

Проектирование корпоративных информационных систем

При проектировании АИС должны быть обеспечены:

- эффективная поддержка принятия корректных решений на основе действующих нормативно-правовых документов, стандартов, методик и технологий проектирования;
- высокий уровень проектных решений, реализующий необходимый срок эксплуатации, а также модернизацию и реконструкцию системы;
- дружественные и технологичные проектные интерфейсы с разработчиками, учитывающие возможности обучения и повышения квалификации;
- эффективные средства управления проектированием и обеспечения информационной безопасности проектных работ.

В узкоспециальном плане системное проектирование рассматривается как набор методов и организационная дисциплина, которые предназначены для проектирования информационных систем определенных видов. Основные классы таких ИС:

- общеуправленческие ИС (*MIS – management information system* и *EIS – executive information system*);
- специализированные ИС отраслям производства (банковские и управленческие системы, управление дискретным промышленным производством, системы профилактической и режимной деятельности органов МВД и др.);
- специализированные ИС по видам деятельности (управление работой склада, система маркетинговых исследований и т.д.);
- адаптивные универсальные ИС по применяемым методам обработки информации (электронный архив, система статистических расчетов, корпоративная система управления процессом выполнения офисных работ и др.)

Вывод

Т. о, в ИС включаются и те типы систем, которые еще 10-15 лет назад рассматривались как отдельные от ИС: диалоговые системы решения задач и системы управления технологическими процессами (как ИС не рассматриваются системы прямого управления механизмами и агрегатами, но они могут быть компонентами ИС).

ЭТАПЫ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

2 направления деятельности, имеющие самое непосредственное отношение к проектированию ИС:

1. собственно проектирование АИС конкретных предприятий (отраслей) на базе готовых программных и аппаратных компонентов с помощью специальных инструментальных средств разработки;
2. проектирование компонентов АИС и инструментальных средств, ориентированных на многократное применение при разработке многих ИС.

Сущность первого направления – системная интеграция. Разработчик АИС должен:

- быть специалистом в области системотехники;
- хорошо знать международные стандарты, состояние и тенденции развития ИТ и программных продуктов;
- владеть инструментальными средствами разработки приложений (CASE-средствами);
- быть готовым к анализу и восприятию автоматизируемых прикладных процессов в сотрудничестве со специалистами соответствующей предметной области

Второе направление

в большей мере относится к области разработки математического и программного обеспечения для реализации функций АИС – моделей, методов, алгоритмов, программ на базе знания системотехники, методов анализа и синтеза проектных решений, технологий программирования, операционных систем и т.п.

Концептуальное проектирование

Выполняется в процессе предпроектных исследований, формулировки технического предложения, разработки эскизного проекта.

Результаты анализа – техническое предложение и бизнес-план создания АИС (предоставляются заказчику для окончательного согласования).

модели преобразования, хранения и передачи информации:

- функциональные,
- информационные,
- поведенческие,
- структурные

- **Функциональная модель системы** описывает совокупность выполняемых системой функций.
- **Информационные модели** отражают структуры данных: их состав и взаимосвязи.
- **Поведенческие модели** описывают информационные процессы (динамику функционирования), в них фигурируют такие категории как состояние системы, событие, переход из одного состояния в другое, условия перехода, последовательность событий.
- **Структурные модели** характеризуют морфологию системы (ее построение) – состав подсистем, их взаимосвязи.

Содержание следующих этапов проектирования:

- определение перечней приобретаемого оборудования и готовых программных продуктов;
- построение системной среды;
- детальное инфологическое проектирование баз данных и их первоначального наполнения;
- разработка собственного прикладного ПО (в свою очередь, делится на ряд этапов нисходящего проектирования).

Разработка проекта корпоративной телекоммуникационной сети

- Если территориально АИС располагается в одном здании или нескольких близкорасположенных, то корпоративная сеть может быть выполнена в виде совокупности нескольких локальных подсетей. Кроме выбора типов подсетей, протоколов и коммутационного оборудования приходится решать задачи распределения узлов по подсетям, выделения серверов, выбора сетевого ПО, определения способа управления данными в выбранной схеме распределенных вычислений и т.д.
- Если АИС располагается в удаленных друг от друга пунктах (например, в разных городах), то решается вопрос об аренде каналов связи для корпоративной сети, поскольку альтернативный вариант использования выделенного канала оказывается часто неприемлемым из-за высокой цены. Проблемы связаны и с обеспечением информационной безопасности и надежности доставки сообщений.

Инструментальные средства проектирования и разработки ИС

Толкование аббревиатуры CASE, соответствующее двум направлениям использования CASE-систем

1. *Computer Aided Software Engineering* – автоматизированное проектирование ПО, соответствующие CASE-системы часто называют инструментальными средами разработки ПО (*RAD – Rapid Application Development*, например, VB, Delphi, PowerBuilder и т.д.).
2. *Computer Aided System Engineering* – подчеркивает направленность на поддержку концептуального проектирования сложных систем, преимущественно слабоструктурированных. Такие CASE-системы часто называют *BPR (Business Process Engineering)*.

Среди RAD различают

- интегрированные комплексы для автоматизации всех этапов жизненного цикла ПО (такие системы называют Workbench)
- специализированные инструментальные средства для выполнения отдельных функций (Tools).

CASE-средства по своему функциональному назначению принадлежат к одной из следующих групп:

- средства программирования;
- средства управления программным проектом;
- средства верификации (анализа) программ;
- средства документирования.

Спецификации моделей информационных систем

Общие черты *функциональных* спецификаций :

- модель имеет иерархическую структуру, представляемую в виде диаграмм нескольких уровней;
- элементарной частью диаграммы каждого уровня является конструкция «вход-функция-выход»;
- необходимая дополнительная информация содержится в файлах поясняющего текста.

- *DFD – Data Flow Diagrams*
- *ERD-Entity-Relations Diagrams*

Примеры языков 4GL

- Informix-4GL,
- JAM,
- NewEra

СЕМЕЙСТВО СТАНДАРТОВ СТРУКТУРНОГО МОДЕЛИРОВАНИЯ *IDEF (Integrated Definition) -*

взаимосвязанная совокупность методик
концептуального проектирования
разработана по программе *Integrated
Computer-Aided Manufacturing* в США

1. **IDEF0** (*Function Modeling Method*) – методология функционального моделирования. С помощью наглядного графического языка система предстает в виде набора связанных функций (функциональных блоков)
2. **IDEF1** (*Information Modeling Method*) – методология моделирования информационных потоков внутри систем, позволяющая отображать их структуру и взаимосвязи.
3. **IDEF1X** (*IDEF1 Extended, Data Modeling Method*) – методология построения реляционных информационных структур. **IDEF1X** относится к типу методологий «сущность-связь» и, как правило, используется для моделирования реляционных баз данных, имеющих отношение к рассматриваемой системе.
4. **IDEF2** (*Simulation Modeling Method*) – методология динамического моделирования развития систем. в настоящее время известны алгоритмы и их компьютерные реализации, позволяющие превращать набор статических диаграмм **IDEF0** в динамические модели, построенные на базе «раскрашенных сетей Петри» (*CPN- Color Petri Nets*).

5. **IDEF3** (*Process Flow and Object State Description Capture Method*) – методология документирования процессов, происходящих в системе. С помощью **IDEF3** описываются сценарий и последовательность операций для каждого процесса. Функция в диаграмме **IDEF0** может быть представлена в виде отдельного процесса средствами **IDEF3**.
6. **IDEF4** (*Object-Oriented Design Method*) – методология построения объектно-ориентированных систем. Средства **IDEF4** позволяют наглядно отображать структуру объектов и заложенные принципы их взаимодействия, позволяя тем самым анализировать и оптимизировать сложные объектно-ориентированные системы.
7. **IDEF5** (*Ontology Description Capture Method*) – методология онтологического исследования сложных систем. С помощью этой методологии онтология системы может быть описана при помощи определенного словаря терминов и правил, на основе которых могут быть сформированы достоверные утверждения о состоянии рассматриваемой системы в некоторый момент времени. На основе этих утверждений формируются выводы о дальнейшей развитии системы и производится ее оптимизация.

8. **IDEF6** (*Design Rationale Capture Method*) – метод рационального представления процесса проектирования информационных систем, позволяющий обосновать необходимость проектируемых моделей, выявить причинно-следственные связи и отразить это в итоговой документации системы.
9. **IDEF8** (*User Interface Modeling*) – *Human-System Interaction Design* – метод проектирования взаимодействия пользователя с системами различной природы (не обязательно информационно-вычислительными).
10. **IDEF9** (*Business Constraint Discovery Method*) – метод изучения и анализа бизнес-систем в терминах «принуждений» (*constraint*). Принуждения инициируют результат, руководят и ограничивают поведение объектов и агентов (автономных программных модулей) для выполнения целей или намерений системы.
11. **IDEF14** (*Network Design Method*) – метод проектирования вычислительных сетей, позволяющий устанавливать требования, определять сетевые компоненты, анализировать существующие сетевые конфигурации и формулировать желаемые характеристики сети.

Анонсируемые корпорацией ***KBSI***
(*Knowledge Based System Inc.*) методы

IDEF7 (*Information System Audit Method*),
IDEF9 (*Information Artifact Modeling*),
IDEF12 (*Organization Design*)

не получили дальнейшего развития.

КЛАССИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

- представительность соответствующих технологий,
- ориентация на наиболее массовую часть ИС,
- наличие не только теоретических оснований, но и промышленных методик и стандартов (ГОСТов, ANSI, ISO), использование этих методик в течение десятилетий.

проектные стадии:

- **запуск:** организация основания для деятельности и запуск работ: приказ и(или) договор о разработке автоматизированной системы; задание на выполнение работ (*proposal for development, agreement, mobilization*);
- **обследование:** предпроектное обследование, общий анализ ситуации на предприятии, разработка общего обоснования о целесообразности создания ИС (*feasibility study, scope analysis, strategy study and planning, requirement definition*);
- **концепция, ТЗ:** исследование требований предприятия и пользователей, выработка рекомендаций по разработке ИС. разработка ТЗ на проектирование ИС в целом и частных ТЗ по подсистемам (*strategy planning, analysis, requirement specification, function description*);
- **эскизный проект:** разработка архитектуры будущей ИС в рамках эскизного проекта (*detailed analysis, high level design*);

проектные стадии:

- *опытный вариант ИС: разработка упрощенного варианта, пилотного проекта будущей ИС (pilot project, test development);*
- *опытное использование пилот-проекта ИС, разработка исправлений и дополнений к ТЗ (test, corrected requirement specification);*
- *ТП: разработка технического проекта ИС (detailed analysis and design, test development);*
- *РП: разработка рабочей документации проекта (development, test, system implementation);*
- *ввод в действие: внедрение ИС (deployment, put into operation).*

- Одно из названий такой схемы организации работ (в западной литературе) – «*водопадная или каскадная модель*» (*waterfall model*).
Схема обязана была включать в себя итерационные процедуры уточнения требований к системе и рассмотрения вариантов проектных решений. Предметом была проектируемая ИС целиком, в целостном ее представлении.

Положительные факторы применения каскадной схемы:

- на каждой стадии формировался законченный, отвечающий критериям полноты и согласованности набор проектной, а затем и пользовательской документации, охватывающий все предусмотренные стандартами виды обеспечения ИС: организационное, методическое, информационное, программное, аппаратное и др.;
- выполняемые в логической последовательности этапы работ достаточно очевидным образом позволяли планировать сроки завершения всех работ и планируемые затраты.

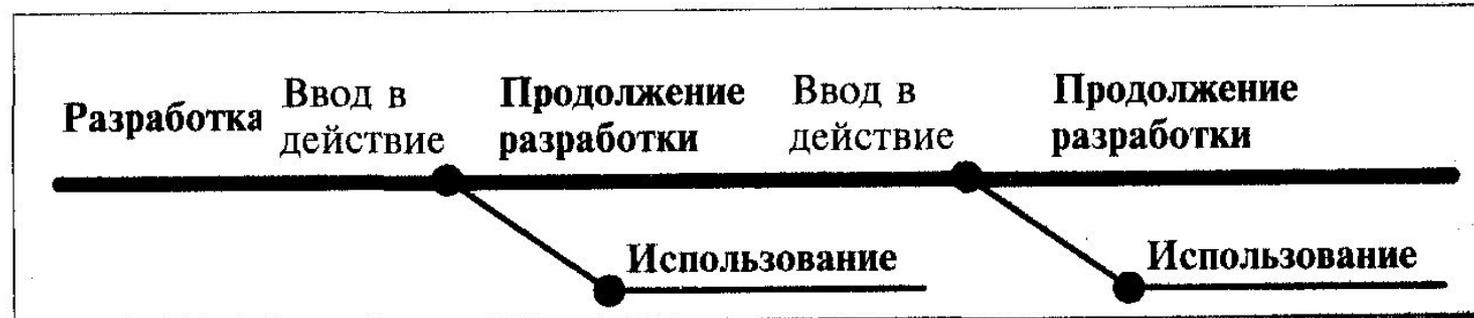
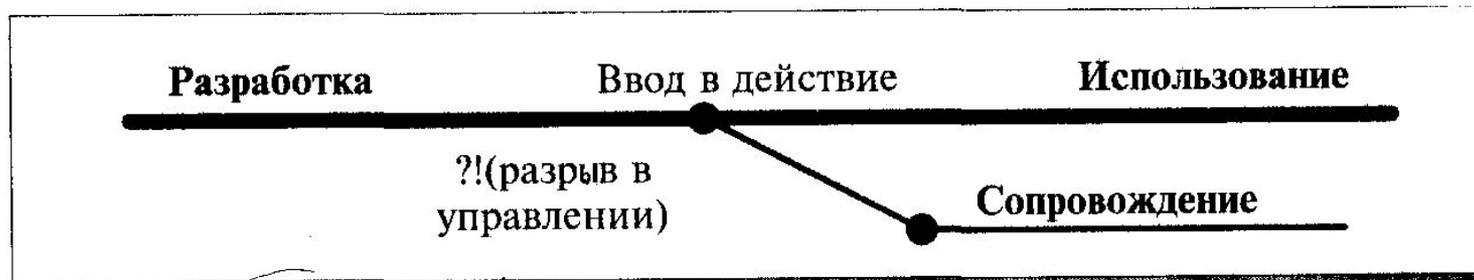
Структура формирования ИС

Стадии проекта	Виды обеспечения ИС				
	организа- ционное	методи- ческое	инфор- мацион- ное	програм- мное	аппарат- ное
Запуск	+ -				
Обследование	+ -	+ -	+ -		
Концепция ТЗ	+ -	+ -	+ -		
Эскизный проект	+ -	+ -	+ -	+ -	
ТП	+ -	++	+	+ -	+ -
РП	++	++	++	++	+
Ввод в действие	++	++	++	++	++
Символами «+», «+-», «++» показаны примерные оценки доли наличия каждого компонента на каждой стадии					

Отрицательные факторы

- *Недостаток 1 (опоздание)* – существенное запаздывание с получением результатов
- *Недостаток 2 (бесполезность)* – проектирование ИС очень часто вело к примитивной автоматизации («что на входе, то и на выходе») без изменения бизнес-процессов (механическое перемалывание существующего бумажного потока)
- *Недостаток 3 и 4 (жесткость и закрытость)*
- *Недостаток 5 (типовые оргструктуры)*

Схема непрерывной разработки



Модель-луковица закрытой ИС



Схема циклической разработки (быстрое прототипирование – rapid prototyping approach, fast-track).

В проектный цикл дополнительно включались стадии:

- разработка макета-прототипа фрагмента будущей ИС (*rapid prototyping*) совместно с будущим пользователем;
- опробование макета-прототипа фрагмента будущей ИС, доработка прототипа до работающего фрагмента ИС (*feedback, improved prototype design and development*).

Качественные изменения в ИТ

1. Феномен персональных вычислений
2. Феномен кооперативных технологий
3. Феномен компьютерных коммуникаций