отказаться от нее или предложить разработку более совершенного продукта с заключением нового договора.

МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Спиральная модель

- **Спиральная модель** - На каждом витке спирали выполняется создание очередной версии продукта, уточняются требования проекта, определяется его качество и планируются работы следующего витка.

- Особое внимание уделяется начальным этапам разработки - анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством создания прототипов (макетирования).



МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Спиральная модель

Достоинства модели:

- позволяет быстрее показать пользователям системы работоспособный продукт, тем самым, активизируя процесс уточнения и дополнения требований;
- допускает изменение требований при разработке информационной системы, что характерно для большинства разработок, в том числе и типовых;
- обеспечивает большую гибкость в управлении проектом;
- позволяет получить более надежную и устойчивую систему. По мере развития системы ошибки и слабые места обнаруживаются и исправляются на каждой итерации;
- позволяет совершенствовать процесс разработки – анализ, проводимый в каждой итерации, позволяет проводить оценку того, что должно быть изменено в организации разработки, и улучшить ее на следующей итерации;
- уменьшаются риски заказчика. Заказчик может с минимальными для себя финансовыми потерями завершить развитие неперспективного проекта.

МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Сравнение моделей жизненного цикла

Характеристика проекта	Модель (стратегия)		
	<u>Каскадная</u>	<u>Инкрементная</u>	<u>Спиральная</u>
Новизна разработки и обеспеченность ресурсами	Типовой. Хорошо проработаны технология и методы решения задачи Ресурсов заказчика и разработчика хватает для реализации проекта в сжатые сроки	Ресурсов заказчика или разработчика не хватает для реализации проекта в сжатые сроки	Нетиповой (новаторский). Нетрадиционный для разработчика
Масштаб проекта	Малые и средние проекты	Средние и крупные проекты	Любые проекты
Сроки выполнения проекта	До года	До нескольких лет. Разработка одной версии может занимать срок от нескольких недель до года	
Заключение отдельных договоров на отдельные версии	Заключается один договор. Версия и есть итоговый результат проекта	На отдельную версию или несколько последовательных версий обычно заключается отдельный договор	
Определение основных требований в начале проекта	Да	Да	Нет
Изменение требований по мере развития проекта	Нет	Незначительное	Да
Разработка итерациями (версиями)	Нет	Да	Да
Распространение промежуточного ПО	Нет	Может быть	Да

МЕТОДОЛОГИИ, ПОДДЕРЖИВАЮЩИЕ СПИРАЛЬНУЮ МОДЕЛЬ

Методология RAD - методология быстрой разработки приложений (Rapid Application Development, RAD)

Под **RAD-разработкой** обычно понимается процесс разработки, содержащий 3 элемента:

- небольшую команду программистов (до 10 человек);
- короткий, но тщательно проработанный производственный график (от 2 до 6 месяцев);
- повторяющийся цикл, при котором разработчики по мере того, как приложение начинает обретать форму, реализуют в продукте требования, полученные через взаимодействие с заказчиком.

МЕТОДОЛОГИИ, ПОДДЕРЖИВАЮЩИЕ СПИРАЛЬНУЮ МОДЕЛЬ

Методология RAD

Помимо особенностей, характерных для спиральной модели жизненного цикла, методология RAD подразумевает использование на каждой итерации:

- CASE-средств для формирования и анализа требований, проектирования системы, автоматической генерации кода программ и структуры БД, а также автоматического тестирования программного обеспечения;
- инструментальных средств, обеспечивающих визуальную разработку (программирование) системы. Среда разработки приложений позволяет без написания кода программы создавать («рисовать») сложные графические интерфейсы пользователя, состав и структуру БД, запросы к БД, а также связывать данные с элементами интерфейса (переключателями, полями ввода, таблицами и т. д.);
- инструментальных средств, поддерживающих объектно-ориентированный подход. Эти средства позволяют создать описание предметной области в виде совокупности объектов – сущностей реального мира, характеризуемых свойствами (атрибутами) и поведением (методами);
- инструментальных средств, обеспечивающих событийное программирование. Каждый объект, входящий в состав приложения, может генерировать события и реагировать на события, генерируемые другими объектами;
- шаблонов и библиотек готовых решений как собственной разработки, так и сторонних производителей.

МЕТОДОЛОГИИ, ПОДДЕРЖИВАЮЩИЕ СПИРАЛЬНУЮ МОДЕЛЬ

Методология RAD

Условия применения методологии RAD:

- применима для относительно небольших проектов, разрабатываемых под конкретного заказчика;
- неприменима для построения сложных расчетных программ, операционных систем или программ управления космическими кораблями, т.е. программ, требующих написания большого объема (сотни тысяч строк) уникального кода;
- неприменима для разработки приложений, в которых отсутствует ярко выраженная интерфейсная часть, наглядно определяющая логику работы системы (например, приложения реального времени);
- неприменима для разработки приложений, от которых зависит безопасность людей (например, управление самолетом или атомной электростанцией), так как итеративный подход предполагает, что первые несколько версий, скорее всего не будут полностью работоспособны, что в данном случае исключается.

МЕТОДОЛОГИИ, ПОДДЕРЖИВАЮЩИЕ СПИРАЛЬНУЮ МОДЕЛЬ

Методология XP - eXtreme Programming, XP – автор Кент Бек, 1999

Данный подход ориентирован на разработку информационных систем группами малого и среднего размера в условиях неопределенных или быстро изменяющихся требований.

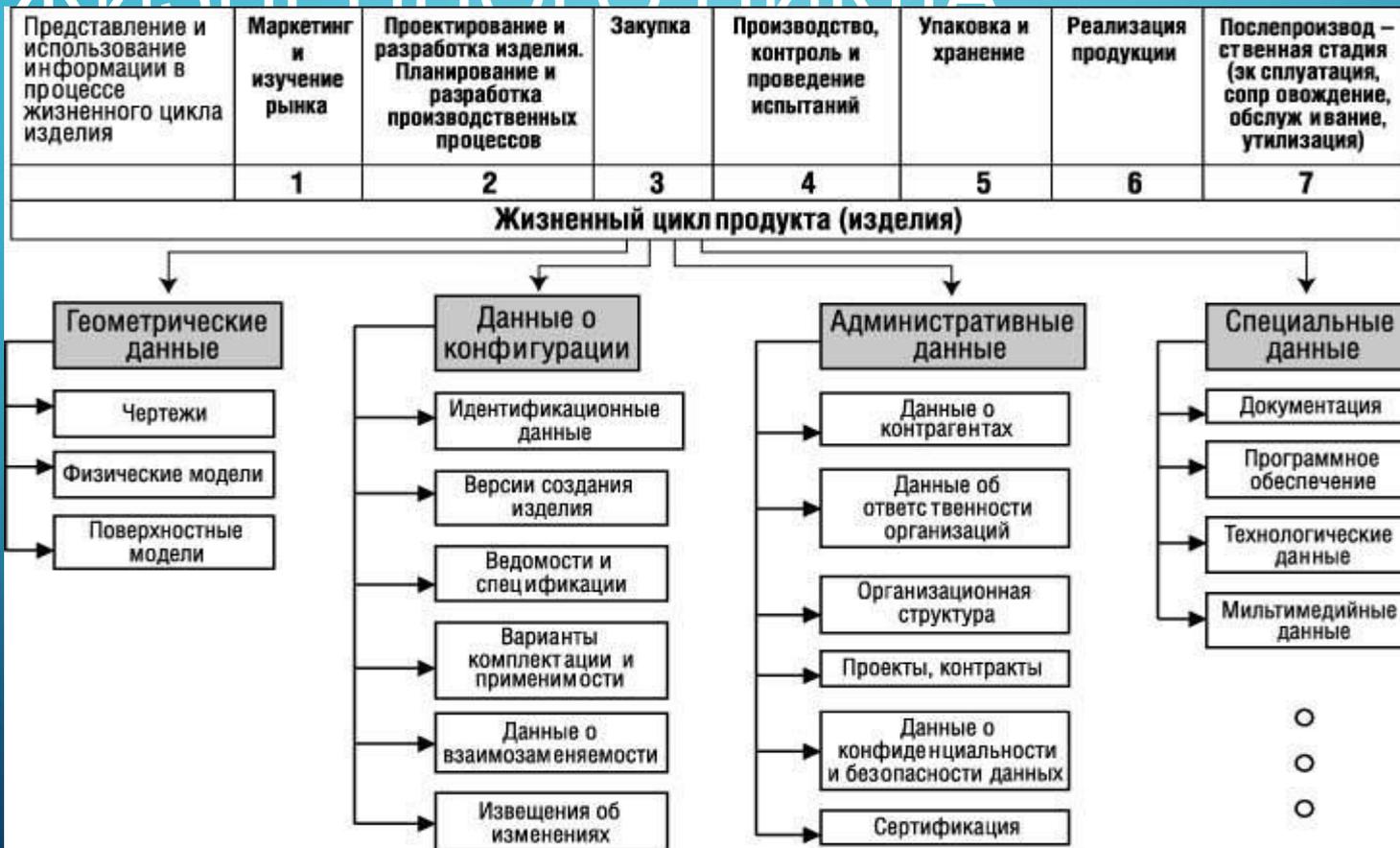
Отличительными **особенностями** XP-разработки являются:

- частая смена версий и модификаций (длительность итераций вплоть до часов и минут, а обычно – 2 недели; в RAD – минимум 2 месяца);
- непрерывная связь с заказчиком – в группе все время находится квалифицированный представитель заказчика;
- простое проектирование – при разработке всегда выбирается наиболее простое решение;
- простой дизайн – система должна быть спроектирована настолько просто, насколько это возможно на каждый момент времени;
- коллективное владение кодом – любой, кто видит возможность улучшить какую-то часть кода, может сделать это в любой момент времени;
- программирование в парах – на пару программистов приходится один компьютер. Пока один из них непосредственно программирует, другой обдумывает вопросы реализации требований (функций, БД и т.п.);
- непрерывное и пересекающееся проектирование, разработка, интеграция и тестирование системы.

СТРУКТУРА ЖИЗНЕННОГО ЦИКЛА ПРОДУКТА (ИЗДЕЛИЯ)

Жизненный цикл продукта (изделия)							
	1	2	3	4	5	6	7
Представление и использование информации в процессе жизненного цикла изделия	Маркетинг и изучение рынка	Проектирование и разработка изделия. Планирование и разработка производственных процессов	Закупка	Производство, контроль и проведение испытаний	Упаковка и хранение	Реализация продукции	Послепроизводственная стадия (эксплуатация, сопровождение, обслуживание, утилизация)
<i>Проектные орган и зации</i>							
<i>Производители</i>							
<i>Дистрибьюторы</i>							
<i>Потребители</i>							
<i>Службы материально-технического снабжения</i>							
<i>Сервисные орган и зации</i>							
<i>Государственные организации</i>							

СТРУКТУРА ДАННЫХ ОБ ИЗДЕЛИИ В ПРОЦЕССЕ ЖИЗНЕННОГО ЦИКЛА

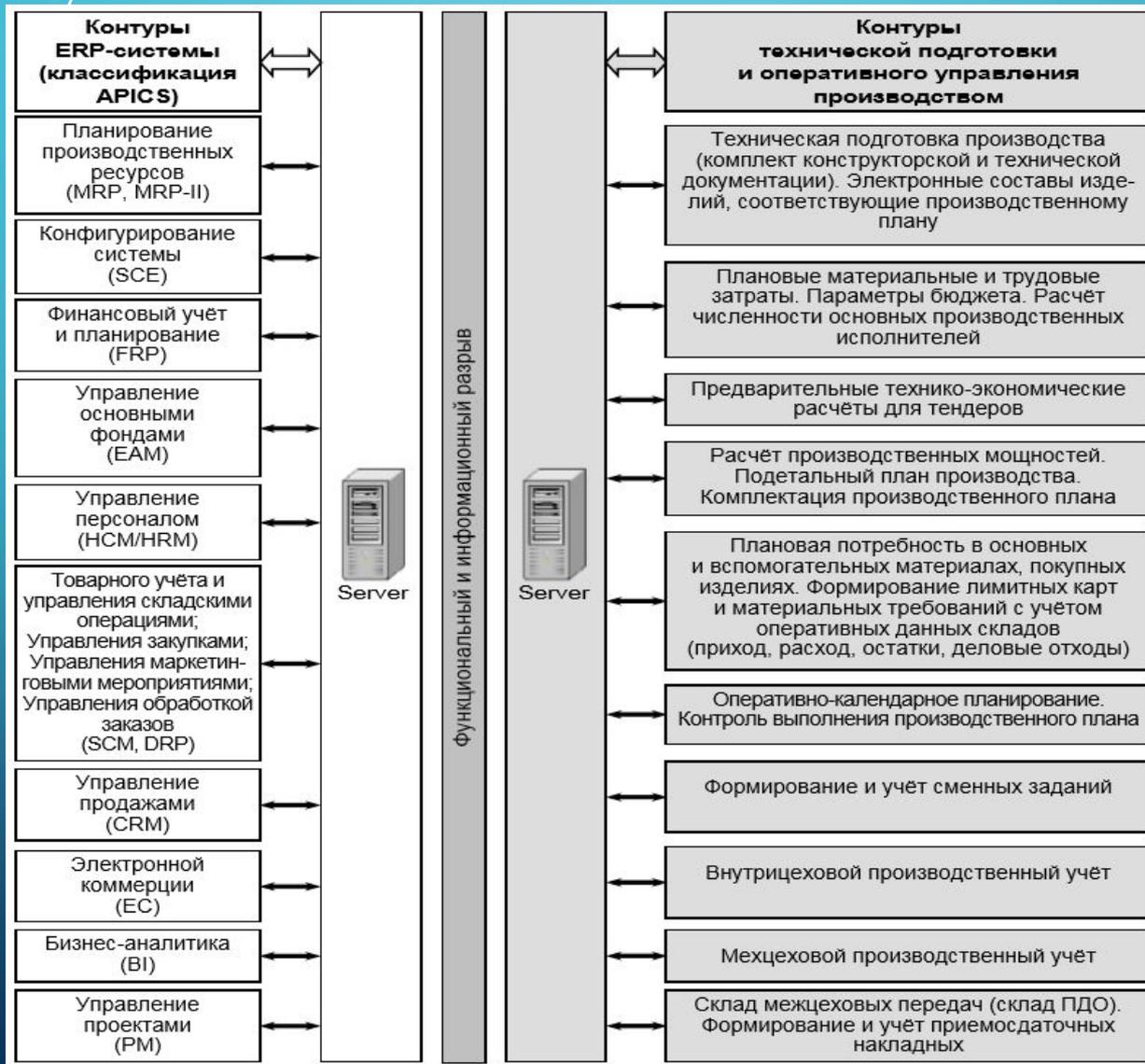


ПРИМЕНЕНИЕ CALS-ТЕХНОЛОГИЙ

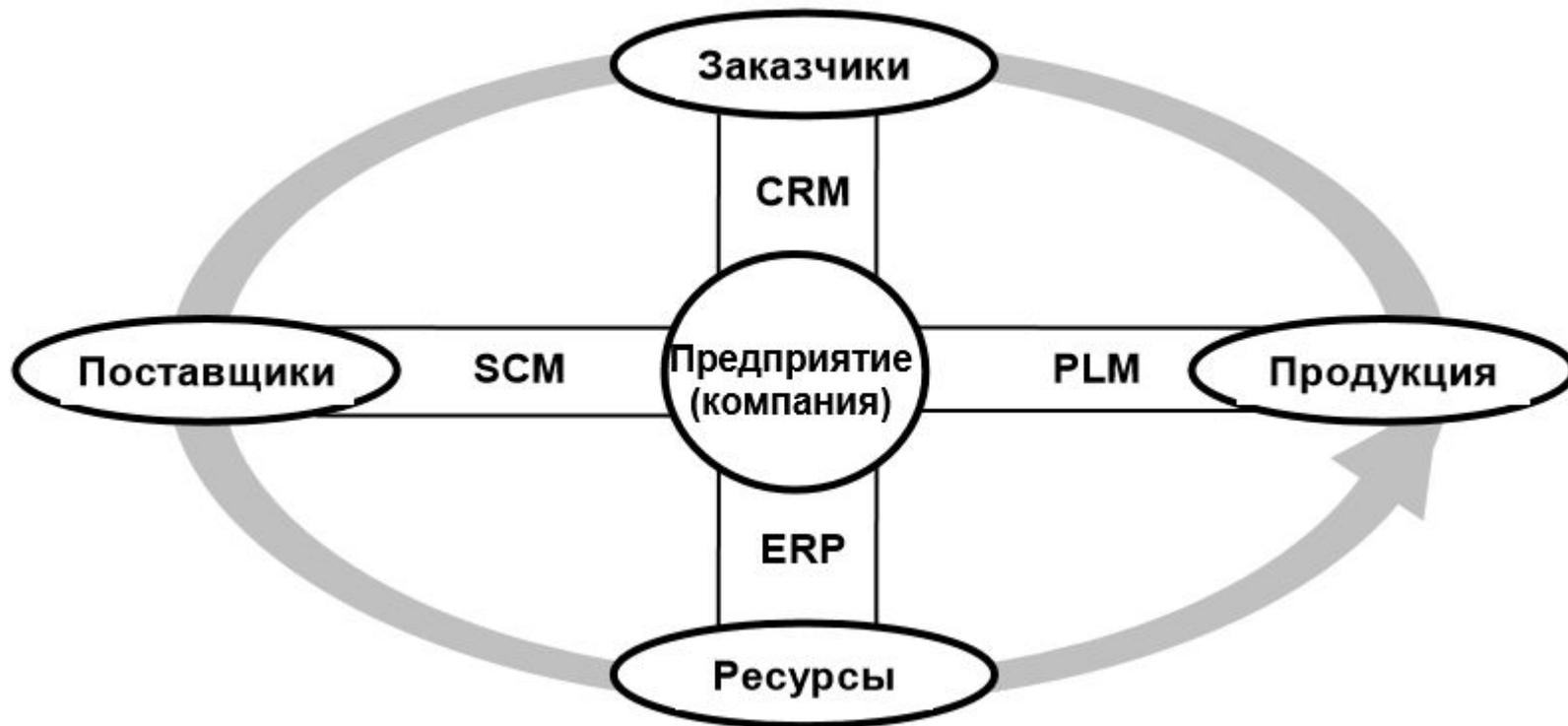
CALS-технологии ([англ.](#) *Continuous Acquisition and Lifecycle Support* — непрерывная информационная поддержка поставок и жизненного цикла изделий), или *ИПИ* (информационная поддержка процессов жизненного цикла изделий) — подход к проектированию и производству высокотехнологичной и наукоёмкой продукции, заключающийся в использовании компьютерной техники и информационных технологий на всех стадиях [жизненного цикла изделия](#).



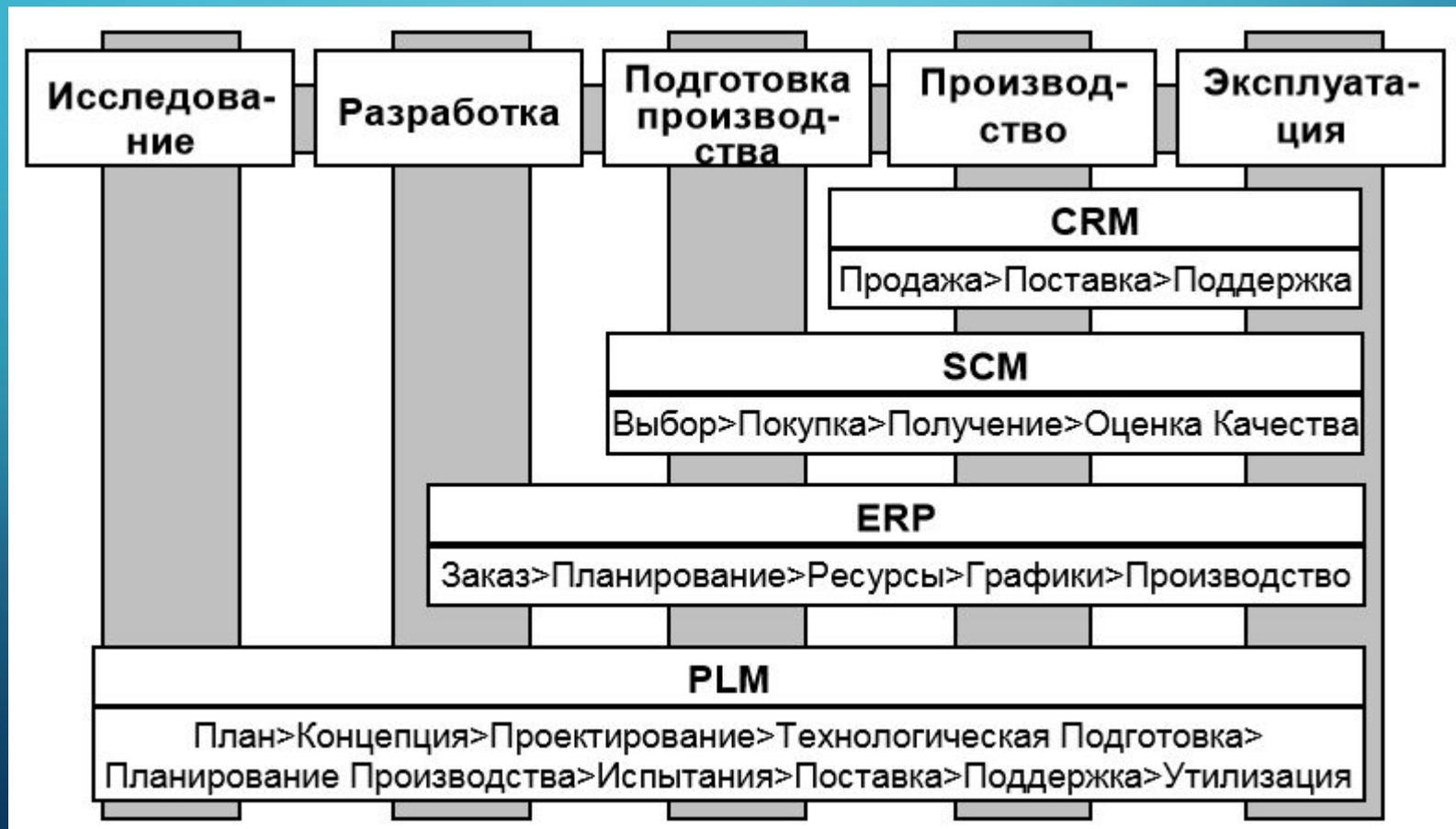
ИЛЛЮСТРАЦИЯ ФУНКЦИОНАЛЬНОГО И ИНФОРМАЦИОННОГО РАЗРЫВА МЕЖДУ УРОВНЯМИ АСУП/ERP СИСТЕМ И УРОВНЯМИ САПР /CAD, АСУПП /MES, АСУТП/САР



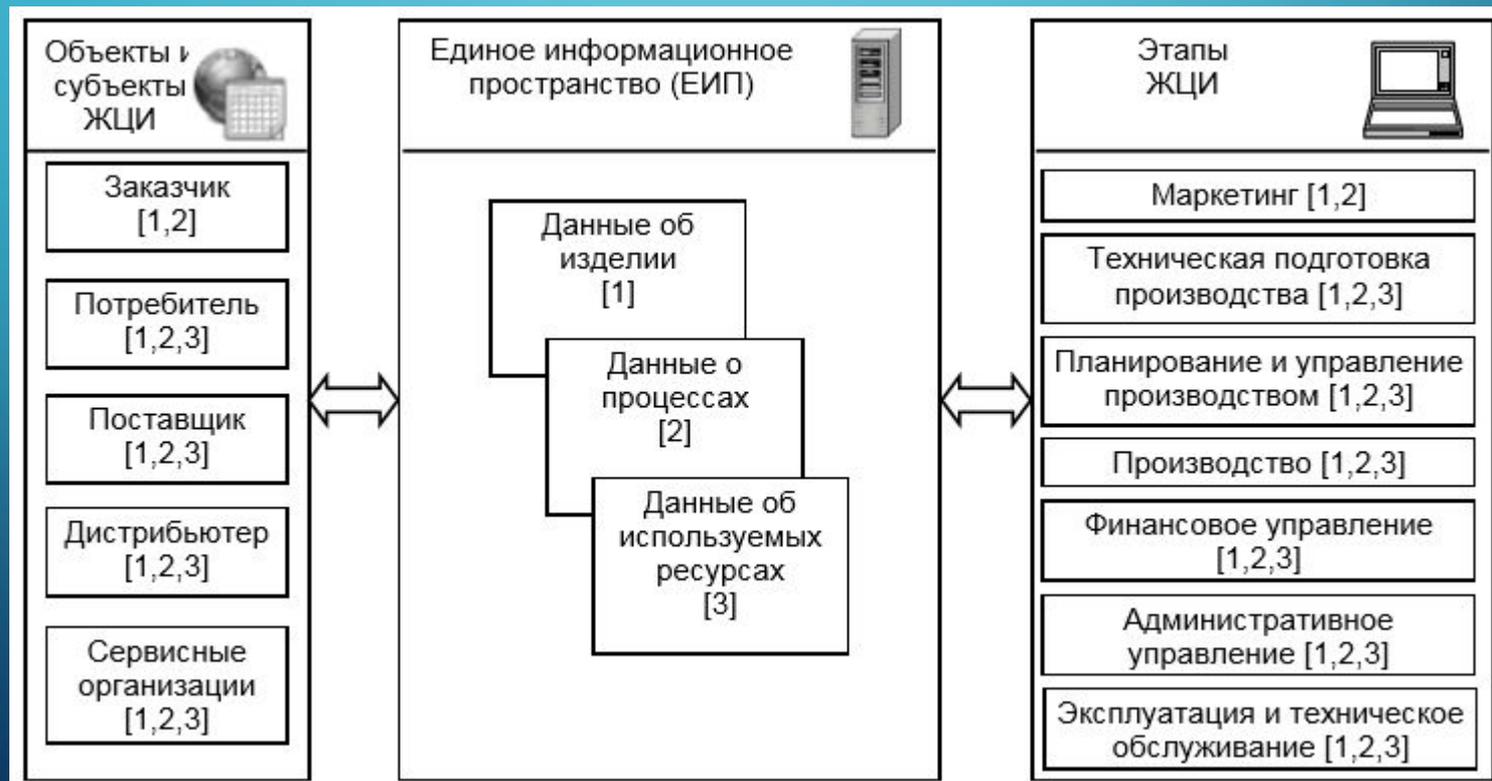
РАСПРЕДЕЛЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ ПО ЦЕЛЕВОЙ НАПРАВЛЕННОСТИ



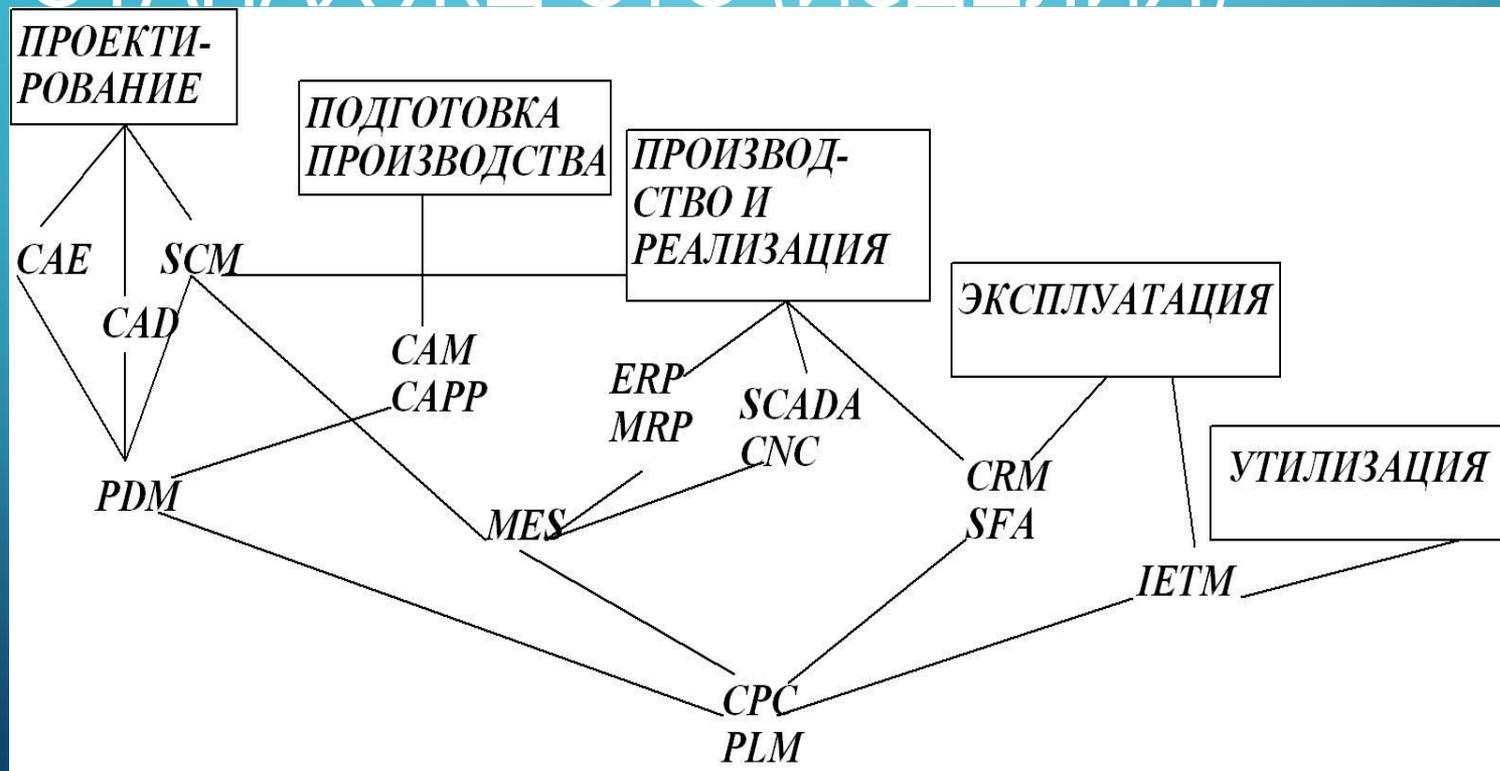
РАСПРЕДЕЛЕНИЕ СИСТЕМ УПРАВЛЕНИЯ ПО ЭТАПАМ ЖЦИ



СОВМЕШНОЕ ИСПОЛЬЗОВАНИЕ ДАННЫХ (ИНФОРМАЦИИ, ЗНАНИЙ) ОБ ИЗДЕЛИЯХ, ПРОЦЕССАХ И РЕСУРСАХ НА РАЗЛИЧНЫХ ЭТАПАХ ЖЦ СТО (ИЗДЕЛИЯ)



СОВМЕСТНОЕ ИСПОЛЬЗОВАНИЕ ДАННЫХ (ИНФОРМАЦИИ, ЗНАНИЙ) ОБ ИЗДЕЛИЯХ, ПРОЦЕССАХ И РЕСУРСАХ НА РАЗЛИЧНЫХ ЭТАПАХ ЖЦ СТО (ИЗДЕЛИЯ)



СОСТАВ CALS

- CAE - Computer Aided Engineering (автоматизированные расчеты и анализ);
- CAD - Computer Aided Design (автоматизированное проектирование);
- CAM - Computer Aided Manufacturing (автоматизированная технологическая подготовка производства);
- CAPP — система проектирования технологических процессов (ТП), которая позволяет с различной степенью автоматизации проектировать единичные, групповые и типовые технологические процессы по многим направлениям: механообработка, гальваника, сварка, сборка, термообработка и т.д.;
- PDM - Product Data Management (управление проектными данными);
- ERP - Enterprise Resource Planning (планирование и управление предприятием);
- MRP-2 - Manufacturing (Material) Requirement Planning (планирование производства);
- MES - Manufacturing Execution System (производственная исполнительная система);
- SCM - Supply Chain Management (управление цепочками поставок);
- CRM - Customer Relationship Management (управление взаимоотношениями с заказчиками);
- SCADA - Supervisory Control And Data Acquisition (диспетчерское управление производственными процессами);
- CNC - Computer Numerical Control (компьютерное числовое управление);
- SFA — это Sales Force Automation (Автоматизация деятельности по продажам);
- IETM - Interactive Electronic Technical Manuals (интерактивные электронные технические руководства)
- CPC - Collaborative Product Commerce (совместный электронный бизнес).
- PLM - Product Lifecycle Management (Управление данными жизненного цикла изделий).