

**Министерство образования и науки Российской Федерации**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего профессионального образования**  
**«Донской государственный технический университет»**

Утверждено на заседании кафедры  
«Строительная механика и теория сооружений»  
28 августа 2023 г.

**КУРС ЛЕКЦИЙ**  
**ИНФОРМАЦИОННОЕ МОДЕЛИРОВАНИЕ ЗДАНИЙ И СООРУЖЕНИЙ**

для самостоятельной работы для обучающихся заочной формы обучения по  
программам подготовки: 08.04.01 Все программы ПГС

Авторы:  
д.т.н., проф. Панасюк Л.Н.,  
К.т.н., доц., Труфанова Е.В.

Ростов-на-Дону  
2023

## Оглавление

<b>ОГЛАВЛЕНИЕ .....</b>	<b>1</b>
<b>1 РАЗВИТИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ .....</b>	<b>3</b>
1.1 ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ.....	7
1.2 ИСТОРИЧЕСКИЕ АСПЕКТЫ РАЗВИТИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ.....	18
1.2.1 РАЗВИТИЕ АППАРАТНЫХ СРЕДСТВ .....	18
1.3 ОСНОВНЫЕ ЭТАПЫ РАЗВИТИЯ ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ.....	23
1.3.1 ПРОГРАММИРОВАНИЕ В КОДАХ И АССЕМБЛЕР.....	23
1.3.2 ЯЗЫКИ ПРОГРАММИРОВАНИЯ ВЫСОКОГО УРОВНЯ .....	25
1.3.3 МОДУЛЬНОЕ ПРОГРАММИРОВАНИЕ.....	29
1.3.4 СТРУКТУРНОЕ ПРОГРАММИРОВАНИЕ .....	31
1.3.5 ОБЪЕКТНО – ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ.....	35
1.3.6 ПРОГРАММИРОВАНИЕ, УПРАВЛЯЕМОЕ СОБЫТИЯМИ .....	37
1.3.7 ВИЗУАЛЬНОЕ ПРОГРАММИРОВАНИЕ ИНТЕРФЕЙСА .....	38
1.3.8 ВЗАИМОДЕЙСТВИЕ ПРОГРАММ В ИНФОРМАЦИОННЫХ СИСТЕМАХ .....	40
1.3.9 БАЗЫ ДАННЫХ И СИСТЕМЫ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ .....	43
1.4 ВОПРОСЫ ПО РАЗДЕЛУ: .....	46
<b>2 ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ .....</b>	<b>48</b>
2.1 СТАДИИ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ.....	48
2.2 ОСНОВНЫЕ КЛАССИФИКАЦИИ ИНФОРМАЦИОННЫХ СИСТЕМ .....	53
2.3 СТАНДАРТИЗАЦИЯ ИС .....	56
2.4 ВОПРОСЫ ПО РАЗДЕЛУ: .....	64
<b>3 ИНФОРМАЦИОННЫЕ СИСТЕМЫ В СТРОИТЕЛЬСТВЕ.....</b>	<b>65</b>
3.1 СИСТЕМЫ САПР.....	65
3.1.1 САПР АРХИТЕКТУРНОГО НАЗНАЧЕНИЯ .....	65
3.1.2 САПР РАСЧЕТОВ НА ПРОЧНОСТЬ .....	68
3.1.2.1 Краткое описание программного комплекса по уточненному расчету зданий в процессе эксплуатации и выравнивания при использовании сетки конечных элементов высокой размерности.....	78
3.1.3 ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ ПРОГРАММНЫХ МКЭ-КОМПЛЕКСОВ .....	84
3.1.4 КОНТРОЛЬНЫЕ ВОПРОСЫ .....	93
3.2 СИСТЕМЫ АСУ .....	93
3.2.1 УПРАВЛЕНИЕ ОБЪЕКТАМИ НЕДВИЖИМОСТИ КАК УПРАВЛЕНИЕ СРЕДСТВАМИ ПРОИЗВОДСТВА .....	94
3.2.1.1 Общие положения .....	94
3.2.1.2 Предпосылки развития автоматизированных систем проектирования и управления средствами производства .....	95
3.2.1.3 Состояние проблемы управления объектами недвижимости .....	97
3.2.1.4 Цели и задачи программы «Управление объектами недвижимости как средствами производства» .....	99
СТРАТЕГИЧЕСКИЕ ЦЕЛИ.....	99
3.2.1.5 Концепция автоматизированной системы управления недвижимостью .....	102
3.2.1.6 Результаты внедрения программы .....	105
3.3 - ПОВЫШЕНИЕ НАДЕЖНОСТИ ЗДАНИЙ ИНФОРМАЦИОННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ ГОРОДСКИМ ХОЗЯЙСТВОМ.....	106
3.3.1 ВВЕДЕНИЕ.....	106
3.3.2 ЛОГИКА ВЗАИМОДЕЙСТВИЯ ДАННЫХ.....	107

3.3.3	СОСТАВ СИСТЕМЫ УПРАВЛЕНИЯ ГОРОДСКИМ ХОЗЯЙСТВОМ. ....	109
3.3.3.1	. Связанные с геоинформационной системой базы данных по основным городским объектам. ....	109
3.3.3.2	Связанные с геоинформационной системой базы данных о городских сетях .....	111
3.3.4	ПРОГРАММНЫЙ КОМПЛЕКС УПРАВЛЕНИЯ ОБЪЕКТАМИ НЕДВИЖИМОСТЬЮ. ....	115

## 1 Развитие информационных технологий

Сегодня невозможно практически представить сферу деятельности человека, в которой бы так или иначе не использовались информационные технологии (ИТ). Даже в самой гуманной (в смысле доступности только человеку, его интеллекту) сфере – искусстве (музыке, живописи, литературе, кинематографе) - применяются различные ИТ.

Человечество вступило в эру *информатизации*, и это проявляется в следующем:

- *информация* и информационные ресурсы на мировом рынке становятся важнейшим высокотехнологичным продуктом;
- фирмы, разрабатывающие *автоматизированные информационные технологии*, занимают ведущие позиции в мировой экономике, определяют дальнейшие направления развития конкурентоспособной продукции;
- без *информатизации* невозможно создание высоких *технологий*;
- *информационные технологии* (ИТ) открывают новые возможности в повышении эффективности производственных процессов, в сфере образования и быта, они выводят на новый уровень автоматизацию *технологических процессов* и управленческий труд, обеспечивают групповое ведение проектных работ, Интернет-технологии, CALS-технологии, дистанционное образование и т. д.;
- *информатизация* общества ведет к интернационализации производства.

Показателем научно-технической мощи страны становится внешнеторговый баланс профессиональных знаний, который реализуется рынком лицензий производственных процессов, "ноу-хау" и консультациями по применению наукоемких изделий. Например, США около 80% нововведений передают дочерним предприятиям в других странах. Пока эти предприятия осваивают предложенную *технологию*, в США готовят новые, т. е. реализуется опережающий технологический цикл высокоразвитой страны. К числу важнейших компонентов информационной мощи США относится глобальное лидерство в разработке, производстве и использовании *информационных технологий*.

Таким образом, эволюция мирового рынка дает преимущества стране, создающей у себя и передающей для производства другим странам наукоемкие изделия. Последние должны включать новые *технологии* и современные профессиональные знания. Идет торговля невидимым продуктом: знаниями, культурой; происходит навязывание высокоразвитыми странами стереотипа поведения. Именно поэтому в информационном обществе стратегическим ресурсом становятся *информация*, знание, творчество. Посредством дистанционного обучения, компьютерных игр, компьютерных видеофильмов и других ИТ компьютерные *технологии* оказывают огромное влияние на формирование условий и среды, в которых развиваются и процветают таланты. Предполагается, что социальное влияние информационной революции будет заключаться в синтезе западной и восточной мысли.

ИТ играют серьезную стратегическую роль в развитии каждой страны. Их значение быстро увеличивается за счет того, что ИТ:

- активизируют и повышают эффективность использования информационных ресурсов, обеспечивают экономию сырья,

энергии, полезных ископаемых, материалов и оборудования, людских ресурсов, социального времени;

- реализуют наиболее важные и интеллектуальные функции социальных процессов;
- занимают центральное место в процессе *интеллектуализации* общества, в развитии системы образования, культуры, новых (экранных) форм искусства, популяризации шедевров мировой культуры и истории развития человечества;
- обеспечивают информационное взаимодействие людей, способствуют распространению массовой *информации*;
- быстро ассимилируются культурой общества, снимают многие социальные, бытовые и производственные проблемы, расширяют внутренние и международные экономические и культурные связи, влияют на миграцию населения по планете;— оптимизируют и автоматизируют информационные процессы в период становления информационного общества;
- играют ключевую роль в процессах получения, накопления, распространения новых знаний по трем направлениям.

Первое из них — информационное моделирование, позволяющее проводить "вычислительный эксперимент" даже в условиях, которые невозможны при натуральном эксперименте из-за опасности, сложности и дороговизны.

Второе направление основано на методах искусственного интеллекта, оно позволяет находить решения плохо формализуемых задач, задач с неполной *информацией* и нечеткими исходными данными по аналогии с созданием метапроцедур, используемых человеческим мозгом.

Третье направление базируется на методах когнитивной графики, т.е. совокупности приемов и методов образного представления условий задачи, которые позволяют сразу увидеть решение либо получить подсказку для его нахождения. Оно открывает возможности познания человеком самого себя, принципов функционирования своего сознания. Кроме того, в этом случае становится возможным реализовать методы информационного моделирования глобальных процессов, что обеспечивает возможность прогнозирования многих природных ситуаций в регионах повышенной социальной и политической напряженности, экологических катастроф, крупных техногенных аварий.

Революционный скачок в развитии и популяризации ИТ за последние 20-25 лет напрямую связан с прорывом в технической базе – в переходе от «больших» ЭВМ, доступных узкому кругу специалистов, к персональным компьютерам. Сегодня «настольные» системы по возможностям значительно превосходят практически все «большие» ЭВМ 70-80-х годов, имеют выход во внешний мир и вместе с тем обладают интеллигентным и дружелюбным системным и прикладным программным обеспечением. Именно существенное развитие программного обеспечения, ставшее доступным на уровне интерфейса любому пользователю, является второй основной причиной массового внедрения ИТ. Например, сегодня задача копирования файла решается настолько просто, что ее решит любой начинающий пользователь компьютера. Но еще в середине восьмидесятых годов прошлого века для выполнения подобной операции требовалось привлечь специалиста, который составил бы специальный пакет-задание для вызова одной из системных утилит (например, IEFUJV, IEBCOPY или IEBGENER), состоящий из 10-15 системных управляющих «команд».

### 1.1 Основные определения

Ежегодно терминология в области ИТ пополняется новыми понятиями, аббревиатурами и т. п., поэтому в настоящем разделе приводятся лишь определения самого общего характера.

Сам термин *информация* происходит от латинского слова *information* — "разъяснение, осведомление, изложение". Понятие "*информация*" достаточно широко используется в обычной жизни современного человека, поэтому каждый имеет интуитивное представление о том, что это такое. Но когда наука начинает применять общеизвестные понятия, она уточняет их, ограничивает использование термина строгими рамками его применения в конкретной научной области. Так, понятие *информации*, становясь предметом изучения многих наук, в каждой из них конкретизируется и обогащается. Понятие *информации* является одним из основных в современной науке. Значение *информации* в жизни общества стремительно растет, меняются методы работы с *информацией*, расширяются сферы применения новых *информационных технологий*. Сложность явления *информации*, его многоплановость, широта сферы применения и быстрое развитие отражаются в постоянном появлении новых толкований понятий *информации* и *информационных технологий*. Поэтому имеются разные определения понятия *информации*, от наиболее общего, философского — "*Информация* есть отражение реального мира", — до узкого, практического — "*Информация* есть все сведения, являющиеся объектом хранения, передачи и преобразования". Приведем для сопоставления также некоторые другие определения и характеристики.

- *Информация* является одной из фундаментальных сущностей окружающего нас мира (акад. Г. Пospelов).

- *Информация* — сведения, передаваемые одними людьми другим людям устным, письменным или каким-нибудь другим способом (БСЭ).
- *Информация* является одним из основных универсальных свойств материи.

Под *информацией* необходимо понимать не сами предметы и процессы, а их отражение или отображение в виде чисел, формул, описаний, чертежей, символов, образов. Сама по себе *информация* может быть отнесена к области абстрактных категорий, подобных, например, математическим формулам, однако работа с ней всегда связана с использованием каких-нибудь материалов и затратами энергии. *Информация* хранится в наскальных рисунках древних людей в камне, в текстах книг на бумаге, в картинах на холсте, в музыкальных магнитофонных записях на магнитной ленте, в данных оперативной памяти компьютера, в наследственном коде ДНК в каждой живой клетке, в памяти человека в его мозгу и т. д. Для ее записи, хранения, обработки, распространения нужны материалы (камень, бумага, холст, магнитная лента, электронные носители данных). Кроме того, нужна энергия — например, для того, чтобы приводить в действие печатающие машины, создавать искусственный климат для хранения шедевров изобразительного искусства, питать электричеством электронные схемы калькулятора, поддерживать работу передатчиков на радио- и телевизионных станциях.

Термин *информатизация* может расшифровываться как эффективное использование обществом *информации* и средств вычислительной техники во всех сферах деятельности, как комплекс мер, направленных на обеспечение полного и своевременного использования достоверных знаний во всех общественно значимых видах человеческой деятельности. Основная цель



*информатизации* — обеспечение решения актуальных проблем общества, удовлетворение спроса на информационные продукты и услуги. Важность *информатизации* подчеркивается ее местом в "концепции четырех И", т. е. *информатизация, интеллектуализация, интеграция и индивидуализация*.

Под *интеллектуализацией* понимается создание и использование систем, решающих интеллектуальные задачи (накопление знаний и вывод новых, распознавание образов, общение с пользователем на естественном языке и т. д.). Интеграция предполагает комплексное решение научных, технических и социальных задач в целях развития общества.

Индивидуализация проявляется в развитии сегмента функциональных и личностных услуг во всех сферах человеческой деятельности.

Термин *технология* произошел от греческого *teche + logos*, т. е. "мастерство + учение". В производственном процессе под *технологией* понимают систему взаимосвязанных способов обработки материалов и приемов изготовления продукции. В общем случае *технология* — это правила действия с использованием каких-либо средств, которые являются общими для целой совокупности задач или задачных ситуаций. Если реализация *технологии* направлена на выработку управляющих воздействий, то это *технология управления*.

В узком смысле *технология* — это набор способов, средств выбора и осуществления управляющего процесса из множества возможных реализаций этого процесса. Под процессом (*processes* (лат.) — продвижение) здесь понимается функционально законченная, планируемая последовательность типовых операций со структурами данных, совершаемых за конечный промежуток времени в определенной среде, свойства которой диктуются требованиями и свойствами динамики процесса. В свою очередь, процесс может быть применен и к *информации* с целью ее преобразования.

В последнее время широкое распространение получили термины *безбумажная технология*, *интерактивная технология*, *технология программирования*, *технология проектирования баз данных*, *CALS (Continuous Acquisition and Lifecycle Support)-технология*, *сетевая технология*, *Internet-технология*, *технология анализа и реинжиниринга бизнес-процессов* и др. Все они предполагают использование *информации*, т.е. любого вида сведений о предметах, фактах, понятиях предметной области.

Современная *технология* должна отвечать следующим требованиям:

- высокая степень расчлененности процесса на стадии (фазы);
- системная полнота (целостность) процесса, который включает все элементы, обеспечивающие необходимую завершенность действий в достижении поставленной цели;
- регулярность процесса и однозначность его фаз, позволяющие применять средние величины при характеристике этих фаз, а следовательно, их стандартизацию и унификацию.

В понятии "*технология*" важно выделить два аспекта. Во-первых, *технология* неразрывно связана с процессом, т. е. совокупностью действий, осуществляемых во времени. Во-вторых, *технологический процесс* протекает в искусственных системах, созданных человеком для удовлетворения каких-либо потребностей.

В широком смысле под *технологией* понимают науку о законах производства материальных благ. В это понятие вкладывают три основные части:

- идеологию, т. е. принципы производства;
- орудия труда, т. е. станки, машины, агрегаты;

- кадры, владеющие профессиональными навыками.

Эти составляющие называют, соответственно, информационной, инструментальной и социальной. Другими словами, *информационный* аспект включает описание принципов и методов производства, *инструментальный* — орудия труда, с помощью которых реализуется производство, *социальный* — кадры и их организацию.

В более узком промышленном смысле *технология* рассматривается как последовательность действий над предметом труда в целях получения конечного продукта, например, *технология* получения интегральных схем или изготовления РЭС.

Для конкретного производства *технологию* понимают в узком смысле как совокупность приемов и методов, определяющих последовательность действий для реализации производственного процесса. Уровень *технологий* связан с научно-техническим прогрессом общества и влияет на его социальную структуру, культуру и идеологию. Для любой *технологии* могут быть выделены цель, предмет и средства.

*Целью технологии* в промышленном производстве является повышение качества продукции, сокращение сроков ее изготовления и снижение себестоимости.

Методология любой *технологии* включает в себя: декомпозицию производственного процесса на отдельные взаимосвязанные и подчиненные составляющие (стадии, этапы, фазы, операции); реализацию определенной последовательности выполнения операций, этапов и стадий производственного процесса в соответствии с целью *технологии*; технологическую документацию, формализующую выполнение всех составляющих.

Производство *информации* направлено на целесообразное использование информационных ресурсов и снабжение ими всех элементов организационной структуры и реализуется путем создания информационной системы. Информационные ресурсы являются исходным "сырьем" для системы управления любой организационной структурой. Конечным продуктом является принятое решение. Принятие решения в большинстве случаев осуществляется в условиях недостатка *информации*, поэтому степень использования информационных ресурсов во многом определяет эффективность работы организации.

В своем становлении любая отрасль, в том числе и информационная, проходила стадии от кустарного ремесленного производства к производству, основанному на высоких *технологиях*.

В развитии *технологии* выделяют два принципиально разных этапа. Один характеризуется непрерывным совершенствованием установившейся базисной *технологии* и достижением верхнего предельного уровня, когда дальнейшее улучшение является неоправданным из-за больших экономических вложений. Другой этап отличается отказом от существующей *технологии* и переходом к принципиально иной, развивающейся по законам первого этапа.

Под *информационными технологиями* понимается вся совокупность форм, методов и средств автоматизации информационной деятельности в различных сферах.

До настоящего времени не разработано общей теории *информационных технологий* (ИТ) как системы целостных взаимосвязанных приемов, методов и средств обработки *информации*, не определены основные понятия ИТ. Но достаточно понимать сущность ИТ, а также объяснить ее научное и практическое значение. Тем более что в проектировании и создании

конкретных ИТ переплетается много задач из различных научных дисциплин.

Как наука ИТ включает методологические и методические положения, организационные установки, методы использования инструментально-технических средств и т. д., — все то, что регламентирует и поддерживает информационное производство и деятельность людей, вовлеченных в это производство. Трансформация новых научных знаний в конкретную *информационную технологию* — основная задача ИТ как науки.

Ввиду дискуссионности предмета обсуждения приводится несколько понятий ИТ:

- ИТ — это совокупность научных методов и технических приемов производства информационных продуктов и услуг с применением всего многообразия средств вычислительной техники и связи;
- ИТ — это пограничная область, которая охватывает как вычислительную *технология*, так и конкретную социальную информационную практику, рационализирующую ее за счет широкого применения вычислительной техники;
- ИТ — это совокупность принципиально новых средств и методов, обеспечивающих создание, обработку, передачу, отображение и хранение *информации*.

Одновременно с широким использованием новых *информационных технологий* появилось понятие "*информационная система*" (ИС). **Информационная система** осуществляет сбор, передачу и переработку *информации* об объекте, снабжает работников различного уровня *информацией* для реализации функции управления.

Успешное внедрение *информационных технологий* связано с возможностью их типизации. Конкретная *информационная технология*

обладает комплексным составом компонентов, поэтому целесообразно определить ее структуру и состав.

Конкретная *информационная технология* определяется в результате компиляции и синтеза базовых технологических операций, специализированных технологий и средств реализации.

***Технологический процесс*** — часть информационного процесса, содержащая действия (физические, механические и т. д.) по изменению состояния *информации*.

*Информационная технология* базируется на реализации информационных процессов, разнообразие которых требует выделения базовых, характерных для любой *информационной технологии*.

***Базовый технологический процесс*** основан на применении стандартных моделей и инструментальных средств. Он может быть использован в качестве составной части *информационной технологии*. К их числу можно отнести: операции извлечения, транспортировки, хранения, обработки и представления *информации*.

Среди базовых технологических процессов выделяют:

- ***извлечение информации;***
- ***транспортирование информации;***
- ***обработку информации;***
- ***хранение информации;***
- ***представление и использование информации.***

Процесс извлечения информации *связан с переходом от реального представления предметной области к его описанию в формальном виде и в виде данных, которые отражают это представление.*

В процессе транспортирования *осуществляют передачу информации на расстояние для ускоренного обмена и организации быстрого доступа к ней, используя при этом различные способы преобразования.*

Процесс обработки информации *состоит в получении одних "информационных объектов" из других "информационных объектов" путем выполнения некоторых алгоритмов; он является одной из основных операций, выполняемых над информацией, и главным образом путем увеличения ее объема и разнообразия.*

Процесс хранения *связан с необходимостью накопления и долговременного хранения данных, обеспечением их актуальности, ценности, безопасности, доступности.*

Процесс представления и использования информации *направлен на решение задачи доступа к информации в удобной для пользователя форме.*

*Базовые информационные технологии строятся на основе базовых технологических операций, но кроме этого включают в себя ряд специфических моделей и инструментальных средств. Этот вид технологий ориентирован на решение определенного класса задач и используется в конкретных технологиях в виде отдельной компоненты. Среди них выделяют:*

- *мультимедиа-технологии;*
- *геоинформационные технологии;*
- *технологии защиты информации;*
- *CASE-технологии;*
- *телекоммуникационные технологии;*
- *COLS-технологии;*
- *технологии искусственного интеллекта.*

*Специфика конкретной предметной области находит отражение в специализированных информационных технологиях, например, организационное управление, управление технологическими процессами, автоматизированное проектирование, обучение и другие. Среди них наиболее прогрессивными являются следующие информационные технологии:*

- *организационного управления (корпоративные информационные технологии);*
- *в промышленности и экономике;*
- *в образовании;*
- *автоматизированного проектирования.*

Аналогом инструментальной базы (оборудование, станки, инструмент) являются средства реализации *информационных технологий*. Последние можно разделить на методические, информационные, математические, алгоритмические, технические и программные.

*CASE-технология (Computer Aided Software Engineering — Компьютерное Автоматизированное Проектирование Программного обеспечения) является своеобразной "технологической оснасткой", позволяющей осуществить автоматизированное проектирование информационных технологий.*

Методические средства определяют требования при разработке, внедрении и эксплуатации *информационных технологий*, обеспечивая информационную, программную и техническую совместимость. Наиболее важными из них являются требования по стандартизации.

Информационные средства обеспечивают эффективное представление предметной области; к их числу относятся информационные модели,



системы классификации и кодирования *информации* (общероссийские, отраслевые) и т. д.

Математические средства включают в себя модели решения функциональных задач и модели организации информационных процессов, обеспечивающие эффективное принятие решения. Математические средства автоматически переходят в алгоритмические, обеспечивающие их реализацию.

Технические и программные средства задают уровень реализации *информационных технологий* как при их создании, так и при реализации.

CALS-технология предназначена для унификации и стандартизации спецификаций промышленной продукции на всех этапах ее жизненного цикла.

Таким образом, конкретная *информационная технология* определяется в результате компиляции и синтеза базовых технологических операций, "отраслевых *технологий*" и средств реализации.

Внедрение информационных систем повышает эффективность производственно-хозяйственной деятельности предприятия за счет не только обработки и хранения *информации*, автоматизации рутинных работ, но и принципиально новых методов управления. Последние основаны на моделировании действий специалистов при принятии решений (методы искусственного интеллекта, экспертные системы и т. п.), использовании современных средств телекоммуникации (электронная почта, телеконференции), глобальных и локальных вычислительных сетей и т. д..

По сфере применения информационные системы классифицируются следующим образом:

- ИС для научных исследований;
- ИС автоматизированного проектирования;

- ИС организационного управления.

Научные ИС используются для автоматизации научной деятельности, анализа статистической *информации*, управления экспериментом.

ИС автоматизированного проектирования применяют для:

- разработки новых изделий и *технологий* их производства;
- различных инженерных расчетов;
- создания графической документации (чертежей, схем, графиков и т. д.);
- моделирования проектируемых объектов.

ИС организационного управления предназначены для автоматизации функций административного аппарата. К ним относятся ИС управления как промышленными предприятиями, так и непромышленными объектами (банками, биржами, страховыми компаниями, гостиницами и т. д.) и отдельными офисами (офисные системы).

ИС управления *технологическими процессами* создают для автоматизации различных *технологических процессов*.

## **1.2 Исторические аспекты развития информационных технологий**

### **1.2.1 Развитие аппаратных средств**

Понятие *информационная технология* возникло в последние десятилетия XX века в процессе становления информатики. Особенностью *информационной технологии* является то, что в ней и предметом, и продуктом труда является *информация*, а орудиями труда — средства вычислительной техники и связи. *Информационная технология* как наука о производстве *информации* возникла именно потому, что *информация* стала рассматриваться как вполне реальный производственный ресурс наряду с

другими материальными ресурсами. При этом производство *информации* и ее верхнего уровня — знаний — оказывает решающее влияние на модификацию и создание новых промышленных *технологий*.

**Информационная технология** — совокупность методов и способов получения, обработки, представления *информации*, направленных на изменение ее состояния, свойств, формы, содержания и осуществляемых в интересах пользователей.

Выделяют три уровня рассмотрения *информационных технологий*:

- первый уровень — теоретический. Основная задача — создание комплекса взаимосвязанных моделей информационных процессов, совместимых параметрически и критериально;
- второй уровень — исследовательский. Основная задача — разработка методов, позволяющих автоматизированно конструировать оптимальные конкретные *информационные технологии*;
- третий уровень — прикладной, который подразделяют на две страты: инструментальную и предметную.

Инструментальная *страта* (аналог — оборудование, станки, инструмент) определяет пути и средства реализации *информационных технологий*, которые можно разделить на:

- методические;
- информационные;
- математические;
- алгоритмические;
- технические;
- программные.

Предметная страта связана со спецификой конкретной предметной области и находит отражение в специализированных формационных *технологиях*, например организационное управление, управление *технологическими процессами*, автоматизированное проектирование, обучение и другие.

Потребность в передаче и обмене *информацией* человечество испытывало уже на ранних стадиях своего развития. Если сначала для ускорения передачи *информации* использовались костры, курьеры, потом почта, семафорный телеграф, то с изобретением электрического телеграфа и телефона принципиально изменились возможности передачи *информации*. Было изобретено радио и телевидение, а затем компьютер, цифровые системы связи и вычислительные сети; создание в 1978 году первого персонального компьютера явилось причиной быстрого его распространения и развития в качестве инструментального средства накопления, преобразования и передачи *информации* и позволило новым, *автоматизированным информационным технологиям* внедриться практически во все области человеческой деятельности. Интеграция достижений человечества в области средств связи, обработки, накопления и отображения *информации* способствовала формированию *автоматизированных информационных технологий* (АИТ).

Основу *автоматизированных информационных технологий* составляют следующие технические достижения:

- создание средств накопления больших объемов *информации* на машинных носителях, таких как магнитные и оптические диски;
- создание различных средств связи, таких как радио- и телевизионная связь, телекс, телефакс, цифровые системы связи, компьютерные сети,

космическая связь, позволяющих воспринимать, использовать и передавать *информацию* практически в любой точке земного шара;

- создание компьютера, особенно персонального, позволяющего по определенным алгоритмам обрабатывать и отображать *информацию*, накапливать и генерировать знания.
- *Автоматизированные информационные технологии* ориентированы на увеличение степени автоматизации всех информационных операций и, следовательно, на ускорение научно-технического прогресса общества.

Революционный скачок в развитии и популяризации ИТ за последние 20-25 лет напрямую связан с прорывом в технической базе – в переходе от «больших» ЭВМ, доступных узкому кругу специалистов, к персональным компьютерам. Сегодня «настольные» системы по возможностям значительно превосходят практически все «большие» ЭВМ 70-80-х годов, имеют выход во внешний мир и вместе с тем обладают интеллигентным и дружелюбным системным и прикладным программным обеспечением. Именно существенное развитие программного обеспечения, ставшее доступным на уровне интерфейса любому пользователю, является второй основной причиной массового внедрения ИТ. Например, сегодня задача копирования файла решается настолько просто, что ее решит любой начинающий пользователь компьютера. Но еще в середине восьмидесятых годов прошлого века для выполнения подобной операции требовалось привлечь специалиста, который составил бы специальный пакет-задание для вызова одной из системных утилит (например, IEFUJV, IEBCOPY или IEBGENER), состоящий из 10-15 системных управляющих «команд».

Огромный толчок развитию *информационных технологий* принесла разработка мультимедийных средств.

*Мультимедиа* — это объединение нескольких средств представления информации в одной системе. Обычно под *мультимедиа* подразумевается объединение в компьютерной системе таких средств представления информации, как текст, звук, графика, мультипликация, видеоизображения и пространственное моделирование. Термин "*мультимедиа*" стал популярным сравнительно недавно, в связи с появлением мощных недорогих компьютеров.

В настоящее время имеются настольные компьютеры, способные работать со звуковой и видеоинформацией, манипулировать ею для получения специальных эффектов, синтезировать и воспроизводить звуки и видеоинформацию, создавать все виды графической информации, включая анимационные изображения, и объединять все это в едином представлении *мультимедиа*. Представления с использованием средств *мультимедиа* являются захватывающими, так как они многомодальны, т. е. одновременно воздействуют на несколько органов чувств и поэтому вызывают повышенный интерес и повышенное внимание у аудитории. Такое объединение сред обеспечивает качественно новый уровень восприятия информации: человек не просто пассивно созерцает, а активно участвует в происходящем. Именно этот феномен участия, а также технологические успехи производителей определил и мультимедийный бум последних лет.

*Информационные технологии* обеспечивают переход от рутинных методов к промышленным методам и средствам работы с информацией в различных сферах человеческой деятельности, давая возможность рационально и эффективно ее использовать.

### 1.3 Основные этапы развития технологий программирования

Массовая разработка и внедрение современных информационных технологий была бы невозможна без кардинального изменения инструмента разработки ИТ – *программирования*.

Наверное, стоит вначале рассмотреть вопросы развития и совершенствования инструментария для разработки ИТ, т.к. именно без существенного изменения инструментария невозможно было получить в массовом масштабе новые информационные системы. С другой стороны, возможности и стандарты в технологии программирования так или иначе трансформируются в возможности продукта – информационные системы, влияют на структуру любой информационной системы.

#### 1.3.1 Программирование в кодах и ассемблер

Любая программа для ЭВМ – системная или прикладная – воспринимается (распознается) процессором только в том случае, если она состоит из специальных команд, коды которых известны процессору определенного типа. Команды записаны в памяти компьютера в специальном формате. Каждая команда состоит из ее кода, определяющего действие (сложение, вычитание и т.д.) и адресов операндов, над которыми это действие необходимо выполнить. В вольном переводе на русский язык некоторую команду можно, например, интерпретировать так: *сложить два числа, находящиеся в памяти по адресам 100 и 120*. В компьютере команды хранятся обычно в виде набора двоичных чисел. Часть набора определяет сущность команды (ее код), остальная часть определяет адреса операндов.

Разные типы ЭВМ имеют отличные друг от друга способы кодировки команд. Так, на персональных IBM-совместимых компьютерах<sup>1</sup> некоторая команда сложения может иметь вид: 0000 0011 1100 0011. А на "древних"

---

<sup>1</sup> Кстати, само понятие *совместимых компьютеров* означает, что они "понимают" одинаковую систему команд, хотя могут строиться на различной элементной базе.

компьютерах типа М-220 команда сложения двух чисел могла выглядеть так: 001 00000001100100 00000001111000 00000001111011. Поэтому программа в кодах компьютера является машинно-зависимой и непереносимой, т.е. подготовленная для компьютера одного типа, она не сможет выполняться на других. Этот факт определяет основной недостаток программирования в кодах.

Вторым недостатком программирования в кодах является сильное дробление программы. Дело в том, что логически команды процессора достаточно примитивны и обуславливают выполнение простейших операций. Так, программирование несложной формулы  $x=(a+b)(c+d)$  требовало задания серии команд типа:

- Сложить  $a$  и  $b$ , промежуточный результат записать в  $a_1$ ,
- Сложить  $c$  и  $d$ , промежуточный результат записать в  $a_2$ ,
- Умножить  $a_1$  на  $a_2$ , результат записать в  $x$ .

Первоначально программирование велось в кодах. Визуально каждая программа состояла из большого количества команд-строк, похожих на приведенные выше двоичные коды. Это определяло третий недостаток программирования в кодах - затрудненную читаемость программы и, как следствие, сложность исправления (отладки) или доработки программы.

Однако программированию в кодах присущи и значительные плюсы. Программист управляет всеми ресурсами компьютера, полностью контролирует текущее состояние ЭВМ, выбирает наиболее оптимальный код команды. Самые короткие по объему и наиболее быстрые по выполнению программы или их фрагменты разрабатываются и сегодня в кодах. Для облегчения наглядности программы в кодах разработаны специальные символические языки – *ассемблеры*. В них каждой команде компьютера сопоставляется определенный символьный код, являющийся сокращением "родных" для человека слов. Специальная программа (она также называется



ассемблером) переводит (*транслирует*) "непонятную" для компьютера (но более понятную для человека) символьный строку в коды компьютера. Так, приведенные выше коды команд сложения на ассемблере могли выглядеть так: *ADD AX, BX* (сложить числа из регистров *AX* и *BX* и результат запомнить в *AX*). При программировании на ассемблере человек может оперировать не с адресами памяти, в которых хранятся данные, а с их символическим представлением. Например, вначале ассемблеру специальной инструкцией сообщается, что по такому-то адресу хранится число, названное для программиста как *a*. Далее человек не задумывается над тем, по какому адресу находится соответствующее число, но просто использует его имя *a*.

Ассемблер является машинно-зависимым языком программирования, так как его инструкции соответствуют кодам команд компьютера. Поэтому ассемблерная программа может выполняться только на тех ЭВМ, для которых она разрабатывалась. Кроме того, для работы на ассемблере требуется детальное знание особенностей конкретной ЭВМ.

### 1.3.2 Языки программирования высокого уровня

Развитие программирования связано с разработкой в 50-х годах так называемых алгоритмических языков программирования. Их называют языками программирования высокого уровня, а также машинно-независимыми языками. Программа записывается с помощью специальных слов почти так же, как на разговорном языке (в основном – английском). От программиста не требуется детального знания аппаратных особенностей ЭВМ. Сложные формулы имеют вид, практически соответствующий стандартам математики. Поэтому при использовании языков программирования высокого уровня человек больше внимания уделяет сути задачи, алгоритму ее решения, а не деталям реализации на ЭВМ. Программы

на машинно независимых языках сравнительно легко переносятся между компьютерами разных типов.

Естественно, что ЭВМ не "понимает" программу, написанную на алгоритмических языках. Поэтому для каждого языка и для каждого типа ЭВМ разрабатывается специальная программа–переводчик, которая транслирует исходный текст в последовательность команд конкретного компьютера. Эти специальные программы называются *трансляторами* (несколько позже они получили название *компиляторов*).

Первые алгоритмические языки назывались Algol (сокращение от *Algorithmic Language* – язык алгоритмов) и Fortran (сокращение от *Formula Translation* – транслятор формул). Эти языки обладали рядом сходных возможностей. К ним относятся:

- простота записи математических выражений. Так, на Algol'е приводимая ранее формула записывается как  $x := (a+b) * (c+d);$ , а на Fortran'е так:  $X = (A+B) * (C+D)$ .
- использование *ветвления* в программе за счет механизма ***условных выражений и операторов перехода***, позволяющих выполнять тот или иной фрагмент программы в зависимости от результатов предыдущих этапов (например, алгоритм определения корней квадратного уравнения зависит от знака дискриминанта), и ***циклов***, позволяющих осуществить многократное повторение некоторого фрагмента программы (например, табуляция значений функции).
- использование ***массивов***, позволяющие программировать операции над матрицами и векторами.
- использование ***процедур*** (в Fortran'е- *подпрограмм*) и ***функций***, позволяющих выделить в отдельный блок некоторый логически законченный алгоритм над абстрактными данными (*формальные*

*параметры*) и многократно обращаться к ним с указанием конкретных данных (*фактические параметры*). Так, например, для многократного использования тригонометрических функций следует запрограммировать алгоритм их вычисления от абстрактного аргумента (например,  $X$ ), а затем многократно их использовать, указывая, что на место абстрактного  $X$  подставляются конкретные данные –  $Y$ ,  $Z$ ,  $F_i$ ,  $30$  и т.д. Или запрограммировать процедуру обращения квадратной матрицы. Алгоритм записывается для некоторой абстрактной матрицы (скажем,  $A$ ), а затем при их вызове указывается конкретная матрица.

Кроме общих моментов, эти языки имели отличия. Мало существенным является отличие в синтаксисе, т.е. в разной форме записи аналогичных инструкций – циклов, условных операторов и т.д. Существенным оказалось наличие двух принципиальных отличий.

Первое – способ управления объемом резервируемой под данные программы памяти. Готовая и оттранслированная программа используется многократно, с заданием различных данных. Предположим, что выполняются операции над матрицами. В разных вариантах счета мы могли использовать матрицы разных размеров. Алгол позволял выделять для хранения массивов память во время выполнения программы. Поэтому размер выделяемой памяти определялся уже во время счета, и ее отводилось столько, сколько необходимо для хранения конкретного массива. После использования памяти некоторым массивом ее можно было освободить и использовать под другие. А Фортран распределял память в начальный момент, при старте программы. Поэтому программа на Фортране не в состоянии распределить память в момент выполнения, когда можно вычислить ее требуемый объем. Так же нельзя освободить память после того,

как вспомогательные матрицы или векторы уже не будут использоваться. Способ распределения памяти при старте программы называется **статическим управлением** памятью. Распределение памяти в произвольный момент выполнения называется **динамическим управлением**. Итак, Алгол мог использовать как *статическое*, так и *динамическое* управление памятью. В Фортране реализован только механизм *статического* управления. Поэтому в программах на Фортране либо указывались размеры массивов по максимуму, либо программу требовалось исправлять и вновь транслировать при каждом запуске. В условиях малого объема доступной памяти статическое управление являлось существенным отрицательным фактором для программ на Фортране.

Второе отличие связано с особой структурой организации программы на Фортране. Дело в том, что программа на Фортране может состоять из отдельных сегментов (которые называются **модулями** и обычно хранятся в отдельных файлах). Любой из них может транслироваться самостоятельно, независимо от остальных. Окончательный вариант программы формируется из отдельных модулей, как блочное здание из отдельных законченных фрагментов. Программа же на Алголе представляла единый неразрывный текст, и транслировать ее можно было только целиком. За счет модульной структуры программу на Фортране могли разрабатывать несколько программистов. Каждый работал над своим модулем, из которых затем составлялась цельная программа. Один и тот же модуль мог входить в состав различных программ. Ряд фирм разрабатывал и продавал специализированные модули (объединенные в библиотеки подпрограмм), которые могли быть использованы другими разработчиками. Т.е., благодаря модульной структуре коммерческую ценность приобрели не только законченные программы, но и наборы специализированных библиотек. Например, разработчики программ расчета строительных конструкций могли

приобрести и использовать в своих программах библиотеку решения проблем линейной алгебры, библиотеку численного решения дифференциальных уравнений. Разработчики экономических и управленческих программ могли использовать библиотеки корреляционного анализа, библиотеки статистики и т.п. Важным здесь являются два момента:

1. Библиотеки модулей поставляются в оттранслированной форме, т.е. в машинных кодах. Поэтому определенным образом скрыты особенности использованного в модулях алгоритма, или *Now-How* разработчиков.
2. Специализированные модули разрабатываются специалистами по соответствующей проблематике, что определяет использование современных и оптимальных алгоритмов.

### 1.3.3 Модульное программирование

Увеличение роста специализированных библиотек определило дальнейшее развитие концепции и языков программирования. Этот стиль получил название **модульного программирования**. Лучшие аспекты Алгола и Фортрана (динамическое распределение памяти и модульная структура программ), а также множество других положительных особенностей были заложены в языке *PL/1* (сокращение от *Programming Language* – язык программирования). Его стандарт был разработан в конце 60-х годов, а массовое использование на "больших" ЭВМ IBM-360/370 (аналог в СССР – ЕС ЭВМ) началось со второй половины 70-х и продолжалось минимум 10-15 лет. ПЛ/1 в то время был наиболее развитым языком программирования. Наряду с модульной структурой, динамическим распределением памяти, укрупненными операциями над массивами, работой с текстовыми данными, в него были включены новые возможности: переменные файлового типа, работа с адресами памяти (тип *pointer* - указатель), объединение в одно

целое данных разных типов – структуризация данных. Развитие концепции модульного программирования обусловило возможность объединения в одну программу модулей, написанных на разных языках программирования. Одни модули могли быть написаны на ПЛ/1, другие – на Фортране, третьи – на ассемблере. При стыковке модулей важно было лишь корректно передать параметры, учитывая специфику реализации механизма их передачи в разных языках.

Произошло и развитие «старых» языков. Попытка реанимировать Алгол под модульное программирование привела к появлению языков АЛГАМС и Алгол-68, был разработан язык Фортран-77. Так же был предложен стандарт языка *ADA* для проектирования систем.

Однако развитие техники привело к дальнейшему развитию методов программирования.

Совершенствование элементной базы ЭВМ, увеличение объема памяти и скорости процессора, появление графопостроителей и дисплейных станций, а так же большой объем научных разработок в области алгоритмов решения общематематических и прикладных задач, увеличение количества готовых библиотечных модулей (в том числе модулей графического и текстового отображения информации) позволили перейти от разработки отдельных прикладных программ или их пакетов к разработке программных комплексов и информационных систем.

В свою очередь, развитие аппаратных средств обусловили изменение отношения к компьютерной технике в обществе. ЭВМ и программное обеспечение из инструмента научных исследований превращались в удобные инструментальные средства управления, проектирования, прогнозирования. Наблюдается стремительное увеличение спроса на ЭВМ и на новые прикладные программные продукты. Так, на рубеже 70-х – 80-х годов фирма

IBM не могла удовлетворить все заказы на поставку больших ЭВМ. Объем заказов превышал возможности производства на десять лет вперед. Так же и спрос на программистов существенно превышал объем их подготовки.

#### 1.3.4 Структурное программирование

Первые программы состояли из нескольких десятков – сотен машинных команд. Программные комплексы и информационные системы включали в себя от нескольких десятков тысяч до миллионов операторов на языках высокого уровня. По мере увеличения объема программных средств на первое место по влиянию на скорость их разработки вышел процесс *отладки*, т.е. обнаружение и исправление допущенных ошибок. Практика показала, что написать более-менее большую программу без ошибок практически невозможно. Более того, даже во время эксплуатации готовых программ обнаруживаются пропущенные при отладке ошибки. Ошибки делятся на несколько категорий.

Самая простая – синтаксические ошибки. Они связаны с неверным написанием конструкций языка и обнаруживаются сразу, в тот момент, когда компилятор пытается перевести программу на машинный язык

Вторая категория ошибок связана с *компоновкой* программы, т.е. стадией построения программы из отдельных оттранслированных модулей. Ошибки в основном здесь связаны с неверными именами процедур или функций.

Если исправлены все синтаксические ошибки и ошибки компоновки, то программа начинает выполняться (стадия *Run*). Однако эта стадия может быть прервана недопустимой операцией (например, делением на ноль), и в этом случае говорят об ошибках времени выполнения (*RunTime Errors*). Либо программа досчитала до конца, но полученные результаты оказываются неверными. Вызвать это могут ошибки алгоритма, либо ошибки организации

межмодульных связей, ошибки преобразования типов данных, либо простые описки (например, пропущен ноль при записи числа 100 и программа использовала в вычислениях 10). Ошибки времени выполнения обнаруживаются решением серии тестовых примеров (*прогонка программы*), результат для которых известен заранее. Как правило, время, затраченное на поиск подобных ошибок, существенно превышает время написания самой программы и исправления синтаксических ошибок. Развитие концепции программирования тесно связано с проблемой снижения брака при написании программы.

Программирование стало не только средством, а так же и предметом научного изучения. Из ремесла программирование превратилось в академическую дисциплину. В работах Э.Дейкстры и К.Хоора определен новый взгляд на программирование как на предмет научного изучения и поле интеллектуальной деятельности. Хоор показал, что программы поддаются точному анализу, основанному на математических выкладках. Убедительно показано, что можно избежать многих ошибок программирования, если программисты осознанно будут применять те методы и приемы, которые ранее использовались интуитивно. Этот подход получил название «революции» в программировании. Разрабатываются теоретические основы *структурного программирования, теория доказательства правильности программы, доказательное программирование*.

Концепция *структурного программирования* предполагает такой научно обоснованный подход к построению программы и к возможностям языка программирования, при котором будет существенно снижен объем возможных ошибок. Существенное развитие концепции достигнуто в работах Н.Вирта, завершившихся разработкой языка структурного программирования *Pascal*. Основная идея состоит в краткой фразе, принятой



Н.Виртом в качестве заголовка одной из его книг: «АЛГОРИТМЫ + СТРУКТУРЫ ДАННЫХ = ПРОГРАММЫ», т.е. «...программы представляют собой в конечном счете конкретные формулировки абстрактных алгоритмов, основанных на конкретных представлениях и структурах данных». Показана тесная связь между использованными алгоритмами и формой представления данных. Решения о структурировании данных нельзя принимать без знания алгоритмов, которые будут обрабатывать эти данным. Наоборот, выбор алгоритмов существенным образом зависит от используемой в программе структуры данных.

Суть структурного программирования состоит в том, что данные представляют собой, прежде всего абстракции реальных объектов и формулируются предпочтительно, как абстрактные структуры, которые могут не иметь конкретной реализации в языке программирования. В процессе конструирования программы представление данных постепенно уточняется вслед за уточнением алгоритма.

Язык структурного программирования должен предоставлять возможность конструирования собственных типов данных. На низшем уровне стоят predetermined в языке *фундаментальные структуры* (например: *скалярные типы, запись, массив и множество*). Из фундаментальных структур, как из молекул, строят *усложненные структуры*. Переменные фундаментальной структуры могут менять только значение, сохраняя тип или множество допустимых значений и занимаемый ими размер. Усложненные структуры характеризуются изменением не только значения, но и самой формы представления, составом входящих в них фундаментальных структур. Динамическое управление памятью реализуется на уровне указателей, и динамические данные объединяются в древовидную структуру (*деревья*).

Кроме того, предъявляются определенные требования к типизации данных, за счет чего тип любого выражения или переменной можно определить без необходимости проведения вычислений, т.е. на стадии трансляции. При этом можно избежать большого количества трудно находимых ошибок времени выполнения за счет выявления их на стадии трансляции программы.

Концепция структурного программирования предполагает не только использование специальных языков и применения определенного стиля программирования. В первую очередь это - философия программирования, которая затрагивает и такой аспект, как *методика обучения* программистов. Отмечается, что программирование представляет собой обширную и разнообразную деятельность, часто требующей сложной умственной работы. Неверно считать, что программирование можно свести к использованию готовых рецептов, основанных на систематическом описании простейших блоков. У учеников необходимо развивать способность творчески мыслить, и здесь чрезвычайно важна роль учителя, показывающего, как он делает то-то и то-то. Творчество не подвластно канонам, методикам и технологиям. В качестве метода обучения принимается тщательный выбор и рассмотрение характерных примеров. Необходимо рассматривать сложные и длинные программы, какие часто и встречаются на практике и лучше всего подходят для выявления того неуловимого, но важного свойства, которое называется *стилем* программирования. Разбор длинных программ служит упражнением в искусстве их чтения, которое не менее важно, чем умение писать программы.

### 1.3.5 Объектно – ориентированное программирование

Объектно – ориентированное программирование (**ООП**) – логически следующая за структурным ступень развития. ООП начинает бурно развиваться с конца восьмидесятых годов. Начало девяностых отмечено повсеместным внедрением новых объектно – ориентированных языков программирования. К их числу в первую очередь относится C++ и объектно – ориентированный *Pascal*. Развитие ООП привело к такому существенному изменению стиля программирования, что даже новые операционные системы стали разрабатываться под влиянием этих концепций. Так, в системах Windows 9x/NT многие сущности ориентированы на ООП. Например, одним из назначений реестра Windows является регистрация в ОС *классов*, входящих в состав программных комплексов.

*Классы* (в некоторых версиях Pascal они названы *объектами*) – основные элементы ООП, расширяющие понятие типов данных и объединяющие (*инкапсулирующие*) в единое целое данные (в ООП они называются *поля*) и программный код (в ООП – *методы*), обрабатывающий их. Здесь структура данных объединена в единое целое с основными методами, осуществляющими доступ к ним и их обработку. После разработки класса программист, использующий его, не нуждается в знании о структуре данных внутри класса. Если изменить структуру данных и при этом соответственно изменить внутренние алгоритмы обработки этих данных, то программист, использующий класс во внешних модулях, не заметит изменения структуры. Если в традиционном структурном программировании изменение структуры данных приведет к необходимости переработать практически все модули, использующие эту структуру, то в ООП переработка затронет только методы класса. Таким образом, ООП является расширением концепции структурного программирования.

Базируется ООП на следующих основных принципах:

- *инкапсуляция* (описана выше),
- *наследование*,
- *полиморфизм* (многообразность),
- в последнее время к основным понятиям относят *скрытие данных*

Механизм **наследования** позволяет использовать структуры данных и методы уже разработанного класса в новых классах, которые имеют много общего со старым, но отличаются от него некоторыми деталями. Старый является *родительским* классом, а новый – *дочерним* классом. Если бы при создании нового класса все приходилось бы начинать с нуля, то эффективность использования классов могла быть поставлена под сомнение. Однако, построив генеалогическую структуру реальных объектов, выделив их общие стороны и отличия, программист может программно реализовать эту структуру благодаря наследованию. При этом для дочернего класса он опишет только его отличительные признаки, а все общее дочерний класс автоматически «заимствует» у родительского.

**Полиморфизм** подразумевает возможность для конкретного экземпляра объекта во время выполнения принимать образ различных классов программы. Это позволяет организовать массив объектов разного типа, и обрабатывать в головной программе каждый из них по единому алгоритму, не задумываясь над тем, какому классу реально принадлежит очередной объект.

*Скрытие данных* запрещает внешней программе оперировать непосредственно с полями класса. Чтение или запись данных в поля можно выполнять только посредством специальных методов, которые называются *свойствами*.

### 1.3.6 Программирование, управляемое событиями

Раньше программе соответствовал некоторый жесткий алгоритм, единственным образом определявшим последовательность выполнения операторов. Ветвление программ было прозрачным для программиста – в программе жестко задавались условия переходов в ту или иную точку.

Программирование, управляемое событиями, является развитием ООП и сегодня воспринимается в едином контексте с моделью ООП. Однако этот механизм был включен в программирование только в начале девяностых годов двадцатого века, уже после основного внедрения ООП.

В программировании, управляемом событиями, приложение – это не последовательность каких-то операторов, не некий жесткий алгоритм. Здесь программа выступает как совокупность объектов и способов их взаимодействия. При этом в роли главного объекта во многих случаях можно считать самого пользователя программы. Он же является и основным, но не единственным *источником событий*, которые управляют работой приложения.

Программные объекты как бы существуют в некоторой среде взаимодействия друг с другом. Основой этой среды являются сообщения, которые генерируются в результате наступления каких-то событий. События могут порождаться действиями пользователя (например, при работе с мышкой или клавиатурой), но могут наступать и в результате других причин (внешних). Например, отключение питания, получение письма по электронной почте или локальной сети, сбой дисководов и т.д. Так же события могут являться результатом работы самой программы, результатом работы ее объектов. Например, программа моделирует процесс движения нескольких тел, и в случае «столкновения» каких-либо из них генерируется событие с посылкой соответствующего сообщения. В каждом объекте программист

определяет множество событий, на которые объект должен реагировать. При получении соответствующего сообщения, объект реагирует на него «автоматически», запуская на выполнение соответствующий метод. В этом контексте объект можно определить как *совокупность свойств и методов, а также событий, на которые он должен реагировать и сообщений, которые он может генерировать*.

Теперь код программы состоит из отдельных, непосредственно не связанных друг с другом фрагментов. Последовательность их выполнения определяется не явно (т.е., не операторами перехода внутри программы), но соответствующей настройкой программной среды. Например, если отключить реакцию кнопки ДА в какой-то форме на щелчок мышки, то при выполнении программы эта кнопка станет недоступной. Однако программный текст до и после отключения реакции кнопки останется неизменным.

### 1.3.7 Визуальное программирование интерфейса

Надо заметить, что ранее в большинстве программ интерфейс с пользователем отсутствовал. Пользователь должен был составить текстовый файл исходных данных, используя инструкцию к программе, а далее программа обрабатывала эти данные без вмешательства пользователя. При изменении данных требовалось составить новый или откорректировать имеющийся файл данных. Однако сегодня невозможно представить себе ни одной серьезной программы, у которой отсутствует интерактивный интерфейс с пользователем. Качественная разработка подобного интерфейса позволяла «оживить» программу, предоставив пользователю удобные функции управления процессом ее выполнения.

Однако разработка удобного интерфейса с пользователем – т.е. системы окон, кнопок, меню, горячих клавиш и т.д. – являлась одной из

наиболее трудоемких задач при программировании. Программирование «вручную»<sup>2</sup> интерфейса и сервиса занимало до 90% объема текста программы. Т.е., объем смысловой части программы, объем того, ради чего она и разрабатывалась, становился гораздо меньшим, чем объем вспомогательной, сервисной части. Соответственно и дорогостоящие ресурсы труда программистов использовались в основном для выполнения достаточно рутинной работы.

Одним из подходов снижения трудоемкости при разработке стандартных интерфейсов был связан со стандартизацией и развитием специальных функций API (Application Program Interface – стандартные библиотеки процедур и функций, входящие в состав операционной системы, которые программист может использовать в своем приложении).

Вторым, революционным шагом, явилось появление *визуального программирования*, появившегося в Visual Basic и значительно развитого в системах Delphi и C++ Builder. Этот шаг позволил серьезно развить так называемую CASE – технологию (Computer Aided Software Engineering – автоматизированное проектирование программного обеспечения).

Визуальное программирование позволило свести проектирование пользовательского интерфейса к серии простых и наглядных операций, позволяющих программисту оформить внешний вид своей программы практически так же просто, как выполнить его прорисовку в несложных графических редакторах. При этом программист видит оформление своей программы еще на стадии разработки, до ее запуска на выполнение.

Более того, именно в этой среде программист проводит настройку реакций визуальных объектов (кнопок, переключателей и т.д.) на различные

---

<sup>2</sup> Под программированием вручную интерфейса понимается разработка специального программного кода, который размещает элементы интерфейса (меню, кнопки) и определяет последовательность переходов при обработке связанных с ними событий (действий с мышкой, клавиатурой и т.п.)

события. То есть проектирование сводится к размещению компонентов на форме, заданию некоторых их свойств (например: цвет, подпись) и написанию обработчиков необходимых событий.

Благодаря визуальному объектно – ориентированному программированию, управляемому событиями, создана RAD – технология (Rapid Application Development – быстрая разработка приложений). Благодаря этой технологии, программисты имеют гораздо больший резерв времени для программирования сути задачи, для оптимизации своего приложения, для вариантной разработки программных проектов.

### 1.3.8 Взаимодействие программ в информационных системах

Сказанное выше относилось к технологиям разработки отдельных программ. Но в настоящее время программы, как правило, должны являться частью некоторой *информационной системы (ИС)*, т.е. функционировать не сами по себе, а во взаимодействие с большим количеством другим программ. Также программы должны уметь «использовать» другие программы и системы, разработанные сторонними организациями. Например, ваша программа может подготовить некоторый отчетный документ в формате Microsoft Word, и этот документ будет доступен на всех компьютерах, на которых установлен Word (но на которых может не быть вашей программы). Аналогично, можно из прикладной программы создать итоговые чертежи в формате распространенных графических систем (например, в формате AutoCAD или Corel Draw). В этом случае один из основных вопросов – организация взаимного общения программ друг с другом и с источником данных. В качестве последнего в ИС выступают *базы данных (БД)*, вместе с *системой управления базами данных (СУБД)*. Программы, работающие в составе ИС, получают информацию из БД, к которым имеют доступ и другие



программы. В этом случае естественным образом создается возможность взаимосвязи приложений через данные. Например, одна программа записывает результаты своей работы, а другая использует их как начальные данные для своей работы. Этот простейший уровень взаимосвязи требует одного – унификацию данных и форматов их хранения. Для целей унифицированного доступа к данным используются специальные языки, например *SQL* (*Structured Query Language* – язык структурированных запросов).

Но во многих случаях этого простейшего механизма общения недостаточно для современной ИС – например, программа не должна ожидать, когда кем-то будет запущена другая программа – поставщик данных. Необходимо иметь возможность запускать из одной программы другую, передавая при этом ей управляющую информацию. Запуск основного приложения порождает в среде операционной системы *процесс*, для которого операционная система выделяет необходимые ресурсы компьютера (память и время процессора). Дочерняя программа может выполняться как в адресном пространстве вызвавшей ее программы, так и в собственном адресном пространстве и в другом *потоке*.

Однако часто требуется обмен информацией между программами, выполняющимися одновременно (параллельно). Желательно, чтобы этот обмен не зависел от языка программирования, на котором написаны разные программы, а в сетевых системах не зависел и от операционных систем, установленных на разных компьютерах. Яркий пример подобной организации взаимосвязи – Интернет, в глобальную сеть соединены компьютеры с различными операционными системами (Windows, Unix, Solaris, сотовые телефоны и др.).

Простейшими средствами параллельного общения вначале были файлы *совместного доступа*, или *разделяемые файлы* (файлы, к которым могут иметь одновременный доступ несколько программ), которые появились еще на заре Windows. Также ранним средством является буфер обмена Clipboard, доступный практически всем приложениям Windows, в котором можно временно хранить для передачи другим программам информацию различного формата – текстовую, графическую и т.п. Несколько позже для межпрограммного взаимодействия использовалась технология **DDE** – *динамический обмен данными*. Сегодня актуальность DDE ниже из-за появления новых технологий, а использование БД, буфера обмена и разделяемых файлов как простейшего варианта межпрограммного взаимодействия остается актуальным.

Позднее появилась технология *связывания и внедрения объектов* **OLE1** (*Object Linking and Embedding*). Благодаря OLE1 появилась возможность создавать составные документы (например, в документ Word вставлять таблицу Excel, при ее редактировании из Word используются возможности Excel). На смену ей пришла технология **OLE2**, позволяющая различным программам предоставлять друг другу свои функции (*сервис*). Программа, предоставляющая свои функции, называется *сервером*, а программа, их использующая – *клиентом*. В этой технологии одна программа может не просто вызвать другую, но использовать ее отдельные функции.

Следующим шагом на пути совершенствования межпрограммного обмена явилась технология *компонентной модели объектов* (**COM** – *Component Object Model*). Эта технология заключается в стандартизированном описании служб программы, к которым она дает доступ другим программам. В технологии COM неважно, на каких языках написана программы, и где они выполняются: в одном потоке, в разных

потоках, на разных компьютерах. Расширение этих возможностей дает технология **DCOM** – распределенная модификация COM. Отметим, что COM подразумевает взаимосвязь на уровне специальных *объектов*, структура которых во многом схожа с рассмотренными ранее объектами внутри одной программы.

Необходимо сказать о еще одной стремительно развивающейся технологии – Интернет. В Интернете располагаются и базы данных, и серверы, с которыми общаются приложения пользователя.

#### 1.3.9 Базы данных и системы управления базами данных

Говоря о развитии технологий программирования, невозможно обойти вниманием тему разработки и управления базами данных. *Базами данных (БД)* называют электронные хранилища информации, доступ к которым осуществляется с помощью одного или нескольких компьютеров. БД создается для хранения и доступа к данным, содержащим сведения о некоторой предметной области, т.е. некоторой области человеческой деятельности или окружающего нас мира. *Системы управления базами данных (СУБД)* – набор программных средств, предназначенных для создания, наполнения, обновления, выборки, формирования отчетов и удаления информации в базах данных.

Классификацию современных СУБД можно выполнить по различным признакам.

**По назначению** различают три основных вида СУБД: промышленные универсального назначения, промышленные специального назначения и разрабатываемые для конкретного заказчика. Специализированные СУБД создаются для управления БД конкретного назначения – бухгалтерские, складские и т.д. Универсальные СУБД не имеют четко обозначенных границ

применения, рассчитаны на произвольные задачи. Поэтому они достаточно сложны и требуют от пользователя специальных знаний.

**По размещению** отдельных частей БД различают *локальные* и *сетевые* СУБД.

Все части локальных СУБД размещены на одном компьютере. Если с локальной БД должны работать несколько пользователей одновременно, то на их компьютерах должны храниться точные копии этой БД. Существенной проблемой при этом является *синхронизация* копий данных. Если один пользователь внес изменения в свою копию БД, то они станут доступными на других компьютерах только после их копирования. Сегодня применение локальных БД ограничено.

Сетевые СУБД делятся на *файл-серверные*, *клиент-серверные* и *распределенные*. Обязательным атрибутом сетевых СУБД является сеть, обеспечивающая аппаратную и программную связь компьютеров и позволяющая корпоративно работать множеству пользователей с одними и теми же данными.

В файл – серверных СУБД данные размещаются на одном достаточно мощном компьютере, постоянно подключенном к сети. Такой компьютер называется *файл-сервером*. Достоинством такой СУБД является относительная простота ее создания, обслуживания и настройки. Заметим, что между локальной и файл-серверной СУБД практически нет различий, т.к. все фрагменты программного обеспечения СУБД (за исключением данных) находятся на всех компьютерах сети. Отличие между ними состоит в том, что программы обращаются к данным через сеть. Для многих вариантов архитектуры сети на уровне прикладного программного обеспечения это различие “скрыто” от программиста. Из этого вытекает основной недостаток файл-серверных СУБД. Он связан со значительной нагрузкой на сеть, т.к.

любое обращение к данным, их корректировка вызывает пересылку значительных объемов информации по сети. Например, если пользователю требуется выбрать несколько фамилий из списка персонала по определенному признаку, то на его компьютер вначале пересылается весь список, и затем поиск производится на пользовательском компьютере. При увеличении числа пользователей СУБД пропускной способности сети становится недостаточно для объемов пересылаемой информации, что приводит к существенному замедлению скорости работы, а иногда и к системным сбоям.

В клиент-серверных БД нагрузка на сеть существенно снижена. Клиентская программа работает с данными через специализированное программное обеспечение (посредника) – *сервер базы данных*. Сервер размещается на машине с данными. Клиентская программа посылает серверу запрос, сервер принимает его, отыскивает в данных нужные записи и передает их клиенту. Поэтому через сеть передаются только сравнительно короткие запросы и только ограниченное количество найденных записей, а не вся таблица. Запрос к серверу формируется на специализированном *языке структурированных запросов* – *SQL*. Поэтому часто серверы БД называют *SQL-серверами*. Серверные программы являются достаточно сложными, производятся различными фирмами. Наиболее известными являются серверы:

- *Microsoft SQL Server* компании Microsoft,
- *Sybase SQL Server* компании Sybase,
- *Oracle* компании Oracle<sup>3</sup>,
- *DB2* компании IBM,

---

<sup>3</sup> О значительном объеме продаж SQL-серверов свидетельствует тот факт, что Oracle занимает второе место по объемам доходов среди компаний, занимающихся только разработкой и продажей программного обеспечения (первое место у Microsoft)

- *InterBase* компании Inprise.

Разновидностью клиент-серверных БД являются *трехзвенные системы*. В них используется промежуточное программное обеспечение между клиентом и сервером, которое называется *сервером приложений*. Назначение сервера приложений состоит в том, чтобы избавить клиента от многих рутинных аспектов работы с данными, в результате клиентские программы становятся более компактными и содержат в основном код смысловой обработки данных (такие клиентские программы называют *тонкие клиенты*).

Распределенные БД включают большое количество серверов, а число клиентских мест может достигать сотен тысяч. Обычно такие БД работают в организациях государственного масштаба, данные разделены между серверными компьютерами по территориальному признаку. Информация на отдельных серверах может дублироваться для снижения вероятности отказов и сбоев и максимального обеспечения сохранности наиболее важной информации.

#### **1.4 Вопросы по разделу:**

- *Что характеризует эру информатизации?*
- *Дайте определение понятия "информация". В чем состоят ее особенности?*
- *Раскройте понятие "технология" и ее аспекты.*
- *Что явилось причиной возникновения понятия "информационные технологии"?*
- *Какие достижения человечества обусловили появление автоматизированных информационных технологий?*
- *Что такое информационная система?*

- *Каковы цель, методы и средства автоматизированной информационной технологии?*
- *Что понимается под программированием в кодах?*
- *Что Вы знаете о языке ассемблер?*
- *Какие языки программирования называют машинно-независимыми?*
- *Какие языки высокого уровня Вам известны? Какими владеете?*
- *Что понимается под модульным программированием и отдельным модулем?*
- *Что Вы знаете о структурном программировании?*
- *Что Вы знаете об объектно-ориентированном программировании?*
- *Какие основные принципы ООП?*
- *Что дает механизм наследования в ООП?*
- *Что такое полиморфизм в ООП?*
- *Что такое инкапсуляция в ООП?*
- *Возможности программирования, управляемого событиями?*
- *Какие современные языки программирования Вам известны? Какими владеете?*
- *Что Вы знаете о принципах взаимодействия программ в информационных системах?*
- *Что понимается под БД и СУБД? В чем их различие?*
- *Классификация СУБД по назначению и по размещению.*

## 2 Проектирование информационных систем

### 2.1 Стадии проектирования информационных систем

Разработка качественной информационной системы для нужд конкретного предприятия – сложный творческий процесс. Одним из аспектов является то, что непосредственно программисты не являются специалистами по технологии, организации и экономике данного предприятия или технологического процесса. Поэтому крайне важна качественная связь между специалистами-предметниками и специалистами-программистами. Обеспечение этой связи – одна из задач руководителя подразделения, для которого создается ИС. Поэтому он должен ясно представлять основные этапы разработки информационной системы.

Проектирование любой информационной системы осуществляется в несколько стадий. В общем случае следует выделить следующие:

- *предпроектное обследование;*
- *технико-экономическое обоснование;*
- *составление технического задания;*
- *техническое проектирование;*
- *рабочее проектирование.*

Две последние стадии могут для небольших проектов быть объединены одну - технорабочее проектирование.

Прежде чем начинать проектирование, необходимо выполнить обследование объекта, для которого создается ИС. Это достаточно важный этап, так позволяет выделить характерные особенности объекта, которые следует учесть в характеристиках разрабатываемой ИС и которые определяют дальнейшую работу по проектированию. Любой процесс проектирования (и проектирования ИС в частности) является итерационным процессом, когда неоднократно приходится возвращаться на предыдущие



этапы проектирования коррекции имеющихся результатов. От качества предпроектного обследования во многом зависит, придется ли в дальнейшем пересматривать основные концепции создаваемой ИС и вносить в нее принципиальные изменения, что всегда является трудоемкой задачей. На этапе предпроектного обследования следует сразу же настраиваться на то, что любое предприятие имеет свою специфику в производственных и бизнес-процессах. Поэтому знания о других предприятиях и о стандартных правилах организации этих процессов могут служить в большей части подспорьем в изучении предприятия, но отнюдь не являться целью внедрения. Обследование сводится к анализу существующей системы и объекта, для которого создается система. В частности, при этом следует уделять особое внимание общению на предприятии с экспертами и специалистами в интересующей предметной области, а также анализу документов и их движения. Обследование (сбор материалов) выполняется по двум основным направлениям: обоснованию эффективности создаваемой системы и выбору технических средств.

Материалы для обоснования эффективности создаваемой системы включают в себя:

- *структуру существующей системы;*
- *объемы выполняемой работы и трудозатраты;*
- *качество выполняемой работы;*
- *методы выполнения работы;*
- *ведение документации и др.*

Данные для выбора технических средств включают в себя:

- *структуру объекта;*
- *технология передачи информации, системы оперативной и диспетчерской связи;*

- *сбор исходных данных;*
- *наличие вычислительной техники;*
- *систематизацию и оформление документов.*

В результате обследования должны быть получены и отражены в пояснительной записке следующие материалы, которые затем будут использованы в процессе проектирования:

- *общая характеристика объекта, для которого создается ИС;*
- *функции, выполняемые в системе: периодичность выполнения, трудозатраты на их выполнение и т. д.;*
- *характеристика используемой информации;*
- *существующие принципы действия системы;*
- *быстродействие системы;*
- *структурные схемы существующей системы (организационная, функциональная, алгоритмическая и др.);*
- *необходимые информационные потоки: виды документов, маршруты их движения и т. д.*

На основании изучения объекта формируется перечень задач, которые должна решать ИС. Обычно процесс создания ИС является многоэтапным, и должны быть предусмотрены возможности ее развития. Предпроектное обследование позволяет наметить состав первой очереди системы и дальнейшие пути ее совершенствования.

Технико-экономическое обоснование создания ИС содержит следующие моменты:

- *исходные положения, характеристики и технико-экономические данные об объекте*
- *обоснование цели создания ИС;*

- обоснование комплекса задач, решаемых в ИС, и др.;

Технический проект содержит материалы, дающие представление о составе и функционировании ИС, и включает в себя:

- общую характеристику объекта, для которого создается ИС;
- организацию управления в условиях использования ИС;
- используемый комплекс технических средств;
- описание и постановку решения задач, входящих в ИС;
- описание стандартного программного обеспечения;
- описание организации информационной базы и т. д.

Главное назначение технического проекта - определение основных направлений действия создаваемой системы, затрат, экономической эффективности, требуемых технических и программных средств, штатов обслуживающего персонала.

Рабочий проект включает в себя документацию, необходимую для внедрения и функционирования системы:

- документацию по используемым и разработанным программам (кстати, документация по разработанным программам может послужить прообразом справочной системы — см. 12);

- инструкции по обработке данных (сбор, регистрация, обработка и передача информации);

- должностные инструкции персонала и т. д.

Следует обратить внимание на инструкцию для администратора БД - технического специалиста, который будет поддерживать работоспособность БД. В ней, кроме операций по архивированию, регистрации новых пользователей и т. п., обязательно должны быть описаны действия при различных сбоях в работе БД - от полного выхода из строя компьютера, где находится БД, до проблем, возникающих у пользователя при подключении

БД. Кроме того, администратор обязательно должен знать структуру БД, поэтому желательно создавать ее с подробным, включая комментарии, описанием всех таблиц и их полей.

Технический и рабочий проекты включают в себя следующие собственные этапы, которые, как правило, выполняются в указанной ниже последовательности:

- *выбор технических средств и стандартного программного обеспечения с учетом следующих особенностей:*

- *используемых в организации программных и аппаратных средств, а также других информационных систем;*

- *перспектив развития информационных технологий в организации (например, переход к работе с помощью Internet-технологий);*

- *структуры организации и требований к безопасности информации;*

- *уровня знаний и возможностей разработчиков;*

- *создание ИС и БД;*

- *создание программного обеспечения:*

- *создание средств ввода, корректировки и удаления информации;*

- *создание средств поиска информации;*

- *создание средств отображения информации, включая формирование отчетов;*

- *обеспечение контроля вводимой информации (выполняется параллельно с другими этапами создания программного обеспечения);*

- *создание средств администрирования БД;*

- *обеспечение работы программного обеспечения в сети;*

- *создание справочной системы (желательно выполнять параллельно с другими этапами технорабочего проектирования);*

- *локализация программного обеспечения;*

- *формирование рабочего варианта программного обеспечения (удаление отладочной информации, создание ярлыка программы и т. д.);*
- *разработка системы сбора информации;*
- *создание инструкций по работе с системой.*

Безусловно, на количество и объем приведенных здесь этапов влияет, пожалуй один из самых важных критериев - стоимость разработки.

## **2.2 Основные классификации информационных систем**

Несмотря на значительное количество различных информационных систем, общая классификация их по назначению достаточно узкая.

В общем можно выделить следующие направления ИС:

- операционные системы,
- АСУ – автоматизированные системы управления,
- САПР – системы автоматизированного проектирования,
- ГИС – геоинформационные системы,
- Связь и телекоммуникация,
- Справочно-поисковые системы,
- Системы информационной безопасности,
- Системы подготовки и обработки мультимедийной информации (звука, изображения, видео),
- редакционно-издательские системы.

В отдельных системах могут сочетаться различные комбинации базовых. Например, АСУ магистральных газопроводов будет включать в себя и ГИС, и АСУ ТП (автоматизированную систему управления технологическими процессами), и элементы телекоммуникаций и т.п.

Несмотря на достаточно узкую классификацию по основным направлениям, внутри каждого может быть множество разновидностей.

Одно из разделений - по роду деятельности (машиностроение, торговля, строительство). Например, АСУ могут подразделяться на:

- *Системы автоматизации бухучета,*
- *Автоматизация управлением делопроизводства,*
- *Автоматизация управления торгами,*
- *Автоматизация управления банками,*
- *Автоматизация управления торговлей,*
- *Автоматизация таможенной деятельности,*
- *Автоматизация управления технологическими процессами,*
- *Автоматизация управлениями объектами недвижимости и т.д.*

Или, САПР делятся на:

- *САПР в строительстве,*
- *САПР в машиностроении,*
- *САПР в электронной промышленности,*
- *САПР в авиастроительстве и т.д.*

Другое разделение соответствует назначению системы. Так, например, системы САПР могут разделяться на:

- *системы подготовки чертежной документации,*
- *системы расчета на прочность, жесткость и устойчивость,*
- *системы подготовки проектно-сметной документации,*
- *системы подготовки документации на конкурс и т.п.*

Кроме того, следует рассмотреть деление по возможности пересечения видов деятельности. При этом необходимо рассматривать системы *общего* и *специализированного* назначения. Например, такие системы разработки чертежной документации, как AutoCAD и MicroStation являются системами САПР общего назначения. Опираясь общими графическими примитивами (отрезками, дугами, размерными линиями и т.п.) пользователь может подготовить чертежную документацию для любой отрасли промышленности. Наоборот, САПР ArchiCAD, speedikon, ArCON являются специализированными для строительства и здесь пользователь оперирует не общими, а специализированными примитивами-объектами, как то: стены, окна или проемы, лестницами и т.д. С помощью этих систем можно быстрее и качественнее подготовить проектную документацию по объекту строительства, чем с помощью систем общего назначения. Однако подготовить проектную документацию для строительства корабля или самолета практически невозможно. Аналогично обстоит дело с САПР расчета на прочность. Например, системы ANSYS и NASTRAN – системы общего назначения, с их помощью можно рассчитать хоть здание, хоть самолет. А вот система *ProFET & Stark ES* ориентированна на расчет здания, с ее помощью можно быстрее и более «профильно» рассчитать здание. А вот при расчете самолета эти САПР лучше не использовать.

Заметим, что вокруг оболочки программ САПР общего назначения создаются десятки различных расширений возможностей. Многие компьютерные фирмы разрабатывают подсистемы к программам общего назначения, предоставляющие пользователю больший круг возможностей для использования общей системы в конкретной отрасли промышленности.

Вместе с тем, на рынке программных продуктов существует множество различных ИС сходного назначения. Так, для автоматизации бухгалтерского

учета сегодня предлагаются системы «1С», «Инфо-бухгалтер», «Парус», «Инотек НТ», «Gendalf», «Овионт информ», «Камин», «Плюс-микро», «СБиС++» и многие другие. Успех той или иной системы на рынке зависит порой не только от качества программного продукта, но и от грамотно организованной маркетинговой и рекламной политики фирмы, от организации разветвленной сети дилеров и технического сопровождения. Аналогичное многообразие программных продуктов наблюдается и в других сферах деятельности человека.

### **2.3 Стандартизация ИС**

Естественно, что многообразие сходных ИС порождает требования унификации и стандартизации многих параметров, начиная от интерфейса общения системы с пользователем и заканчивая протоколами обмена данными. Разработка, обновление и поддержка различных стандартов в информационных технологиях – еще одна крупная отрасль информационных технологий. Проблемами стандартизации ИС занимаются многие исследовательские организации, в т.ч. и специальные отделы в крупных компьютерных фирмах.

Рассмотрим, для примера, развитие стандартов АСУ для производственных предприятий.

Первый стандарт управления бизнесом - MPS (master planning scheduling)- обеспечивавший объемно - календарное планирование путем формирования объема продаж и его разбивки по календарным периодам с последующим определением плана пополнения запасов готовой продукции за счет производства или закупки и формированием отчета о планируемом движении денежных средств появился в 60-х годах.

Однако этого стандарта было вполне достаточно только для мелкого и простого производства с небольшой номенклатурой изделий и



комплектующих и ненапряженным, статичным графиком продаж. Обнаружилось множество производств с существенно динамичным, статистическим поведением потоков продаж и закупок, с громадной номенклатурой закупаемых комплектующих, имеющих различные задержки выполнения заказов на поставки. Для них пришлось разрабатывать, во-первых, методы статистического управления запасами (SIC - statistical inventory control), во-вторых, методы планирования поставок комплектующих, получивших название MRP (material requirement planning). При этом по сути впервые в полном объеме потребовались компьютерные ресурсы, т.к. вручную расчеты по соответствующим алгоритмам реализовать было просто нереально.

Соответствующий стандарт MRP появился в 70-х годах. Он обеспечивал минимизацию издержек, связанных со складскими запасами, определяя объемы и сроки заказа на поставки комплектующих на основании данных прогноза продаж, спецификации изделия, технологической карты и состояния запасов.

Однако его серьезным недостатком было отсутствие учета и планирования производственных ресурсов (мощностей) (учет и планирование загрузки оборудования и персонала, учет стоимости рабочего и машинного времени и на их основании расчет планируемых и учет фактических затрат и т.п.). Когда это осознали, достаточно быстро была разработана методология планирования производственных мощностей -CRP (capacity requirement planning). Алгоритм планирования здесь похож на алгоритм MRP, но существенно сложнее вследствие значительно большего количества учитываемых параметров, и требует значительных вычислительных ресурсов. Поэтому в целях ускорения расчетов дополнительно была разработана методология чернового планирования

производственных мощностей (RCCP), которая учитывает только ключевые ресурсы, поэтому работает значительно быстрее, чем полная процедура расчета и используется для отработки производственного графика. Совокупность методологий MPS-SIC- MRP-RCCP-CRP получила название MRP II (manufacturing resource planning).

Стандарт MRP II со временем превратился в стандарт планирования ресурсов предприятия (Enterprise Resource Planning - ERP), называемый иногда планированием ресурсов в масштабе предприятия (Enterprise-wide Resource Planning). В основе ERP лежит принцип создания единого хранилища данных (repository), содержащего всю деловую информацию, накопленную организацией в процессе ведения деловых операций, включая финансовую информацию, данные, связанные с производством, управлением персоналом, или любые другие сведения. Это устраняет необходимость в передаче данных от системы к системе. Кроме того, любая часть информации, которой располагает данная организация, становится одновременно доступной для всех работников, обладающих соответствующими полномочиями.

Концепция ERP стала очень известной в производственном секторе, поскольку в результате ее применения согласованная работа всех подсистем предприятия обеспечивает:

- *уменьшение запасов*
- *улучшение сроков поставки*
- *сокращение времени издержек на производство и реализацию продукции повышение качества продукции*
- *ускорение реакции на предприятия на изменения во внешней среде и условиях производства*

- *повышение уровня предоставляемого сервиса клиентам увеличение доходности бизнеса.*

Главные функции, общие для современных программных продуктов класса MRP II и ERP:

- *Прогнозирование продаж, запасов, закупок*
- *Финансовое планирование*
- *Объемно - календарное планирование производства*
- *Планирование потребности в материалах*
- *Планирование потребности в производственных мощностях*
- *Управление цехом*
- *Управление запасами (статистическое)*
- *Управление закупками*
- *Управление продажами*
- *Бухгалтерский и финансовый (управленческий) учет*
- *Управление спецификациями изделий и технологических процессов*

Типовое распределение подфункций названных функций по основным MRP подсистемам таково:

- *MPS*
- *описание плановых единиц и уровней планирования*
- *описание спецификаций планирования*
- *формирование основного производственного плана графика MRP*
- *управление изделиями (описание материалов, комплектующих и единиц готовой продукции)*
- *управление запасами*
- *управление конфигурацией изделия (состав изделия)*

- *ведение ведомости материалов*
- *расчет потребности в материалах*
- *формирование MRP заказов на закупку*
- *формирование MRP заказов на перемещение CRP*
- *рабочие центры (описание структуры производственных рабочих центров с определением мощности)*
- *машины и механизмы (описание производственного оборудования с определением нормативной мощности)*
- *производственные операции, выполняемые в привязке к рабочим центрам и оборудованию*
- *технологические маршруты, представляющих последовательность операций, выполняемых в течение некоторого времени на конкретном оборудовании в определенном рабочем центре*
- *расчет потребностей по мощностям для определения критической загрузки и принятия решения*

Финансовое планирование как часть бизнес - планирования в системе MRP II обеспечивается специальным инструментарием формирования финансового плана и составления бюджетных смет, прогнозирования и управления движением денежных средств, на основании которых определяется возможность реализации производственного плана с точки зрения наличных и предполагаемых денежных средств.

ERP системы (Enterprise Resources Planning), как дальнейшее развитие интегрированных информационных систем управления предприятием кроме вышеуказанной функциональности, включают управление ресурсами для проведения технологического обслуживания и выполнения ремонтов и

людскими ресурсами (кадры и зарплата), а также, как правило, управление ресурсами (DRP) и цепочками поставок (SCM),

Системы управления ресурсами DRP обеспечивают оптимальное решение (планирование, учет и управление) транспортных задач по перемещению материально-технических ресурсов и готовой продукции с распределенной системой складирования.

Системы управления и цепочками поставок SCM (Supply Chain Management) поддерживают управление внешней логистикой - т.е. системой материальных потоков в сети поставщиков сырья и комплектующих.

Кроме этого, в MRPII и ERP системах часто присутствует (в особенности для дискретного производства) специальная подсистема управления реализацией долгосрочных проектов (Project Management), предполагающая полнофункциональное планирование материальных ресурсов, трудовых ресурсов, оборудования, формирования сетевых графиков работ, управление ходом выполнения реализуемых проектов. При этом обеспечивается согласование процессов разработки изделий, подготовки производства и собственно производства. Как правило, в этих случаях реализуются интерфейсы ERP - системы с системами САПР/CAD/CAM. Для технологически сложных производств обеспечивается интерфейс с системами АСУ ТП, причем производители этих систем со своей стороны зачастую включают в комплект поставки интерфейсные модули для известных "тяжелых» ERP- систем SAP\R3 и BAAN. В ERP-системах широко применяются средства штрих - кодирования, которые используются для автоматизации взвешивания и учета складских запасов.

Соответствующая интегрированная база данных состоит из двух частей: производственной и финансовой.

Производственная часть содержит сведения о материальном производстве, включая сведения о материалах и продукции (спецификации изделий и их структуры), о структуре и технологии работы складов для каждого вида запасов, о производстве (спецификации технологических процессов, оборудования и рабочих центров), информацию о планируемых, прогнозируемых и фактических состояниях и событиях, нормативные значения контролируемых показателей.

Финансовая часть базы данных отражает производственно - экономическую структуру деятельности предприятия и содержит все данные, необходимые для так называемого управленческого учета, задачей которого является предоставление руководству предприятия необходимой и достаточной информации для принятия управленческих решений. Информационная система настраивается и работает таким образом, что обеспечивает отражение полного комплекса финансово-хозяйственных операций предприятия и всех участков бухгалтерского учета. Все хозяйственные операции, зарегистрированные в информационной системе, формируют финансовые операции, которые передаются в финансовую часть базы данных для использования в финансовых модулях системы. При этом поток финансовой информации структурируется в соответствии с введенной структурой учетных регистров (счетов). Структуризация включает определение:

- *состава и структуры объектов и направлений учета, включая центры прибыли и затрат, компании, подразделения, рабочие центры и пр.;*
- *состава и структуры учетных регистров - детальные и сводные, включающие план счетов отечественного бухучета и систему аналитических счетов по объектам и направлениям учета;*

- *структуры балансов по счетам, включающей категории учета факт, прогноз, план, обязательство, бюджет;*
- *правила разnosки и консолидации по счетам;*
- *состава и структуры контрольных и целевых показателей;*
- *состава и структуры первичных и вторичных документов*

Структура и содержание финансовой части базы данных проектируется для обеспечения следующих задач, реализуемых финансовыми модулями:

- *финансовая бухгалтерия - расчеты с дебиторами и кредиторами,*
- *прочие расчеты и расчеты по налогам*
- *производственная бухгалтерия - учет материальных запасов, учет производственных затрат*
- *учет основных средств - все операции по учету и движению основных средств*
- *управленческая бухгалтерия - финансовая интеграция и отчетность.*

Настройки на конкретное применение обычно включают определение структуры предприятия, соответствующего плана счетов, структуры складов, описании хранимых запасов, хозяйственных операций и соответствующих проводок, стратегии статистического управления запасами, центров затрат, описания технологического оборудования, спецификаций изделий и технологий производства, настройки алгоритмов планирования, прогнозирования и управления всех уровней. Этот процесс достаточно сложный в связи с сильной взаимосвязанностью функций настраиваемых блоков и для сокращения сроков и стоимости настройки обычно поддерживается специальными программно - методическими средствами поддержки внедрения.

Исторически сложилось так, что в нашей стране одной из наиболее динамично развивавшихся отраслей ИТ были информационные технологии в строительстве. Следует отметить, что одним из наиболее приоритетных направлений здесь являлись разработка методов и алгоритмов сложных инженерных расчетов для комплексной системы «здание – фундамент-основание». Именно в этой области наибольшие успехи отечественных ученых в развитии теории и методов расчета сооружений применительно к информационным технологиям в строительстве.

#### **2.4 Вопросы по разделу:**

- *Какие стадии проектирования ИС?*
- *Классификация ИС.*
- *Принципы стандартизации ИС.*
- *Что включает в себя предпроектное обследование?*
- *Какие данные включаются в технико-экономическое обоснование ИС?*
- *Что включено в техническое задание на ИС?*
- *Какова взаимосвязь заказчика и разработчика на стадии технического проектирования ИС?*
- *Назначение систем САПР? Какие системы МАПР Вам известны?*
- *Что такое справочно – поисковые системы?*
- *Что такое системы ГИС, их назначение?*
- *Что понимается под системами АСУ, их классификация?*
- *Что Вам известно о стандартизации ИС?*



### 3 Информационные системы в строительстве

#### 3.1 Системы САПР

По своему назначению САПР в строительстве подразделяется на :

- архитектурно – планировочные и дизайнерские ИС,
- системы расчета сооружений на прочность, жесткость и устойчивость,
- системы подготовки конструкторской документации,
- системы проектирования внутренних сетей,
- системы подготовки проектно-сметной документации,
- системы подготовки документации на тендер,
- ИС технологии и организации производства.

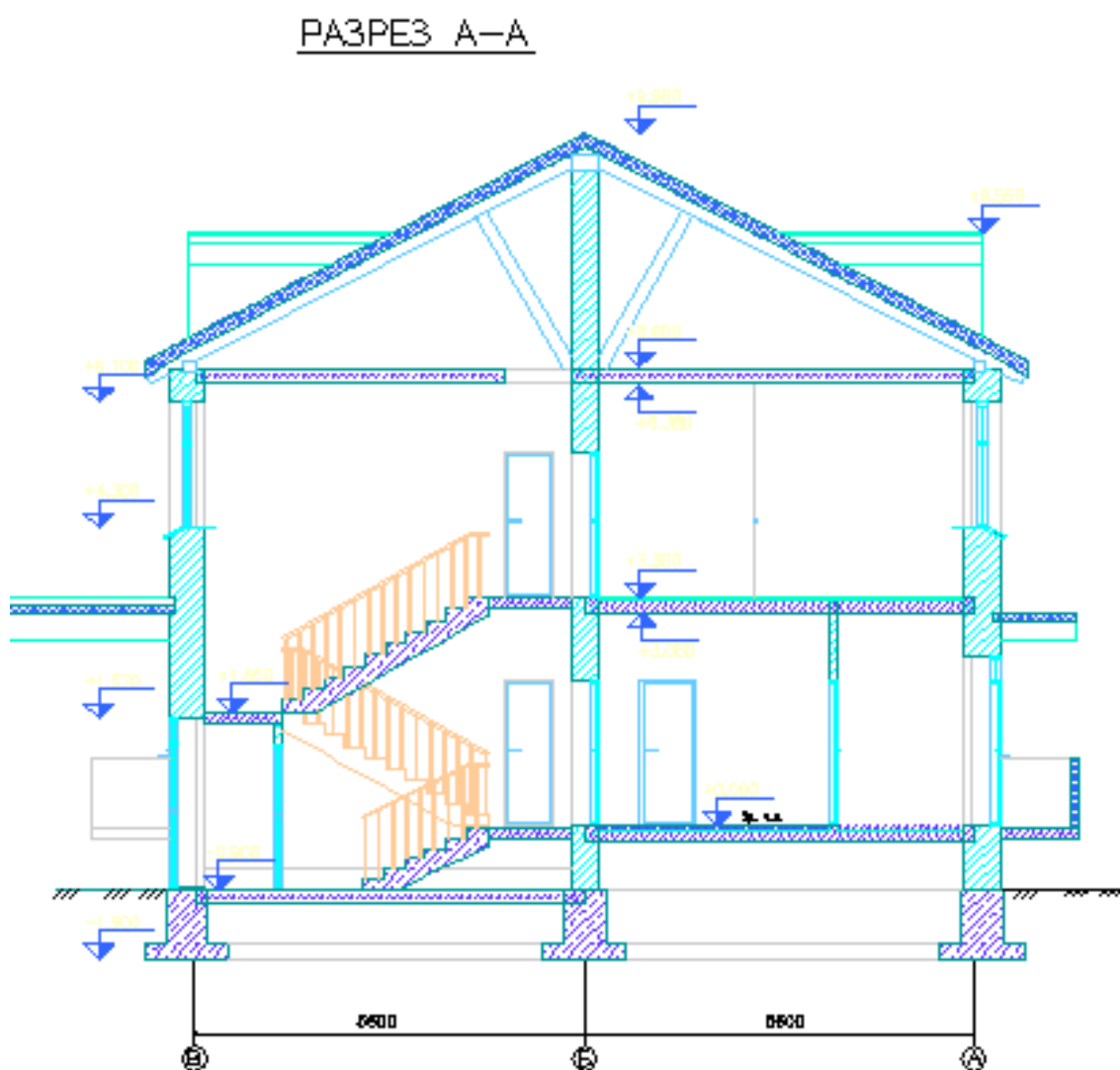
##### 3.1.1 САПР архитектурного назначения

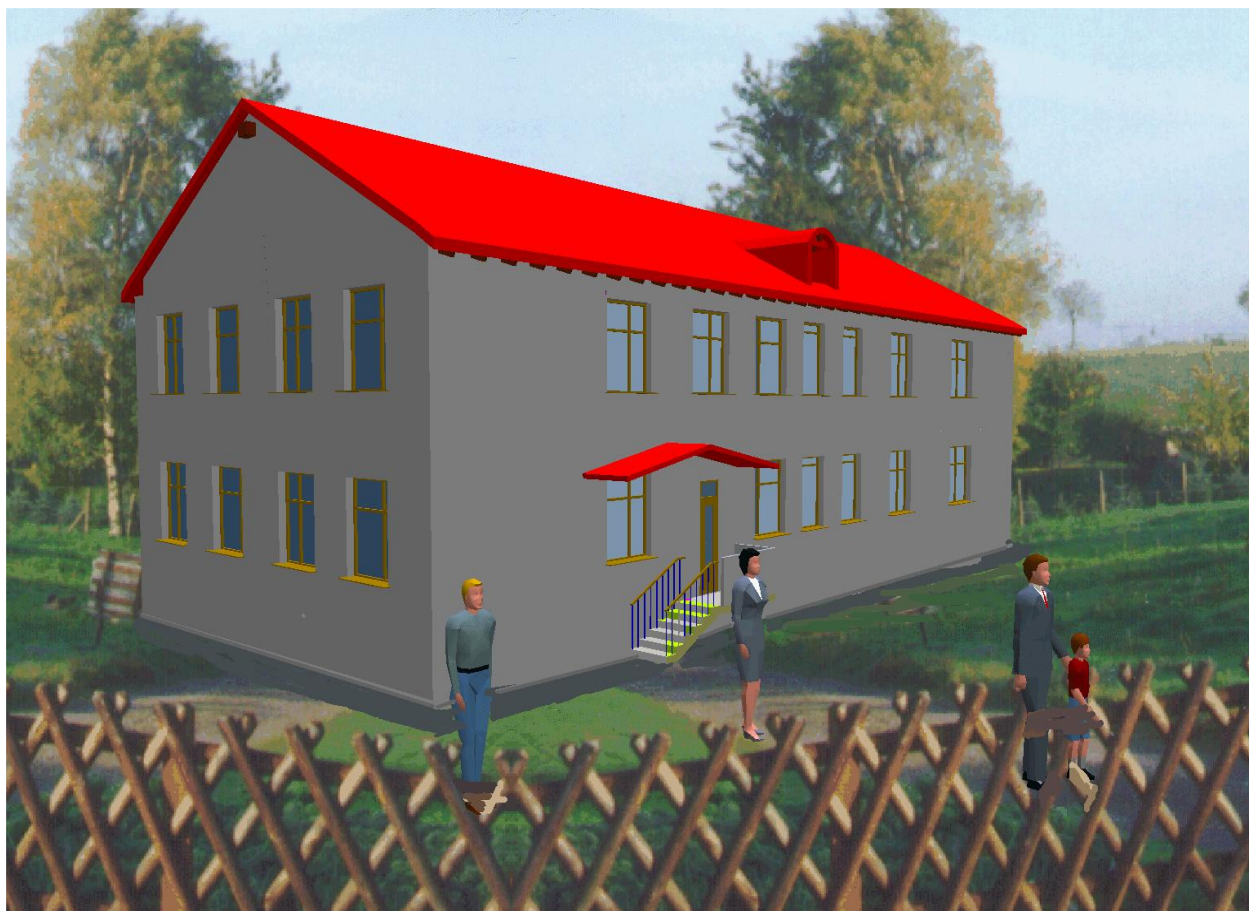
Архитектурно-планировочное направление САПР достаточно слабо было развито в СССР. Это объяснялось практическим отсутствием качественной материальной базы – графопостроителей, графических дисплеев. К моменту появления более-менее качественных и доступных графических аппаратных средств начали бурно развиваться персональные компьютеры. Поэтому в настоящее время в России используются локализованные версии западных производителей, такие как: ABIS (ABIS KROM, Австрия), speedikon-A (IEZ, Германия), ArchiCAD. Кроме того, наиболее активно используется, пожалуй, система общего назначения AutoCAD, дополненная различными прикладными инструментами для архитектора. Однако использование специализированных систем все-таки предпочтительнее, т.к. они базируются на понятии *компьютерная модель здания* (КМЗ). Здесь пользователь «составляет» модель здания на компьютере как бы из элементов строительного конструктора – стен, окон,

дверей, лестниц, крыши. Непосредственно чертежная документация автоматически составляется программой в заданном пользователе формате. Из одной и той же компьютерной модели можно автоматически построить различную чертежную документацию: - планы помещений, кладочные планы стен, аксонометрические и перспективные проекции и разрезы. Кроме того, программы могут автоматически составить спецификацию помещений, подсчитать требуемый объем материалов и строительных работ. Для примера приведена иллюстрация перспективных проекций многоэтажного жилого здания, автоматически построенная программой speedikon-A по введенным планам этажей.



Из российских разработок наибольший интерес представляет программа ArCon фирмы EuroSoft (Москва). Следует отметить, что в ArCon разработана качественная система фотореалистичной визуализации трехмерных изображений («лучевая трассировка»). Сегодня эта система визуализации интегрирована и в немецкую САПР speedikon-A. На следующее иллюстрации приведен автоматически построенный разрез 2-х этажного здания и фотореалистичное изображение перспективы этого же объекта. Цвета на разрезе определяют толщины и типы линий, которыми они будут выводиться на чертеже.





### 3.1.2 САПР расчетов на прочность

Здания и сооружения являются неотъемлемой частью современной действительности. Сегодня, как и тысячи лет назад, невозможно представить себе жизнь и деятельность человека в отрыве от возводимых им зданий. Влияние искусственных сооружений на окружающую среду многообразно. Современные здания и сооружения – сложные объекты, начиненные большим количеством механизмов и систем обеспечения жизнедеятельности и функционального назначения, оказывающих определенное влияние на окружающую среду. Их можно рассматривать с разных планов: архитектурно – художественного, функционального, экологического. Однако во всех случаях красной линией проходит вопрос эксплуатационной надежности, к

которому традиционно относятся требования обеспечения прочности, жесткости и устойчивости объекта.

Опыт массового строительства с начала 60-х годов прошлого века показал, что эксплуатационную надежность зданий и сооружений невозможно рассматривать в отрыве от окружения: грунтового массива, построенных ранее объектов, транспортных и хозяйственных коммуникаций. Ведомственная разобщенность эмпирических исследований не позволяет сегодня не только ответить на множество вопросов в их взаимной связи, но и провести качественное планирование различных текущих работ. Кроме того, экспериментальные исследования различных сложных систем (например, транспортный поток – грунт – фундамент – здание) позволяют построить интегральную информацию, из которой сложно выделить причины и следствия различных эффектов. Наиболее полную картину поведения здания при различных воздействиях можно построить на основе аппарата математического моделирования.

Механико – математическая модель, адекватно отражающая основные особенности сооружения в его взаимосвязи с грунтом основания и рядом находящимися зданиями, позволяет при малых затратах получить адекватную информацию по широкому спектру проблем. Однако построение качественной модели сегодня представляет трудности как теоретического плана (например, построение корректных уравнений состояния определенной среды или контакта разных сред), так и реализации (т.е., методов получения решения сформулированной задачи).

Сегодня наиболее распространенным математическим аппаратом для численного моделирования поведения здания в комплексной постановке «верхнее строение – фундамент - грунт» при различных видах воздействия является *метод конечных элементов (МКЭ)*.

МКЭ позволяет рассматривать области со сложной топологией при разнообразных граничных условиях. Основная идея метода конечных элементов, как и всех сеточных (например, метода конечных разностей) и большинства прямых методов (Рунге, Бундса - Галеркина и т.п.), состоит в переходе от бесконечной системы дифференциальных уравнений в частных производных с граничными условиями к конечной системе алгебраических уравнений относительно конечного числа неизвестных, так или иначе отнесенных к пространственной сетке либо к системе базисных функций. Можно сказать, что метод конечных элементов в определенном смысле является синтезом сеточных и прямых методов, т.к. при использовании способа формирования коэффициентов системы уравнений, адекватного прямым методам, методика построения базисных функций основана на сеточном подходе. Благодаря этому метод сохраняет основные преимущества прямых методов (как то: возможность применения как исходной системы дифференциальных уравнений, так и соответствующих вариационных принципов, возможность интегрирования по всей пространственной области) и, вместе с тем, использует гибкие возможности сеточных методов для учета реальной геометрии рассчитываемого объекта и его фрагментов и граничных условий. Узлы сетки принадлежат рассматриваемой пространственной области и ее границе. После замены исходной пространственной области некоторой сеткой с конкретными параметрами аппроксимации говорят о **дискретизации** расчетной области.

В большинстве случаев при построении системы разрешающих уравнений используются приближенные зависимости для аппроксимации основных функций между соседними узлами сетки. Чем гуще пространственная сетка, чем меньше расстояние между ее узлами, тем выше точность аппроксимации функций, тем точнее приближенное решение. Однако чем больше узлов сетки, тем выше порядок системы алгебраических



уравнений, тем выше трудоемкость и время расчета и тем больше требуется ресурсов памяти компьютера для хранения массивов данных. Поэтому при использовании метода конечных элементов необходим компромисс при назначении дискретизации пространственной области. Сетка должна быть достаточно густой для получения приемлемого по точности решения, и вместе с тем ее размерность не должна превысить некоторого предельного значения, после которого использование этой дискретизации становится невозможным из-за высокой размерности системы. Сгущение обычно проводится не для всей сетки в целом, а в отдельных фрагментах, в которых поведение решения существенно зависит от степени дискретизации. Обычно это области, в окрестности которых приложена нагрузка, имеются границы зон с различными физико-механическими характеристиками (например, граница бетон - грунт), имеются различные геометрические особенности (углы, полости, отверстия и т.п.).

Метод конечных элементов основное развитие получил с середины нынешнего века. Первоначально два различных подхода - инженерный и математический - позднее слились в один общий, что имело огромное значение для его дальнейшего быстрого развития и широкого применения.

В настоящее время при решении задач статики используются три основных подхода МКЭ:

- **Прямой** - используется при решении относительно простых задач, имеет четкий иллюстративный характер,
- **Вариационный подход** (или метод Ритца в форме МКЭ) - основан на принципе стационарности некоторого функционала. Обычно используются функционалы Лагранжа, Кастильяно, Хеллингера - Рейснера, Ху - Васидзу. При решении инженерных задач строительной механики наиболее часто используется вариационный

принцип Лагранжа с перемещениями узлов сетки в качестве основных неизвестных.

- **Подход резидуума** (или метод Бубнова-Галеркина в форме МКЭ) представляет собой общий вид аппроксимации, базирующийся на дифференциальных уравнениях рассматриваемой задачи. Применяется обычно при решении таких задач, в которых трудно сформулировать вариационный принцип или такового не существует.

При решении задач механики грунтов, оснований и фундаментов методом конечных элементов рассматривается область конечных размеров. Как правило, расчетная область грунта представляется в виде параллелепипеда, размеры которого в плане больше, чем фрагмент фундаментной части здания, а по глубине захватывают деформируемый слой грунта. Это необходимо для снижения погрешности в окрестности конструкции здания, которая является следствием краевых эффектов. Грани параллелепипеда (кроме уровня дневной поверхности) закрепляются от перемещений. Наиболее густая сетка элементов составляется для области в окрестности фундаментной конструкции. На определенном расстоянии от нее размеры конечных элементов можно увеличить для снижения трудоемкости. Выбор размеров расчетной области и элементов проводится путем выполнения поверочных расчетов. Вначале проводится серия расчетов по увеличению размеров области до стабилизации решения, затем для области с назначенными размерами проводится сгущение сетки конечных элементов. Объем поверочных расчетов (т.е. быстрота нахождения приемлемой конечно-элементной модели) зависит от опыта и интуиции расчетчика.



Учитывая, что сетка конечных элементов связана не только с основанием и фундаментами, но и с верхним строением, можно сказать, что размерность задачи здесь будет весьма высокой. Размерность задачи существенно увеличивается из-за того, что необходимо сгущать сетку в области трещин, усиления и т.п. Однако пространственная задача обуславливает резкое увеличение порядка системы разрешающих уравнений. Следствием увеличения порядка является невозможность проведения расчетов на приемлемой по густоте сетке конечных элементов на современных компьютерах. Многие авторы используют достаточно эмоциональный термин «*проклятие размерности*».

Традиционные формы МКЭ предполагают формирование глобальной матрицы жесткости системы и ее последующее решение. Порядок систем уравнений в реальных задачах измеряется сотнями тысяч и миллионами неизвестных. Снижение порядка глобальной матрицы с помощью так называемого «суперэлементного» подхода далеко не всегда позволяет обойти ограничения на размерность задачи. Дело в том, что подходу «суперэлементов» по отношению к традиционному МКЭ присущи два недостатка. Первый заключается в ухудшении обусловленности матрицы жесткости ансамбля суперэлементов, что приводит к снижению точности решения. Второй связан с резким увеличением ширины ленты матрицы жесткости ансамбля суперэлементов. При этом теряются преимущества снижения порядка системы уравнений, так как зачастую для хранения матрицы меньшего порядка с увеличенной шириной ленты требуется больше ресурсов оперативной и внешней памяти компьютера, чем для стандартной матрицы жесткости.

Решение реальных нестационарных задач строительной механики в пространственной постановке практически невозможно традиционными методами. При решении нестационарных задач используются методы

«продолжения по параметру времени», иначе – шаговые методы решения нестационарных задач. Временной сегмент разбивается на ряд отрезков. По специальным формулам, уникальным для каждого метода, осуществляют переход от параметров НДС в начальный момент к первой точке временного разбиения. Затем производят аналогичный переход ко второй точке, используя в качестве начальных значения в первой точке и т.д. Количество шагов по времени может быть достаточно большим, и в пределах каждого шага решается задача, по трудоемкости и внешнему виду соответствующая некой квазистатической задаче. В результате общая трудоемкость процесса резко увеличивается.

Качественный скачок вычислительной техники и программных средств по расчету строительных конструкций методом конечных элементов (МКЭ) за последнее десятилетие привел к определенной революции в методах расчета. Если ранее основным недостатком расчетов была низкая точность определения усилий простейшими методами, не учитывающими взаимную работу отдельных конструктивных элементов, то сегодня можно моделировать достаточно сложные объекты с учетом взаимосвязи их отдельных фрагментов.

Потребности проектных, научно – исследовательских и промышленных организаций, в универсальных, достаточно быстрых и удобных для пользователя программах послужила импульсом к разработке различными фирмами пакетов прикладных программ конечно – элементно анализа. К ним можно отнести известные продукты ANSYS и DesignSpace фирмы Ansys Corporation; Cosmos/DesignStar, Cosmos/Works фирмы Structural Research & Analysis Corporation; DesignWorks фирмы CADSI; АПИМ WinMachine Центра программного и научного обеспечения АПИМ; NASTRAN (NAsa STRuctural Analysis – анализ конструкций национального комитета

США по авиации и космическим исследованиям). Из программных средств, адаптированных к российским СНиП, известны комплексы StarKON фирмы EuroSOFT (Москва), Лира и Мономах производства НИИАСС (Киев). Из комплексов, в которых решается задача предельного равновесия в статической и динамической постановках с учетом реологии материала, определение «безопасного времени» функционирования конструкции или отдельного процесса воздействия на нее, известны комплексы «КЛЕН», «ПОЛЮС», «МИНОР» (РГСУ).

В комплексе StarKON впервые, пожалуй, решена задача комплексного проектирования по схеме «архитектурный проект» - «расчет» - «конструирование» - «выдача рабочих чертежей конструкций». В качестве архитектурного используется комплекс «speedikon» фирмы IEZ GmbH, работающий на платформах известных CAD – программ (AutoCAD, MicroStation). В качестве программы для расчета и определения/проверки сечений и армирования используется STARK ES. Комплекс может работать в совместно с программой ввода позиций ProFET. После расчета можно выполнять конструирование с автоматизированной подготовкой рабочих чертежей в системе AutoCAD.

Следует отметить, что широкое внедрение программных МКЭ-комплексов привело к упрощенному отношению к прочностным расчетам. Многие руководители придерживаются принципа: «Есть компьютер, программа – нажми кнопку и получи результат». Может быть, когда-нибудь вычислительная техника и программные средства и достигнут подобного уровня. Однако и сегодня использование современных программных и аппаратных средств не снижает требований к уровню подготовки специалиста.

Еще десятилетие назад расчет методом конечных элементов был доступен узкому кругу специалистов, достаточно досконально знавших и основы метода, и основные его «подводные камни» не только на уровне пользователя программных средств, но и на уровне программиста. Сегодня же интерфейс большинства МКЭ-комплексов удобен в такой степени, что эти программы могут использовать большинство проектировщиков, изучивших основы работы с Windows. При этом создается достаточно опасная иллюзорность того, что использовать МКЭ в расчетах может любой проектировщик, владеющий СНИП и компьютером. Но все дело в том, что знание СНИП (норм, определяющих внешние воздействия и методы назначения или проверки сечений по известным внутренним усилиям) недостаточно для качественного определения этих усилий даже с помощью имеющихся в распоряжении программных средств. На уровне обучения основы МКЭ только начинают давать в строительных ВУЗах (последние 10 - 15 лет), поэтому большинство проектировщиков просто не владеют основами метода.

Наиболее существенным “подводным камнем” метода является то, что МКЭ - приближенный метод. Приближенность метода определяется двумя факторами: качеством конечноэлементной модели (непосредственно погрешность метода и использованных типов КЭ – так называемое качество дискретизации модели), и качеством самой модели. Под этим понимается то, какими механико-математическими моделями моделировали те или иные конструктивные элементы и как учтено их сопряжение. Например, один и тот же элемент можно моделировать стержнем с определенными параметрами сечения (площадь, моменты инерции и т.п.), либо комбинацией пластин (здесь количество интегральных параметров меньше —только толщины элементов), либо трехмерной моделью. Плиту также можно моделировать либо оболочками различных постановок (тонкая плита, Рейснера, Власова),

либо объемной трехмерной моделью. Также существует множество различных моделей (постановок) для грунтового основания. Наиболее общими являются объемные постановки, т.к. в них не используются упрощающие гипотезы, априори определяющие характер деформирования системы. Все остальные упрощающие гипотезы заранее закладывают неустранимую погрешность расчета не только по числам, но и по качественному описанию процесса. Однако сегодня использование только объемных постановок практически невозможно. Поэтому пользователь, создавая исходную модель, должен достаточно ясно представлять погрешности, закладываемые им на стадии формулировки модели. Отметим, что в современных программных комплексах имеются средства построения модели на уровне конструктивных терминов, доступных конструктору и проектировщику, как то: *стены, колонны, балки* и т.д. Используя только эти средства при построении модели без анализа их сущности пользователь получает иллюзорную уверенность в том, что им построена достаточно корректная модель системы.

Не меньшее значение имеет и качество перехода при аппроксимации модели конечными элементами. Точность результатов существенно зависит от принятой конечно-элементной аппроксимации. Стандартными способами проверки качества является серия расчетов при разных степенях сгущения сетки в целом или в отдельных ее фрагментах. Достаточную опасность представляет тот факт, что сходимость результатов по напряжениям и усилиям более медленная, чем по перемещениям. Поэтому анализ качества конечно-элементной модели необходимо проводить не только по перемещениям, но и по усилиям. Некачественная сетка может давать недопустимую погрешность (в два и более раз), поэтому создание качественной сетки – достаточно трудоемкий процесс, требующий

определенной квалификации пользователя именно на уровне применения МКЭ.

Поэтому для уверенного использования метода необходимо иметь специальные знания и о его сути, и об основных постановках теории упругости (хотя бы теории упругости для линейных задач).

Достаточно ясно, что пользователь, научившийся вводить в компьютер текст, еще не является писателем или поэтом. Для этого необходимы специальные знания и талант. Уровень развития МКЭ и программных средств сегодня предъявляет аналогичные требования к расчетчику. Одного *механистического* владения программой не достаточно для решения реальной задачи, построение ее качественной модели – сложный творческий процесс.

### **3.1.2.1 Краткое описание программного комплекса по уточненному расчету зданий в процессе эксплуатации и выравнивания при использовании сетки конечных элементов высокой размерности**

Программный комплекс «КЛЕН базируется на *агрегатном* принципе (или принцип *декомпозиции*) построения САПР. Основанный на использовании абстракций принцип декомпозиции применяется для разбиения программ на компоненты, которые затем объединяются для решения основной задачи. При этом использован частный тип абстрагирования – абстракция через спецификацию, т.е. в возможности серией разных алгоритмов получить конечный искомый результат.

При трансформации спецификаций в структуру системы существуют два подхода. Основное разделение проведено в *ориентированном на обработку* подходе, когда проведено разделение комплекса на ряд агрегатов на основе модульного подхода. Вместе с тем как внутри отдельных модулей, так и для связи данных между модулями, использована *ориентированная на*

данные методология. А именно, использована концепция *объектно-ориентированного* программирования.

Обеспечение дружественного пользователю интерфейса достигнуто за счет использования возможностей визуального программирования и программирования, управляемого событиями. Ввод исходной информации осуществляется в интерактивном графическом режиме, когда пользователь «вычерчивает» схему конструкции с помощью системных указателей, а дополнительные текстовые и числовые данные вводятся в табличной форме. Процессы ввода и коррекции исходной информации не разделены между собой – в любой момент можно вводить новый фрагмент конструкции или исправлять ранее введенный блок.

Вывод результатов осуществляется в трех формах:

- графической (деформированные схемы, поля напряжений и перемещений в виде изолиний, графиков изменения параметров напряженно–деформированного состояния по временным и пространственным координатам),
- текстовой (табличная форма значений основных расчетных характеристик),
- интерфейс со стандартными программами Microsoft Office, т.е. переброска результатов расчета в программы Excel и Word, что позволяет пользователю использовать возможности этих программ для дополнительной обработки результатов.

В соответствии с предложенным алгоритмом комплекс разбит на ряд агрегатов, выполняющих полный расчет по отдельным логически завершенным этапам.

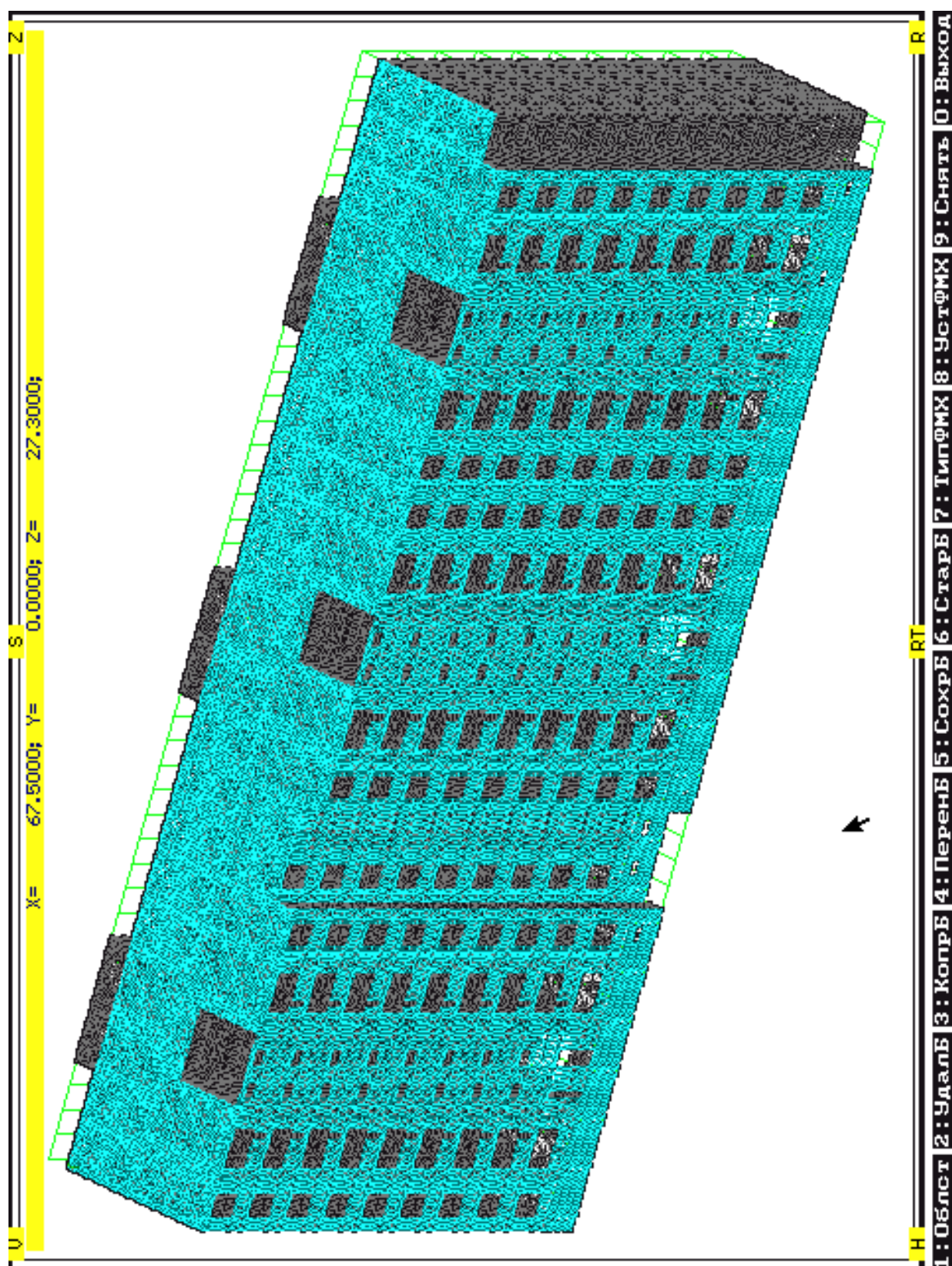
Основными являются следующие агрегаты:

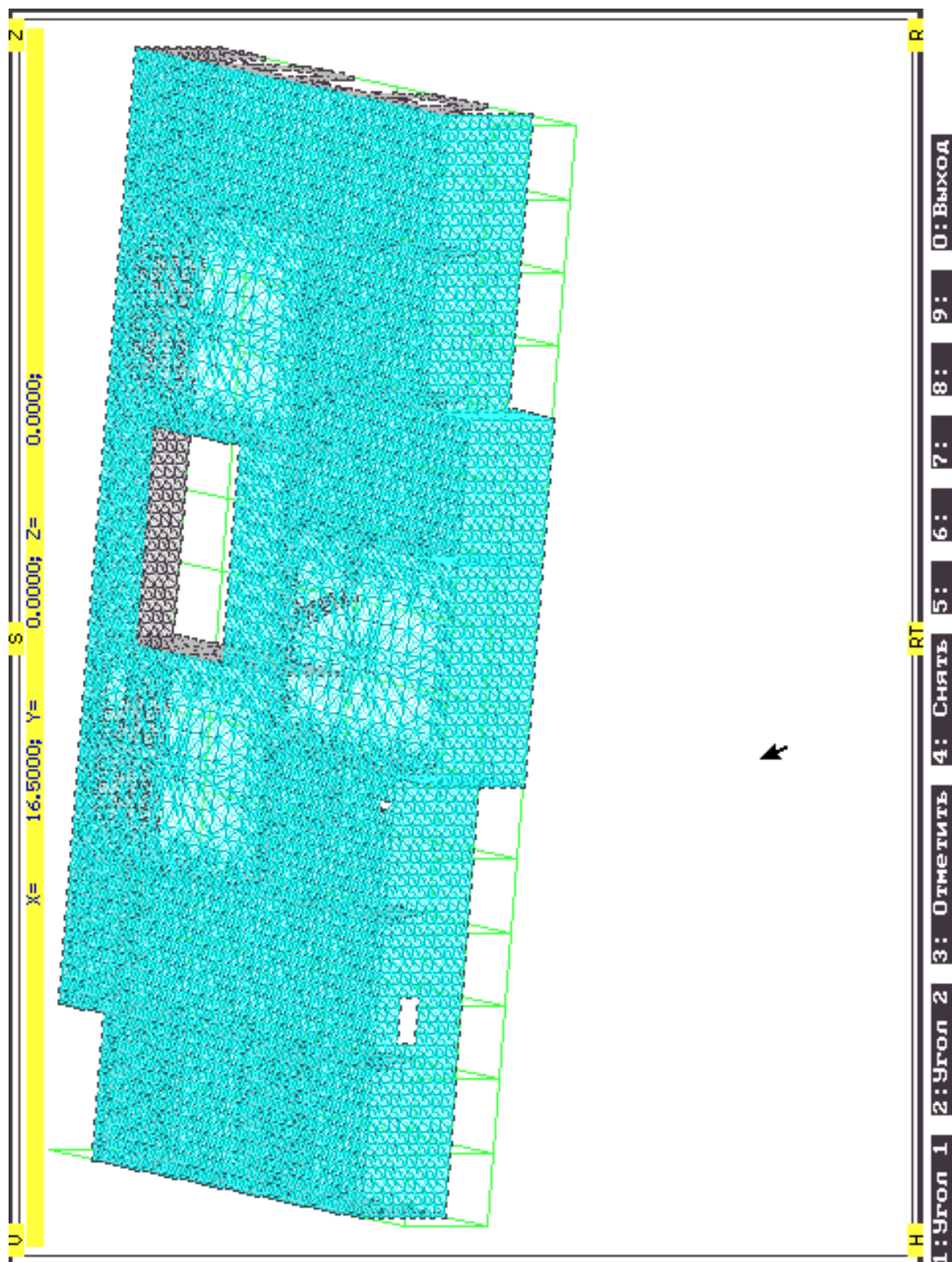
- ввод, коррекция и тестирование на наличие ошибок исходной информации,
- статический расчет конструкций при различных этапах выравнивания,
- блок определения наиболее неблагоприятных состояний в различных фрагментах конструкции,
- эволюционный расчет на длительное воздействие с учетом накопления невосполнимых деформаций,
- тестовый динамический расчет при образовании линии отрыва,
- блок анализа результатов, который разбит на фрагменты анализа результатов от статических воздействий и с учетом длительного воздействия.

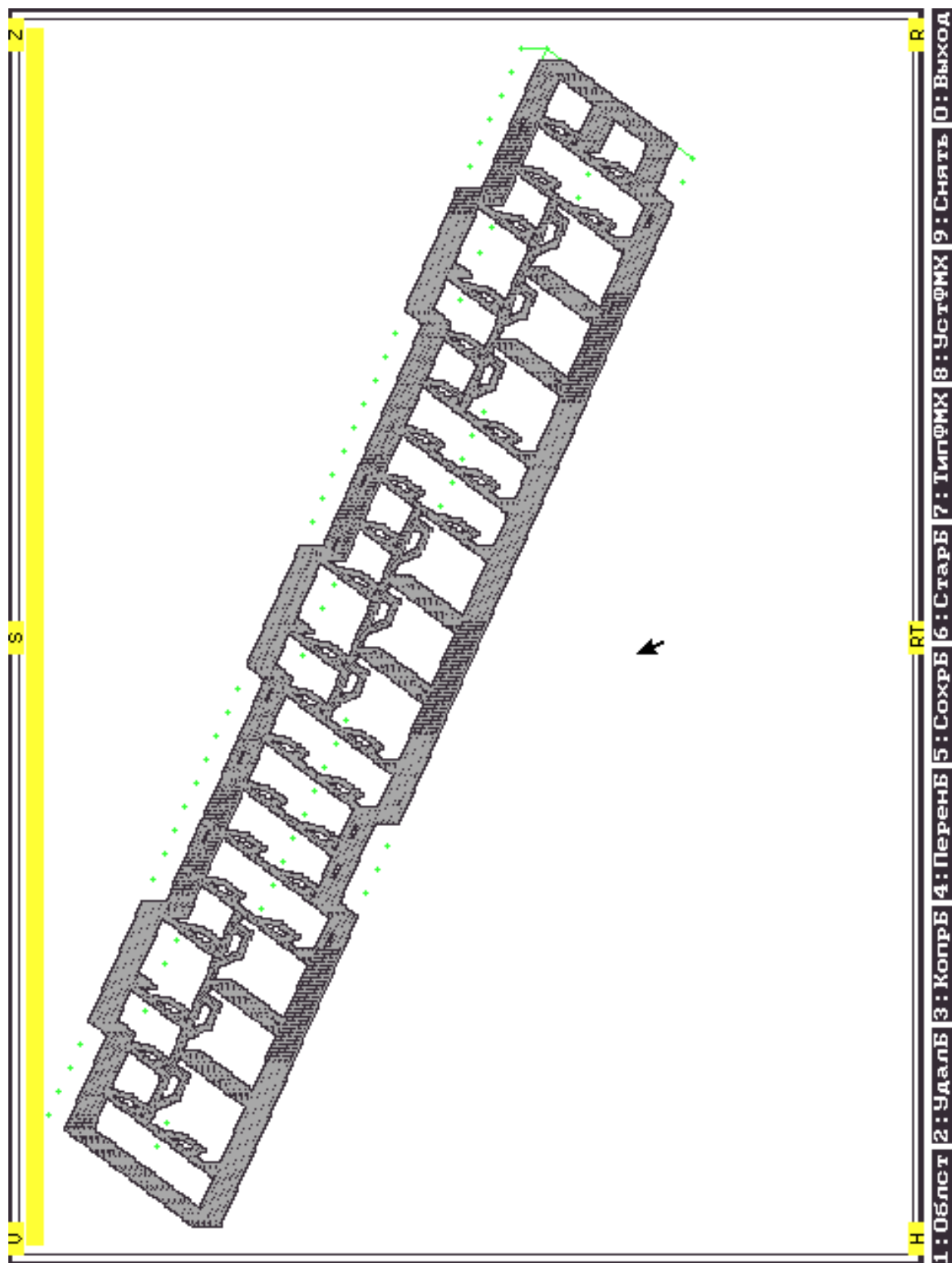
После получения основных результатов от расчета на базовые статические воздействия пользователь определяет группу их возможных сочетаний. Количество введенных групп определяет множество приведенных квазистатических нагрузок. В процессе проведения расчетов на длительное воздействие кроме групп нагрузок задается сочетание погодно – климатических факторов, т.е. закон изменения во времени зависящих от влажности параметров состояния. Для влажности предусмотрено задание ее изменения в течение года. Задание климатических факторов производится в табличных формах, поэтому можно задать произвольный характер их изменения, а не только часто используемые гармонические законы, которые не всегда адекватно отражают картину изменения температуры и влажности во времени. Это является дополнительным фактором, позволяющим приблизить модель к реальным условиям эксплуатации.

На следующих иллюстрациях показана МКЭ-модель 9-и этажного здания и ее фрагменты, подготовленные в комплексе КЛЕН.



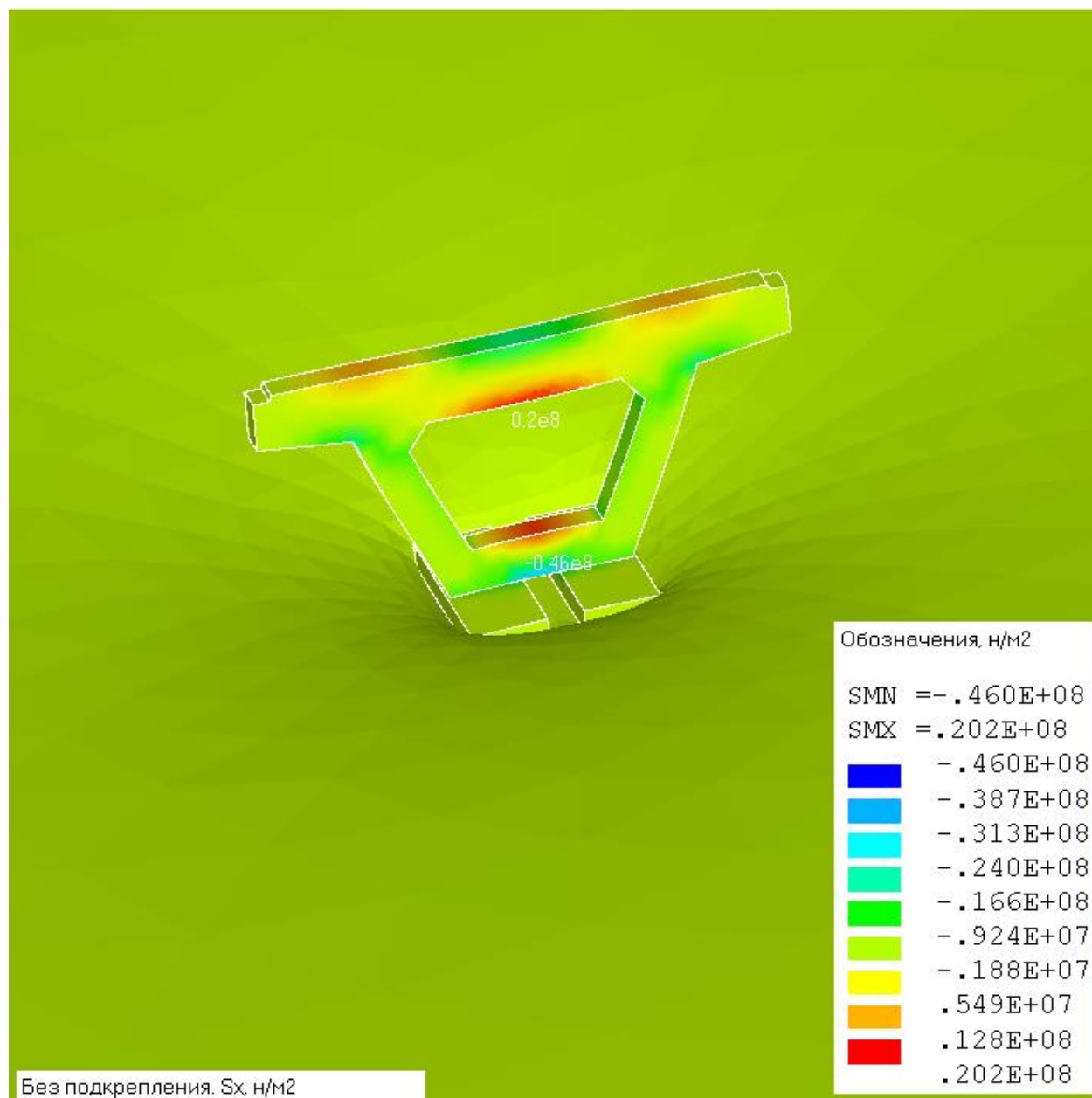




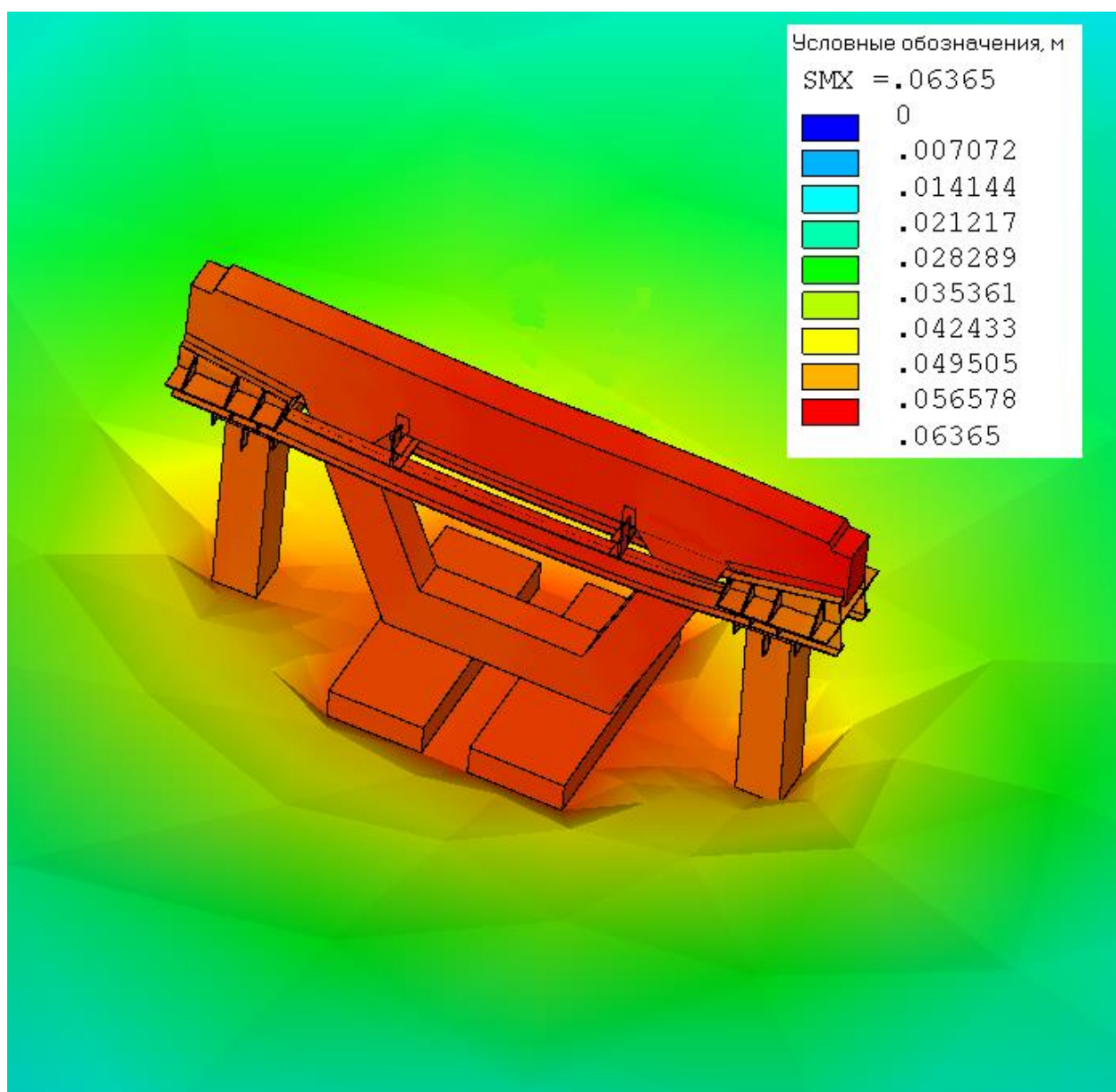




### 3.1.3 Примеры использования программных МКЭ-комплексов

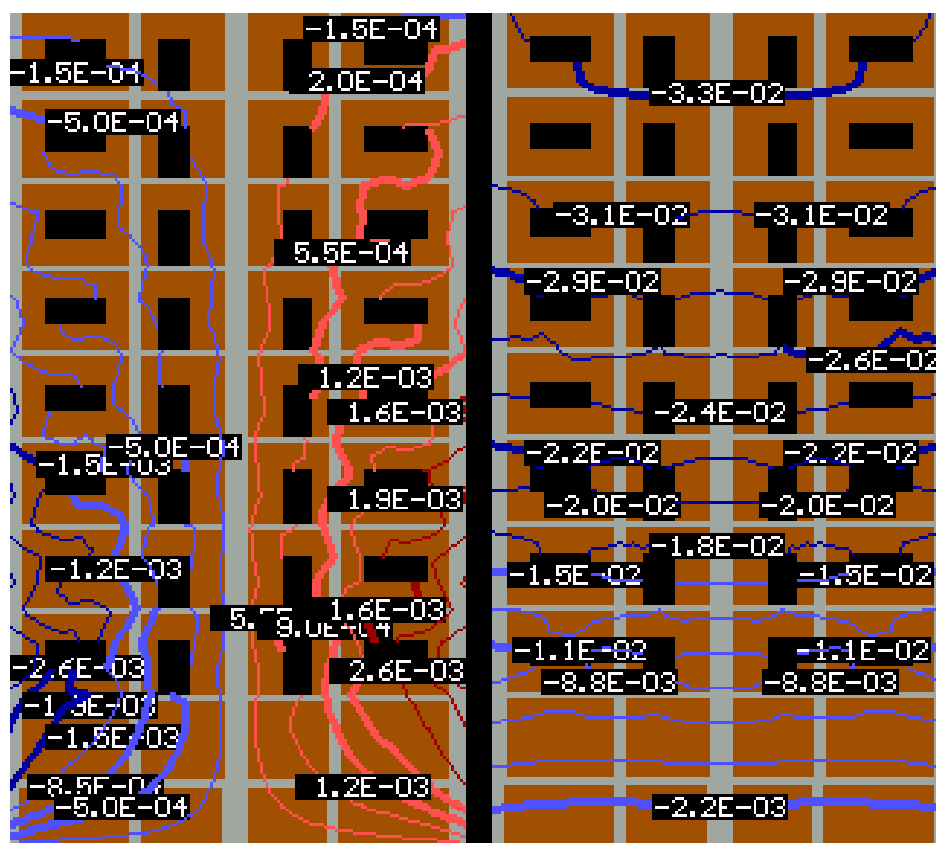


Пример расчета конструктивного элемента цоколя, опирающегося на свайный ростверк – комплекс ANSYS

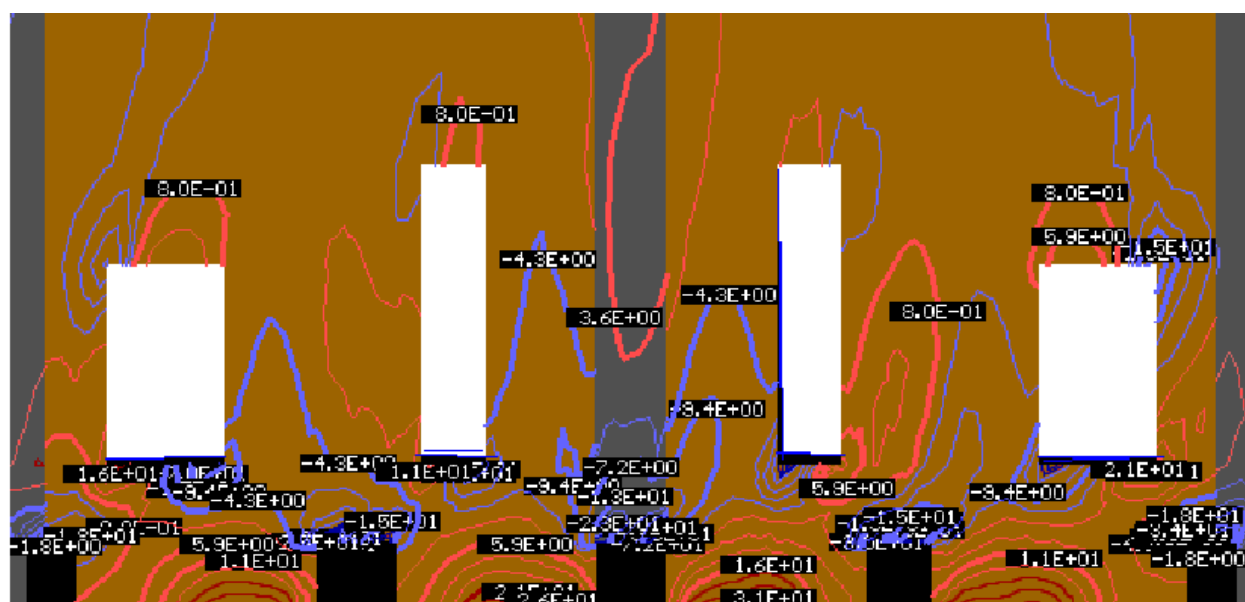


С подкреплением - суммарные перемещения, м

Пример расчета конструктивного элемента цоколя при разработке проекта усиления – комплекс ANSYS

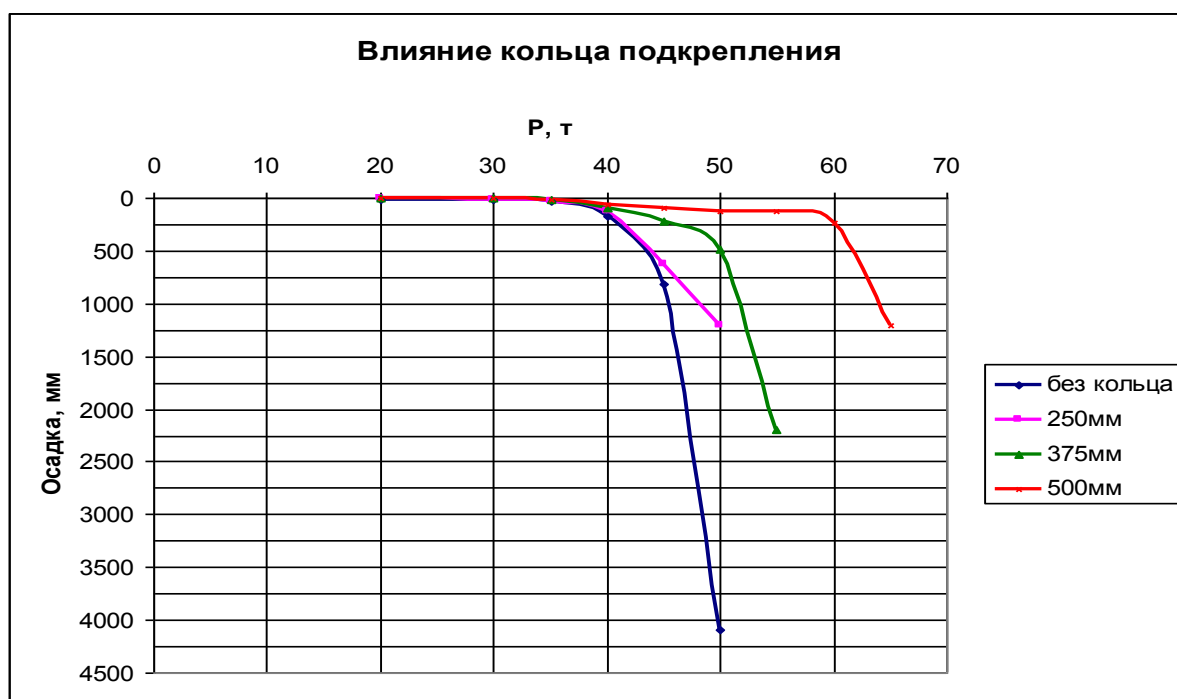
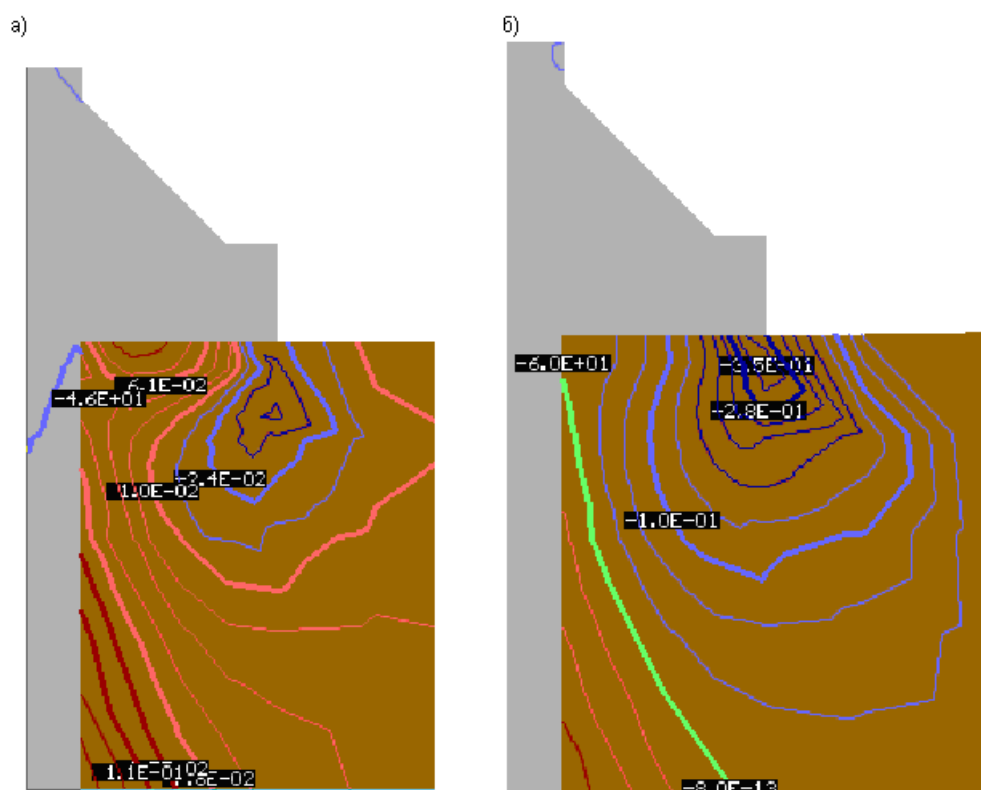


Горизонтальные                      Вертикальные  
Ось 1. Проектное состояние. Изолинии перемещений (см).

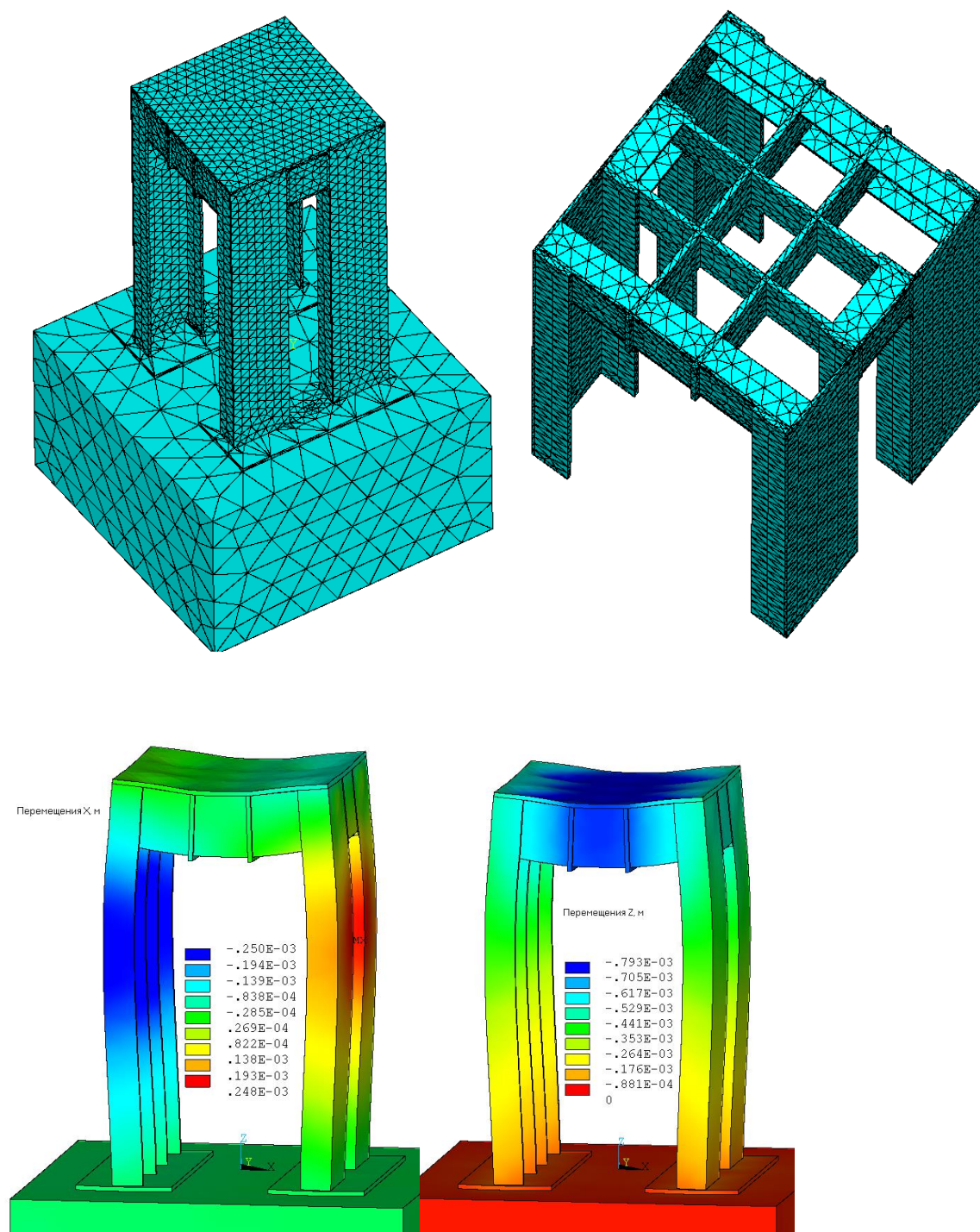


Опираение на домкраты. Горизонтальные напряжения (кг / см<sup>2</sup>). Фрагмент цоколя.

*Пример расчета несущих стен здания при подготовке проекта устранения ренов – комплекс ПОЛЮС*

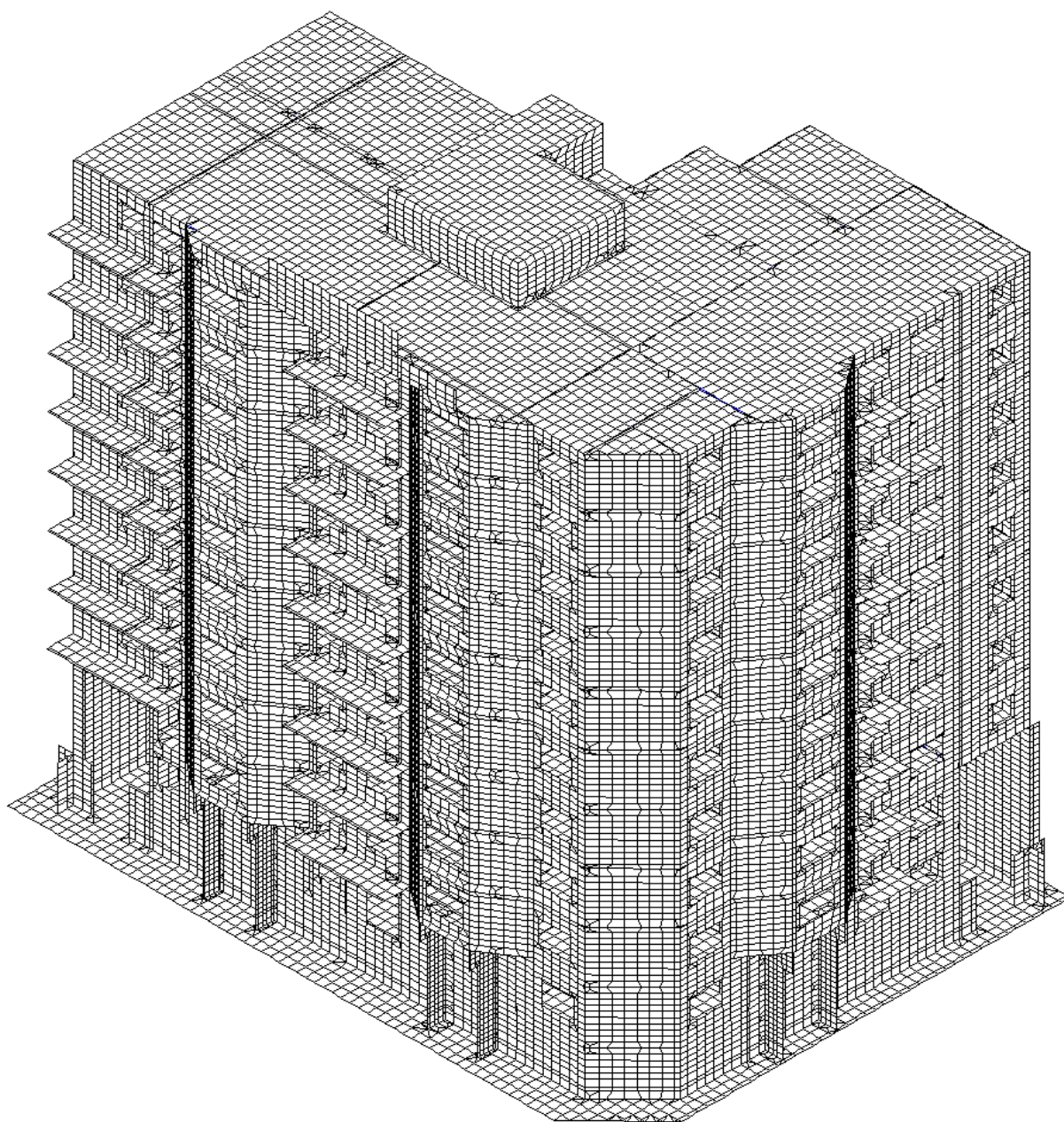


*Пример расчета ростверка сваи при подготовке проекта усиления фундаментов- комплекс ПОЛЮС*

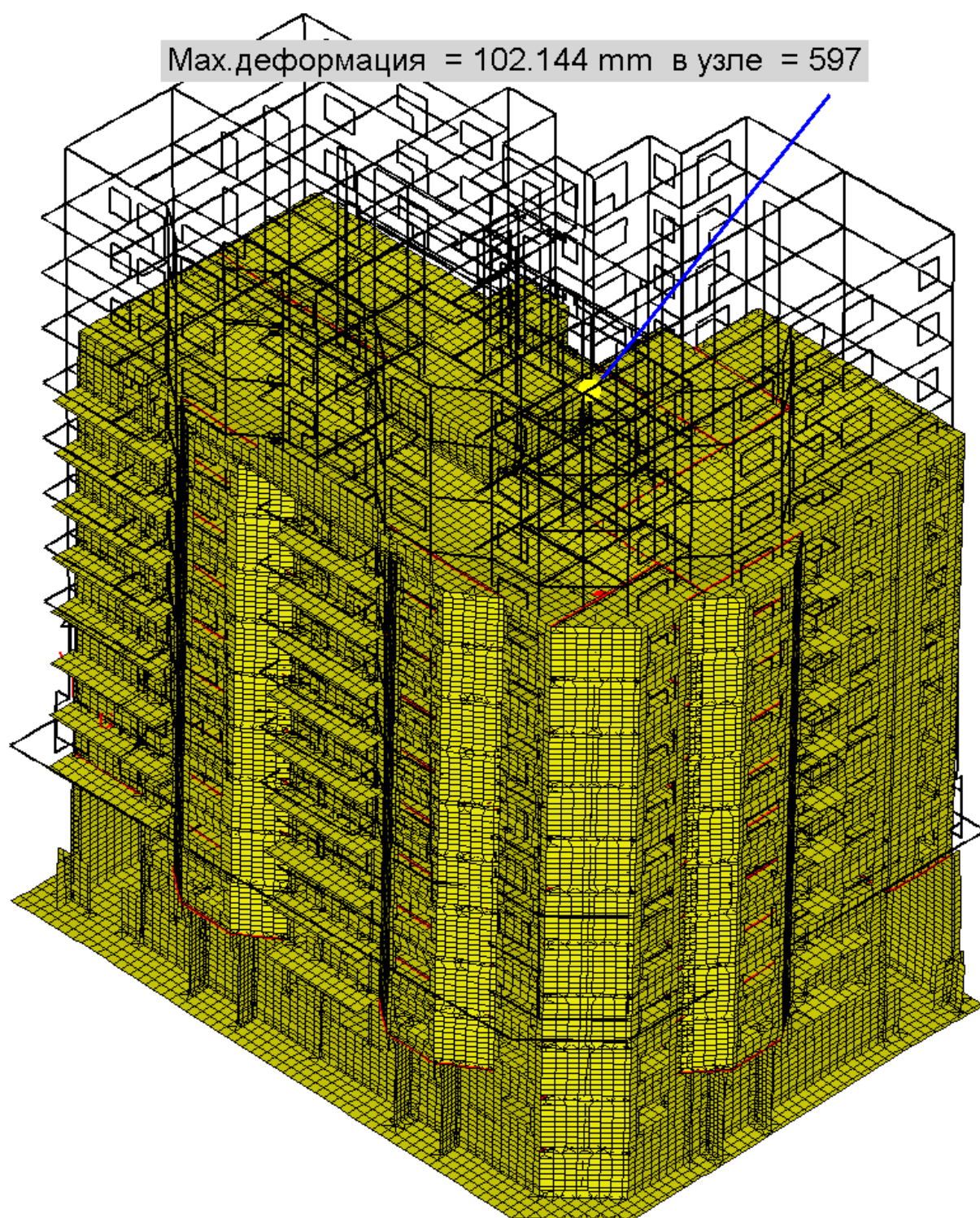


*Пример модели и некоторые результаты расчета вспомогательного домкратного столика*



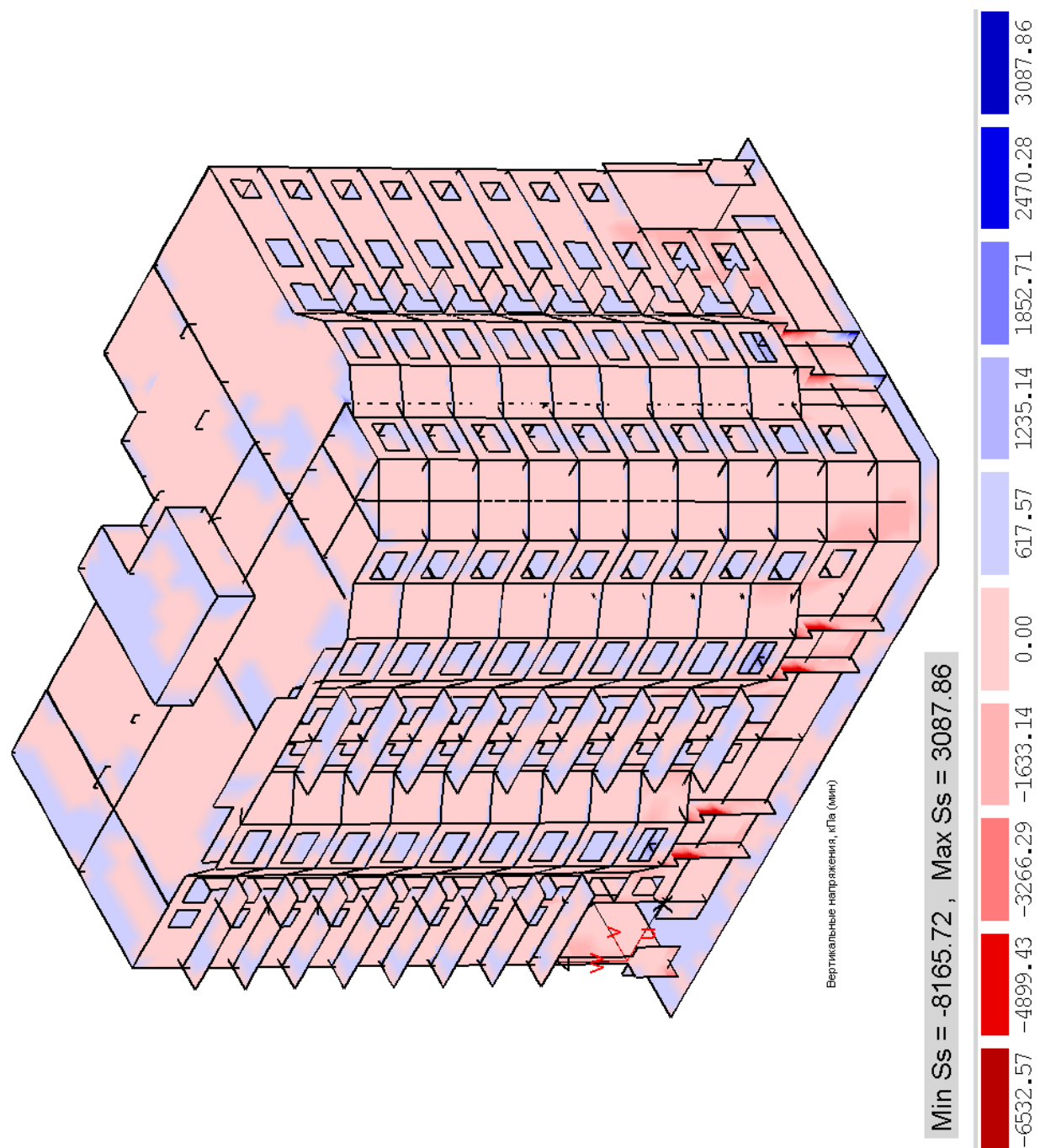


*Пример расчетной схемы жилого здания – комплекс ProFEt&Stark*

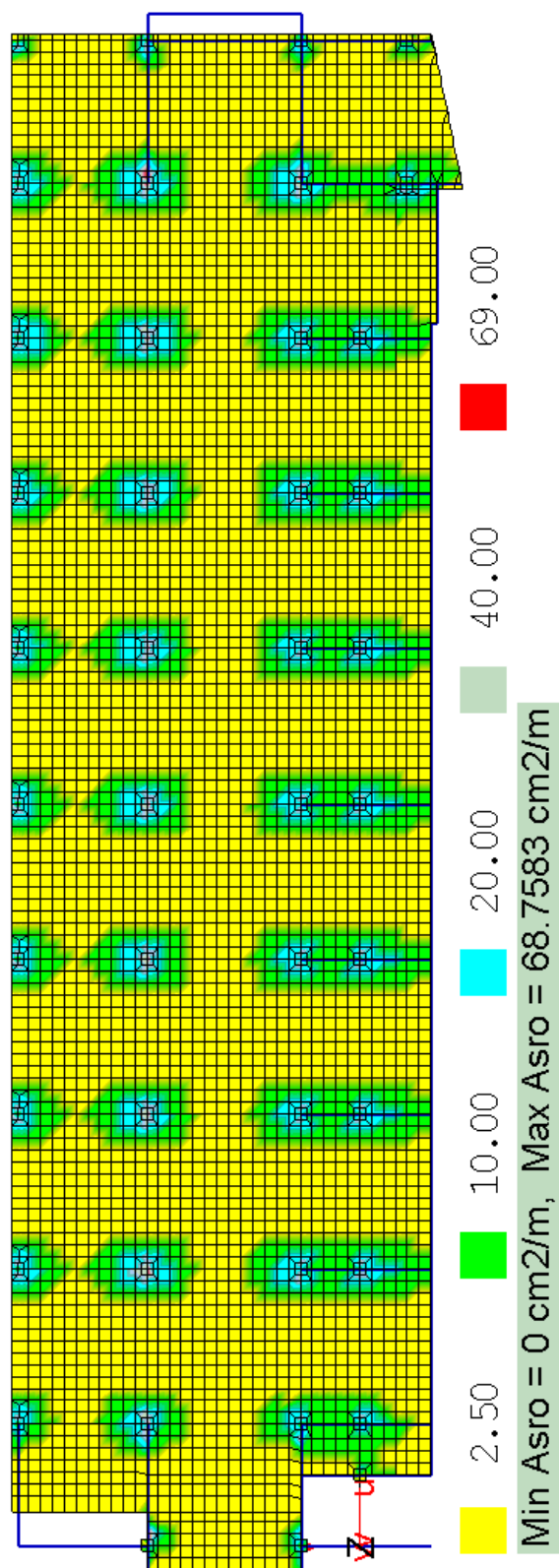


*Пример результатов расчета – деформации здания - комплекс ProFEt&Stark*





Пример результатов расчета – вертикальные напряжения - комплекс ProFEt&Stark



*Пример расчета горизонтальной (верхней) арматуры монолитной плиты перекрытия – вертикальные напряжения - комплекс ProFEt&Stark*

### 3.1.4 Контрольные вопросы

- Классификация систем САПР в строительстве,
- Системы САПР архитектурного назначения,
- Особенности специализированных архитектурных САПР,
- Возможности специализированных архитектурных САПР,
- САПР общего назначения,
- Отличие САПР общего назначения и специализированных САПР,
- Системы САПР расчетов на прочность, жесткость и устойчивость,
- Основные положения метода конечных элементов,
- Основные подходы МКЭ,
- Основные погрешности при использовании МКЭ,
- Основные программные МКЭ – комплексы,
- Какие аппаратные ресурсы влияют на точность получения результатов МКЭ?
- Возможности специализированных МКЭ-комплексов в строительстве.

## 3.2 Системы АСУ

Системы АСУ определяются как Автоматизированные системы управления. Область применения систем АСУ – это, фактически, все пространство деятельности человека. Иногда за внешней простотой использования АСУ просто не видно той громадной подготовительной работы, выполненной для успешного функционирования системы. Например, практически каждый сталкивался с автоматизацией процесса продажи железнодорожных или авиа билетов. Оператор вводит основные параметры вашего заказа, обсуждает с вами автоматически подобранные варианты, и

после выбора нужного система подготавливает необходимые Вам проездные документы. Однако задумайтесь об объеме труда, затраченного для функционирования этой системы. Введены огромные данные для основных баз – городов, рейсов, стоимости билетов в зависимости от класса поезда – вагона – расстояния; разработаны экспертные системы, помогающие оптимизировать маршрут следования (при наличии пересадок); разработано программное обеспечение для корректной обработки запросов и результатов операции – если Вам дадут билет в Ростове, то в Армавире уже не продадут билет на то же место; разработаны программно-аппаратные средства для быстрой связи компьютеров в гигантской сети по всей стране – и это еще далеко не полный перечень работ.

Системы АСУ могут контролировать отдельный технологический процесс – например, выплавку стали или транспортировку нефти по магистральному трубопроводу.

Здесь пойдет речь об АСУ управления недвижимостью.

### 3.2.1 Управление объектами недвижимости как управление средствами производства

#### 3.2.1.1 Общие положения

Эффективное управление объектами недвижимости позволяет сократить расходную часть и увеличить доходную часть.

Под управлением объектами недвижимости здесь следует понимать стратегическое управление проектом, направленное на содержание зданий, сооружений, инженерных систем в состоянии, обеспечивающем безотказную работу и согласование выполнения этих задач с трудовыми ресурсами с целью максимизации проектных выгод.

На западе к проблеме эффективного управления объектами недвижимости относятся достаточно серьезно. Отметим, что различные

ассоциации профессиональных управляющих создавались в различных странах с начала 20 века. Наиболее крупными объединениями сегодня являются:

*IFMA - International Facility Management Association, Houston / Texas, USA;*

*AFM – Association of Facility Managers, London / England;*

*FMN – Facility Management Nederland / Nederland;*

*JFMA – Japanese Facility Management Association / Japan;*

*NOPA – New Office Promoution Association, Tokyo / Japan;*

*EURO-FM – European Facility Management Network, Maasen / Nederland;*

*GEFMA – Deutscher Verband fuer Facility Management e.V., Bonn / Deutschland.*

### **3.2.1.2 Предпосылки развития автоматизированных систем проектирования и управления средствами производства**

Экономические факторы развития общественных отношений в Российской Федерации расширили смысловое содержание термина недвижимость.

Проблема повышения эффективности инвестиционных вложений в объекты недвижимости заставила выработать новый подход при создании проекта на реконструкцию или строительство здания. В настоящее время, создавая объект, мало рассматривать само здание, как вложенный и «замороженный» на многие годы капитал, этот капитал должен приносить доход с первого до последнего дня его существования. Не секрет, что при проведении экономической реформы в России многие учреждения и организации, имеющие в собственности административные здания, смогли выжить благодаря сдаче в аренду части своих площадей.

Однако использование современных систем управления объектами позволяет как существенно снизить затраты на эксплуатацию здания, так и рационально использовать имеющиеся площади, и в конечном итоге, существенно повысить доходность объектов недвижимости. Таким образом, происходит естественное превращение зданий и сооружений - объектов недвижимости - в объекты управления средствами производства.

Система автоматизированного управления средствами производства основаны на использовании компьютерных технологий, позволяющих прорабатывать различные варианты использования и обслуживания помещений (далее – по западной терминологии – CAFM – компьютерная система управления недвижимостью). Такое управление возможно в том случае, когда проект здания имеется в электронной форме, в виде компьютерной модели. Наиболее оптимальный вариант предусматривает использования средств автоматизации при проектировании объекта, однако создать электронную модель можно и для существующего здания или сооружения. Только наличие электронной модели здания позволяет производить постоянное обновление информации об объекте (актуализация данных).

В настоящее время использование ЭВМ для управления объектами чаще всего носит эпизодический характер – разработка автоматизированных систем управления отдельными технологическими процессами (АСУ ТП) или создание автоматизированных рабочих мест (АРМ) для обслуживающего персонала.

Требования, предъявляемые к качеству новых проектов строительства и реконструкции, а также к срокам их выполнения, оказываются все более жесткими по мере увеличения сложности проектируемых объектов и повышения важности выполняемых ими функций. Выполнить эти требования возможно только при использовании средств автоматизации



проектирования. Созданная, таким образом, электронная модель здания может быть изменена и усовершенствована в кратчайшие сроки. Однако переход от традиционных методов проектирования к автоматизированным позволяет не только решить проблему с минимизацией затрат трудовых ресурсов и времени проектирования, но и создает основу для управления построенными объектами.

### **3.2.1.3 Состояние проблемы управления объектами недвижимости**

В настоящее время отсутствует скоординированная система управления объектами недвижимости как средствами производства. Разрозненное ведение учёта отдельных участков обеспечения жизнедеятельности не позволяет сопоставлять данные и быстро реагировать на происходящие изменения. Целостное представление картины происходящего может быть только при наличии в каждый момент времени достоверной информации по интересующему вопросу.

Основным направлением развития САПР в строительстве являлась подготовка графической документации в проектных организациях. Программное обеспечение вместе с аппаратными средствами (графические дисплеи, устройства указания, плоттеры) позволяло автоматизировать наиболее трудоемкие работы чертежного характера. Базовые графические программы расширялись соответственно конкретным требованиям разработчиков. Например, использовались укрупненные блоки графических примитивов, позволявшие облегчить процесс черчения в соответствии с принятыми стандартами условных обозначений. Однако формируемые чертежи представляли собой только набор линий и символов, и любое изменение в одном из них требовало изменения и внесения поправок в другие, связанные с ним листы. Следующим витком развития САПР в строительстве явилось создание компьютерной модели объекта. Теперь

пользователь может создать не чертеж (набор графических примитивов), а электронную копию существующего или проектируемого объекта. В модель могут быть занесены данные о физической сущности элементов объекта. Так, вводятся данные о стенах, колоннах, окнах, проемах, лестницах, перекрытиях, коммуникациях и т.д. В современных программных средствах проектировщик задает соответствующие массивы информации в привычной и удобной для него графической форме. Однако коренным отличием от предыдущего подхода является то, что здесь графическая форма является только средством ввода и отображения (в соответствии с нормами условных обозначений) реальных элементов. После ввода электронной модели проектировщик может сформировать необходимые чертежные листы (планы, проекции, разрезы), провести вычисления (объемов материалов, работ и т.д.). Полная автоматизация этих операций позволяет экономить время, трудовые ресурсы, и значительно повысить качество проектирования за счет вариантных проработок.

Использование модели здания в математической форме открывает новые перспективы получения максимальной прибыли при эксплуатации здания или комплекса зданий.

Связь между проектированием и управлением объекта осуществляется посредством постоянной актуализации его характеристик (рис.). Так, если строительство объекта (здания, группы зданий или сооружения) ведется на основе разработанной электронной модели, то в любой момент времени в нее могут быть внесены необходимые изменения, по которым автоматически производится обновление проекта. Таким образом, уже к окончанию строительства актуальная электронная модель будет содержать информацию о фактическом состоянии объекта. При применении обычных способов проектирования, доработка проекта практически означает создание нового проекта. В процессе эксплуатации актуальные данные пользователи

получают из электронной модели и на их основании принимают решения по управлению (ремонт, перепланировка и т.д.). Затем проведенные фактические изменения заносятся в электронную модель (проводится ее актуализация).

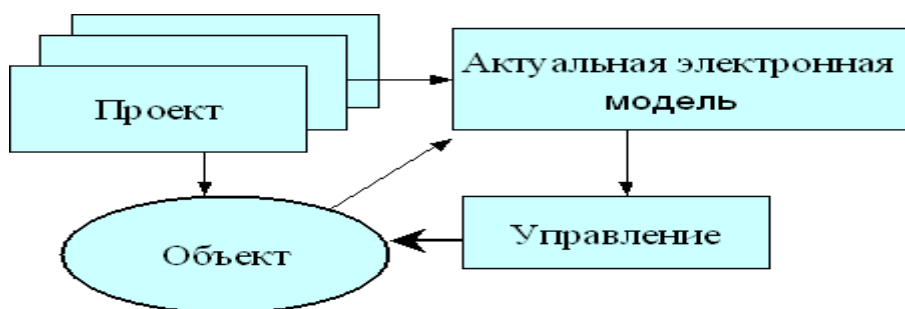


Рис. Схема актуализации информации об объекте

Следовательно, возможность актуализации информации в системе САФМ позволяет не только принимать оперативные решения по управлению, но и иметь проект, соответствующий фактическому состоянию объекта. Удобство актуализации информации обеспечивается за счет применения средств оперативной связи пользователя с компьютером, специальных проблемно-ориентированных языков и высокой информативной насыщенностью имеющихся баз данных.

#### 3.2.1.4 Цели и задачи программы «Управление объектами недвижимости как средствами производства»

Стратегические цели

Основными направлениями работ по осуществлению программы являются:

1. Разработка и внедрение программных средств по учёту и управлению средствами производства.

2. Организация сопровождения разработанных программных средств и организация интерфейса с другими программными продуктами, работающими в сопредельных областях.
3. Организация системы подготовки и повышения квалификации инженерно-технического персонала учреждений в области управления средствами производства.
4. Пропаганда достижений в области управления средствами производства, информационно-аналитическая деятельность.

Предметом автоматизации управления являются:

1. формализация проектных процедур;
2. структурирование и типизация процессов;
3. построение электронной модели здания или сооружения;
4. вариантное проектирование и выбор оптимальных методов и алгоритмов управления;
5. создание банков данных;
6. синтез составных частей в единую систему.

Рациональное распределение функций между управляющим персоналом и компьютерной техникой подразумевает, что человек выполняет творческую работу – предлагает новые варианты и анализирует полученные результаты, принимает окончательные решения, а компьютер с большой скоростью выполняет требуемые расчеты и первичный анализ проектных решений. Существенное преимущество управления объектом с компьютерной поддержкой состоит в возможности проводить многовариантные экспериментальные исследования на математических моделях существующих объектов со значительной экономией времени и трудовых ресурсов. Математические модели при этом должны удовлетворять требованиям универсальности, адекватности, точности и экономичности.

Для создания и внедрения автоматизированной системы необходимы следующие условия:

1. автоматизация сбора, обработки и выдачи необходимой информации;
2. совершенствование процесса управления с целью получения прибыли;
3. использование методов оптимизации и вариантного проектирования на основе современных электронных моделей зданий и сооружений;
4. создание банка данных, содержащего систематизированные сведения нормативно-справочного характера, необходимые для автоматизированного управления объектом и проектирования необходимых изменений;
5. создание и расширение банка данных комплектующих изделий и материалов;
6. подготовка специалистов в области компьютерного управления объектом;
7. увеличение творческой доли труда управленческого персонала;
8. унификация и стандартизация методов проектирования и управления объектами строительства;
9. взаимодействие подразделений управленческого персонала, проектировщиков – специалистов в области САПР и эксплуатационных служб.

Все объекты системы могут быть разделены на две категории: статические и динамические. К статическим объектам могут быть отнесены строительные конструкции здания или сооружения, наружное оборудование, внутреннее технические системы и оборудование (электрические сети, коммуникации связи, водопровод и водоотведение, отопление и вентиляция,

системы удаления мусора и т.д). Такие объекты являются неотъемлемой частью здания. К динамическим объектам можно отнести те, наличие которых не входит в понятие недвижимость: мебель, инвентарь, компьютеры, телефонные аппараты. Управление и обслуживание как статическими, так и динамическими элементами осуществляется зачастую разными организациями, но оплата расходов производится одним лицом – владельцем. Именно по этой причине необходимая оптимизация расходов тоже должна осуществляться одним лицом.

### **3.2.1.5 Концепция автоматизированной системы управления недвижимостью**

Особый подход должен быть определен к выбору данных для системы. Количество информации должно соответствовать текстовым возможностям системы. Необходимо вводить минимальное количество информации, а затем проводить оптимизацию ее использование.

Основная функция системы заключается в выполнении автоматизированного управления всего или отдельных частей здания или сооружения. При управлении объектом с помощью системы следует руководствоваться принципами системного единства, оптимизации связей с внешней средой, совместимости процессов и технологий, типизации, обновления и развития.

Принцип системного единства обеспечивает целостный, иерархический подход к решению проблемы эксплуатации и управления зданием, как частью, принадлежащей соответствующему комплексу зданий, земельному участку, улице, району и городу в целом.

Принцип оптимизации связей с внешней средой обеспечивается, если ввод справочно-нормативной и другой специфической информации делается один раз и может быть использован при проектировании очередного объекта.

Все данные, используемые многократно при управлении на разных иерархических уровнях системы, должны храниться в банке данных.

Принцип совместимости обеспечивает одновременное совместное функционирование отдельных модулей программы системы, таких как техническое обслуживание здания, управление инженерными сетями и оборудованием, управление площадями, управление переездами, , а также рассматривает всю систему в целом.

Принцип типизации ориентирует на преимущественное использование типовых и унифицированных элементов рассматриваемой системы. Типизации подлежат все элементы, имеющие перспективу неоднократного применения, причем каждый из них может рассматриваться как отдельный объект. Типовые элементы могут быть модифицированы по мере необходимости. Создание системы с учетом принципа типизации должно предусматривать разработку проектной документации по объекту с помощью САПР, а также создания модификации комплекса средств автоматизированного проектирования и его компонентов на основе базового варианта.

Принцип обновления и развития обеспечивает пополнение, совершенствование и обновление составных частей-объектов системы, а также взаимодействие и расширение взаимосвязи с автоматизированными системами различного уровня и функционального назначения.

Таким образом, архитектура программного обеспечения системы должна представлять собой синтез реляционных баз данных, обеспечивающих структурирование информации по различным признакам, и объектно-ориентированного программирования, поддерживающего принципы наследования, инкапсуляции и полиморфизма объектов. Такой подход позволяет выделить следующие основные особенности построения системы.

автоматизированная система «Управление объектами недвижимости» – иерархическая система. Она реализует комплексный подход к автоматизации всех уровней управления объектом. Иерархия уровней управления отражается в структуре специального программного обеспечения в виде иерархии подсистем и иерархии технических средств. Комплексный подход подразумевает обязательность автоматизации всех стадий управления, так как использование старых методов хотя бы на одной из них существенно, снижает эффективность всей системы.

автоматизированная система «Управление объектами недвижимости» – совокупность информационно-согласованных подсистем. Информационная согласованность системы обеспечивается единообразием объектов и методов доступа к ним в различных программах. На примитивном уровне обеспечивается единый алгоритм при чтении/записи блоков информации в подсистемах. Так, различные программы могут использовать в качестве исходной информации единые наборы данных; результаты решения одной задачи будут исходными данными для другой и т.д. Усложнение информационной согласованности может основываться на технологиях связи объектов Windows (например, COM, OLE или ActiveX-технологии). При этом различные программы могут выполняться одновременно в интерактивных режимах, и одна программа может генерировать событие, являющееся управляющим сигналом для другой программы. Слабая информационная согласованность превращает систему в совокупность автономных программ, в которых отсутствие учета отдельной информации, учитываемой в других подсистемах существенно снижает качество системы управления в целом.

автоматизированная системы «Управление объектами недвижимости» – открытая и развивающаяся система. Такая система может состоять из любого числа модулей, каждый из которых может быть введен в работу



самостоятельно, по мере его готовности. Введенный в эксплуатацию базовый вариант в дальнейшем расширяется. Более современные электронные модели объектов и более современная вычислительная техника могут заменять старые, менее удачные аналоги. Открытость системы заключена в возможности включения новых методов и средств в существующие программы.

Преимущества системы управления объектом могут быть сформулированы следующим образом:

- возможность быстрого изменения графических данных, входных и расчетных характеристик.
- минимизация издержек при эксплуатации объектов,
- возможность вариантного планирования,
- высокий уровень и комфортабельность самой системы управления,
- поддержание состояния недвижимости на высоком техническом уровне,
- получение прибыли при эксплуатации здания или сооружения.

#### **3.2.1.6 Результаты внедрения программы**

##### **1. Снижение постоянных издержек субъектов**

- Сокращение расходов предприятия по техническому обслуживанию и ремонту
- Сокращение выплат сторонним организациям, рабочим и служащим
- Сокращение издержек на восстановление и уборку соответствующим планированием
- Возможность оперативного анализа затрат по различным категориям и критериям

2. Повышение потребительской стоимости и сохранение материальных ценностей

- Сохранение материальных ценностей специальным уходом
- Сохранение материальных ценностей надлежащим техобслуживанием
- Повышение потребительской стоимости материальных ценностей (площади, оборудования, инвентаря) посредством специализированных услуг

3. Организация и администрирование в одних руках

- Улучшенная координация вследствие уменьшения количества проблемных участков
- Совмещение коммерческих и технических задач
- Оптимизация взаимосвязей
- Сосредоточение услуг в одних руках, вследствие этого сокращение обязанностей сотрудников

4. Улучшение инфраструктуры и повышение надежности для пользователя

- Оказанием всех услуг занимаются квалифицированные специалисты
- Повышение степени завершённости
- Гарантия оказания услуг

### **3.3 - Повышение надежности зданий Информационные системы управления городским хозяйством.**

#### **3.3.1 Введение**

От системного управления отдельными зданиями и сооружениями естественен логический переход к информационным системам управления городским хозяйством в целом.

В данной концепции не рассматриваются технические и программные аспекты построения общегородской информационной сети, связывающей локальные сети в организациях и их территориальные подразделения. Речь идет лишь о системном подходе

к управлению системой городского хозяйства как единым объектом, без учета той межведомственной разобщенности, существующей сегодня.

Более того, использование единого программного продукта с общей базой данных неминуемо приведет к развитию интеграционных процессов между различными ведомствами.

### 3.3.2 Логика взаимодействия данных

Управление системой проводится на основе структурных схем, которые являются составной частью системы. Каждый пользователь в любой момент времени может выбрать (или организовать) наиболее подходящее для себя представление схемы данных на основе задаваемой им структурной схемы. Например, одному пользователю в данный момент удобно организовать иерархию данных по схеме «земельный участок – здание – этаж – помещение». В тот же самый момент другой может рассматривать структуру «организация – подразделение – здания – этаж - помещение». Третий пользователь в этот момент работает на основе структурной схемы «организация – подразделение - инвентарь». Именно структурная схема определяет иерархию данных для конкретного пользователя. Определение структурной схемы проводится либо из множества ранее подготовленных шаблонов, либо определяется на основе имеющегося списка основных типов объектов. Использование структурной схемы позволяет пользователю максимально сосредоточиться на сути решаемой в данный момент задачи, оптимально настраивая структуру и представление данных. Использование структурных схем не влияет на фактическое размещение и схему данных. Фактически, речь идет о выделении для пользователя виртуальной среды данных, обладающей уникальной в данный момент структурой.

Большинство данных представляются как в графическом, так и в текстовом виде (таблицы, формы, отчеты и т.п.). Любому объекту на графическом экране (плану города, района, зданию, комнате, помещению, лестнице, элементу инвентаря и т.д.) соответствует та или иная табличная информация, представляемая либо в виде таблицы, либо в виде формы или отчета.

Состав табличной информации включает в себя ту информацию, которая представима в графической форме (например, планы этажей, размещение на этажах оборудования, мебели, коммуникаций и т.д.) и ту, которая не имеет графического представления (например, сведения о ремонтных организациях – виды ремонта, контактные лица, историю партнерских отношений). Т.е., основные данные хранятся в табличном виде, но большая часть данных выводится в текстовой или связанной графической формах. Изменение (редактирование) данных также может проводиться либо графически, либо в текстовой форме. Так, изменение стены (удлинение, укорочение, разрыв, устройство проема) на графическом экране приведет к изменению табличных данных, и наоборот. При графическом переносе мебели или инвентаря из одной комнаты в другую также изменятся все связанные табличные данные. Все действия запоминаются, и можно получить историю для любого объекта

Графическое редактирование невозможно только при изменении чисто текстовой информации (например, название подразделения, периодичность ремонта и т.п.).

Кроме этих основных данных, имеются вспомогательные данные, связанные с конкретными объектами на основе OLE или COM технологии Windows. Так, с любым объектом можно связать серию произвольных файлов (документы Word, таблицы Excel, рисунки, фотографии, видео или звуковые файлы и т.д.). Например, накапливать копии переписки по

аварийному зданию в форме отсканированных документов (рисунков), электронную почту и документы различных текстовых процессоров, чертежи в виде файлов AUTOCAD и т.д.

За счет объектно – ориентированного подхода пользователь легко изменяет (добавляет или удаляет) категорию информации или определяет состав информации по категории для любого объекта. Кроме того, пользователь может строить новые объекты данных либо на основе существующих (родительских) объектов, либо полностью новые группы данных.

Подобный пользовательский интерфейс позволяет проводить настройку системы на любом рабочем месте под конкретные задачи каждого пользователя, не выполняя изменение программного кода продукта.

### 3.3.3 Состав системы управления городским хозяйством.

Основные подсистемы:

#### 1. Геоинформационная система.

- 1.1 Система управления участками и улицами.
- 1.2 Система управления рельефом.
- 1.3 Система управления транспортом.
- 1.4 Система управления населением.
- 1.5 Система управления зданиями.
- 1.6 Коммуникации (водопровод).
- 1.7 Коммуникации (канализация).
- 1.8 Коммуникации (теплоснабжение).
- 1.9 Коммуникации (газоснабжение).
- 1.10 Коммуникации (электроснабжение: освещение, наземный электротранспорт).
- 1.11 Коммуникации (связь).

#### 3.3.3.1 . Связанные с геоинформационной системой базы данных по основным городским объектам.

Вспомогательные базы:

2.1. База данных рельефа – хранит информацию о высотных отметках территории. Позволяет автоматически строить вертикали, профили и трехмерные виды. Корректировка возможна изменением базовых отметок, изменением плана вертикалей, заданием разрезов. Информация хранится в оптимизированной форме.

2.2. База данных городских участков и улиц. Хранит стандартную информацию для городского кадастра. Является вспомогательной базой для определения адресов, собственников участков и зданий, оптимизации транспортных потоков. С этой базой связана база данных по геологическим изысканиям на участках.

2.3. База данных о населении. Можно использовать или конвертировать имеющиеся базы.

Одна из важнейших баз -

2.4. База данных по зданиям и сооружениям.

Содержит информацию по каждому зданию, которая делится на группы:

2.4.1. Общие архитектурно – планировочные данные, включая планы этажей, помещений в электронной форме.

2.4.2. Данные об инженерных решениях различных конструктивных блоков и элементов – фундаментов, стен, перекрытий и покрытия, кровле, окнах и т.п., включая информацию о текущем состоянии этих элементов.

2.4.3. Данные, связанные с возможным аварийным состоянием здания – периодические геодезические наблюдения, данные о геологии, акты обследования, проектные решения.

2.4.4. Данные о внутренних сетях и коммуникациях – планы и схемы внутренних сетей, данные о подключении к внешним сетям, данные о

вспомогательном инженерном оборудовании (лифт, противопожарная система и т.п.)

2.4.5. Система планирования и учета ремонтных мероприятий. Включает предварительную оценку стоимости ремонтно – восстановительных работ, подсистему слежения за плановыми сроками ремонта, подсистему учета внеплановых аварийных ситуаций (например, прорыв трубопровода, срыв кровли) и систему учета фактических трат на ремонтно – восстановительные работы.

В этом пункте не расписан состав подсистем по отдельным видам планирования и учета ремонтных работ, т.к. состав информации примерно аналогичен для различных систем (водоснабжение, водоотведение и канализация, теплоснабжение, электроснабжение, телефонная связь, кабельное телевидение и радио, центральное кондиционирование и т.д.). Однако для каждой системы в программе предусмотрен отдельный блок специфической информации.

2.4.6. Система учета собственников и арендаторов площадей в зданиях. Позволяет автоматизировать учет арендных и налоговых платежей.

2.4.7. Подсистемы планирования переездов и инвентаризации. Позволяют автоматизировать процесс учета инвентаря и оборудования, а также выбрать наиболее рациональное размещения оборудования и определения функционального назначения помещений..

### **3.3.3.2 Связанные с геоинформационной системой базы данных о городских сетях**

#### **3.1 База данных холодного водоснабжения (городской).**

3.1.1 Участки водопровода (координаты, характеристики труб, техническое состояние участка, данные о вводе в эксплуатацию и ремонтах).

3.1.2 Планирование мероприятий по техническому обслуживанию участков холодного водоснабжения (осмотр, ремонт).

3.1.3 Данные по колодцам (обследование на загазованность, запорно-регулирующая арматура, водомеры).

3.1.4 Планирование мероприятий по техническому обслуживанию колодцев (осмотр, ремонт).

3.1.5 Данные по насосным станциям (количество, марка, характеристики насоса и электродвигателя, вспомогательное и дополнительное насосное оборудование).

3.1.6 Планирование мероприятий по техническому обслуживанию насосных станций (осмотр, ремонт).

3.2 База данных централизованного горячего водоснабжения.

3.2.1 Участки водопровода (координаты, характеристики труб, техническое состояние участка, данные о вводе в эксплуатацию и ремонтах).

3.2.2 Планирование мероприятий по техническому обслуживанию участков горячего водоснабжения (осмотр, ремонт).

3.2.3 Данные по колодцам (обследование на загазованность, запорно-регулирующая арматура, водомеры, измерители расхода воды, измерители расхода тепла).

3.2.4 Планирование мероприятий по техническому обслуживанию колодцев (осмотр, ремонт).

3.2.5 Данные по насосным станциям (количество, марка, характеристики насоса и электродвигателя, вспомогательное и дополнительное насосное оборудование).

3.2.6 Планирование мероприятий по техническому обслуживанию насосных станций (осмотр, ремонт).

3.3 База данных канализации.



3.3.1 Участки канализации (координаты, характеристики труб, техническое состояние участка, данные о вводе в эксплуатацию и ремонтах).

3.3.2 Планирование мероприятий по техническому обслуживанию участков канализации (осмотр, ремонт).

3.3.3 Данные по колодцам (обследование на загазованность, запорно-регулирующая арматура).

3.3.4 Планирование мероприятий по техническому обслуживанию колодцев (осмотр, ремонт).

### 3.4 База данных теплоснабжения.

3.4.1 Участки теплопровода (координаты, характеристики труб, техническое состояние участка, данные о вводе в эксплуатацию и ремонтах).

3.4.2 Планирование мероприятий по техническому обслуживанию участков теплопровода (осмотр, ремонт).

3.4.3 Данные по колодцам (обследование на загазованность, запорно-регулирующая арматура).

3.4.4 Планирование мероприятий по техническому обслуживанию колодцев (осмотр, ремонт).

3.4.5 Данные по насосным станциям (количество, марка, характеристики насоса и электродвигателя, вспомогательное и дополнительное насосное оборудование).

Планирование мероприятий по техническому обслуживанию насосных станций (осмотр, ремонт).

### 3.5 База данных газоснабжения.

3.5.1 Участки газопровода (координаты, характеристики труб, техническое состояние участка, данные о вводе в эксплуатацию и ремонтах), в зависимости от назначения и давления (высокого, среднего, низкого).

3.5.2 Планирование мероприятий по техническому обслуживанию участков газопровода (осмотр, ремонт).

3.5.3 Данные по газораспределительным устройствам (обследование на загазованность, запорно-регулирующая арматура).

3.5.4 Планирование мероприятий по техническому обслуживанию газораспределительных устройств (осмотр, ремонт).

3.5.5 Данные по колодцам и камерам подземных сооружений (обследование на загазованность, запорно-регулирующая арматура).

3.5.6 Планирование мероприятий по техническому обслуживанию колодцев и камер подземных сооружений (осмотр, ремонт).

3.5.7 Данные по коверам (тип, марка, обследование на загазованность, запорно-регулирующая арматура).

Планирование мероприятий по техническому обслуживанию ковров (осмотр, ремонт).

Данные по сборникам конденсата (тип, марка, диаметр, объем).

Планирование мероприятий по техническому обслуживанию сборников конденсата (осмотр, ремонт).

### 3.6 База данных электроснабжения.

3.6.1 Участки кабельных линий (координаты, характеристики проводов, техническое состояние участка, данные о вводе в эксплуатацию и ремонтах).

3.6.2 Планирование мероприятий по техническому обслуживанию участков электроснабжения (осмотр, ремонт).

3.6.3 Данные по силовым трансформаторам (данные о вводе в эксплуатацию и ремонтах, тип, марка).

3.6.4. Планирование мероприятий по техническому обслуживанию трансформаторов (осмотр, ремонт).

Данные по распределительным устройствам (данные о вводе в эксплуатацию и ремонтах, наименование оборудования, тип, марка).

3.6.6. Планирование мероприятий по техническому обслуживанию распределительных устройств (осмотр, ремонт).

### 3.7 База данных связи.

3.7.1 Участки кабельных линий (координаты, характеристики проводов, характеристики труб, техническое состояние участка, данные о вводе в эксплуатацию и ремонтах).

3.7.2 Планирование мероприятий по техническому обслуживанию участков кабельных линий (осмотр, ремонт).

3.7.3 Данные по распределительным ящикам (тип, марка, данные о вводе в эксплуатацию и ремонтах).

Планирование мероприятий по техническому обслуживанию распределительных ящиков (осмотр, ремонт)

Данные по колодцам (обследование на загазованность).

Планирование мероприятий по техническому обслуживанию колодцев (осмотр, ремонт)

Основные направления работ по систематизации управления недвижимостью как средствами производства

### 3.3.4 Программный комплекс управления объектами недвижимостью.

Основой построения компьютерной системы управления средствами производства служит CAFM (computer aided facility management) – компьютерная поддержка управления средствами производства. Это направление, практическая проработка которого началась ещё в 60-е годы.

На международном рынке ещё нет общепринятых стандартов, так как эти системы ещё находятся в стадии разработки. По технологии управления можно различить три направления

Управление земельными участками

Управление зданиями и сооружениями

Управление техническим оборудованием

Предлагаемая система включает в себя интегрированное управление зданиями, сооружениями, техническим оборудованием в согласовании с трудовыми ресурсами.

Можно выделить отдельные направления:

- Техническое управление зданием
- Управление площадями
- Управление услугами
- Финансовый менеджмент
- Управление персоналом
- Управление коммуникациями

Техническое управление (технический менеджмент) зданием

Под термином Техническое управление зданием понимают всевозможные действия, которые необходимы для нормальной эксплуатации здания, в том числе включающие такие аспекты, как:

- Техническое состояние здания и его отдельных конструктивных элементов,
- Оборудование, инженерные коммуникации и сети, машины и механизмы, обслуживающие здание
- Электротехнические приборы
- Транспортные механизмы (лифты, эскалаторы и т.п.)
- Оборудование для кондиционирования, вентиляции и отопления помещений

- Санитарно-техническое оборудование и обслуживание
- Организация рабочих мест (эргономика)
- Активное и пассивное оборудование техники безопасности
- Охранная сигнализация и оборудование
- Контрольно-измерительные приборы и автоматика
- Надежность несущих конструкций стен, перекрытий, перемычек, лестничных маршей и т.п.
- Состояние гидро и пароизоляции
- Состояние теплоизоляции ограждающих и внутренних конструкций
- Эстетическое состояние объекта (фасады, помещения)
- Управление энергетикой

Работы по техническому обслуживанию разделяются по категориям: плановые периодические, плановые разовые и внеплановые. Последние, как правило, обусловлены форс-мажорными обстоятельствами (как то: прорыв водных коммуникаций, пожар, обрушение конструкций и т.п.). При этом обычно терпится значительный материальный убыток, связанный с порчей или уничтожением оборудования от чрезвычайного воздействия. Поэтому одной из целей технического обслуживания является предотвращения возникновения подобных чрезвычайных обстоятельств. Необходимо предусмотреть вспомогательные работы по профилактическим наблюдениям и контролю технического состояния здания и оборудования. Эти вспомогательные работы должны относиться к категории плановых периодических мероприятий с соответствующими периодами. Одной из функций автоматизированной системы является контроль проведения подобных работ, как то: обследование несущих конструкций, наблюдения за геометрией здания, периодичность проверки систем противопожарной безопасности, сроки профилактики и замены сантехнических и электрических коммуникаций и т.д. Более того, опыт показывает, что в

случае возникновения чрезвычайных обстоятельств пострадавшее лицо имеет право на страховую компенсацию только при точном соблюдении контрольных сроков.

Итак, систематизация мероприятий по техническому обслуживанию, их компьютерный учёт и контроль выполнения приводит к понижению количества отказов систем и оборудования.

### Инспектирование зданий, оборудования и инвентаря

Является составной частью модуля техническое обслуживание. Неравномерная осадка зданий, скрытые дефекты оборудования могут создавать нестандартные ситуации с типичными и нетипичными дефектами. Дефектные ведомости, на основании которых можно выявить причины их возникновения составляются за три этапа: инспектирование; анализ дефектов; выявление слабых мест системы. Дефектные ведомости помогают составлять графики технического обслуживания.

Профилактические работы по контролю несущих конструкций на прочность, жесткость и устойчивость необходимо должны включать в себя элемент геодезических наблюдений за равномерными и неравномерными деформациями и кренами зданий в целом, их отдельных фрагментов и деформациями прилегающей территории. Подобные наблюдения должны проводиться с определенной периодичностью и накапливаться в отдельной базе данных. Это позволит предвидеть зарождение аварийной ситуации и принять обоснованное инженерное решение на начальной стадии. Опыт повышения эксплуатационной надежности зданий и сооружений показывает, что по большинству аварийных объектов не имеется достоверной базы наблюдений. Отсутствие многолетних наблюдений приводит к тому, что зачастую невозможно однозначно определить причины, приведшие к

аварийному состоянию здания. В свою очередь, это приводит к снижению качества и степени обоснованности инженерных решений.

Таким образом, организация геодезического мониторинга объекта и накопление базы данных является одной из важных составных частей технического менеджмента объектов строительства.

#### Актуализация данных

Актуализация данных производится на основании отслеживании изменений в процессе проектирования, строительства, эксплуатации объектов недвижимости, на протяжении жизненного цикла оборудования. Создание компьютерной модели здания на этапе проектирования и постепенное насыщение информацией о поступлении, перемещении, ремонте и выбытии объектов учёта позволяет управлять всеми объектами и происходящими событиями в автоматизированном виде. Электронное проектирование вариантов изменений и их последствий позволяет отойти от «посмертного» учёта, используемого сейчас.

Вследствие различий в целях осуществления управления реализуются различные подходы к решению этих проблем. Предлагаемая система должна установить связи между направлениями.

Отдельные составляющие, обеспечивая единый межблочный формат передачи данных, решают проблемы, возникающие на различных участках объектов и подчиняющиеся различным структурным подразделениям.

#### Управление площадями

Модуль управления площадями является одним из важнейших по значимости. Для рационального использования площадей в здании постоянно необходима актуальная информация о имеющихся в наличии



помещениях с указанием их функционального назначения, проценте их загрузки, информации о пользователях и другой информации.

Как показывает опыт эксплуатации объектами строительства, через несколько лет после начала эксплуатации здания проектная информация перестает быть актуальной. Помещения могут быть разделены перегородками или объединены, перенесены дверные проемы, изменяется функциональное назначение помещений, сменяется их «владелец» (например: аудитория закрепляется за другой кафедрой или лабораторией, сдается в аренду и т.п.). Происходит смена отделочных и облицовочных материалов стен, потолков и пола. В результате подобных операций изменяются проектные характеристики помещений (площадь, объем и т.п.). Из-за изменения функционального назначения производится смена инвентаря и оборудования, закрепленного за помещением, а также изменяются затраты, требуемые для плановых и внеплановых ремонтно-технических работ. Из-за смены инвентаря (например, размещение демонстрационных телевизоров, лингофонного оборудования, увеличение размера доски и т.п.) происходит изменение статуса помещения.

Как правило, подобные изменения практически не находят отражения в проектной документации. Даже разовое выполнение актуализации проекта (а это предполагает проведение трудоемких обмеров и чертежных работ по созданию актуальных планов) устаревает через 3 – 5 лет. Поэтому организация актуализации проектной документации по площадям и помещениям строительного объекта является одной из важнейших задач модуля управления площадями. Текущие изменения в планировке, отделке, составе оборудования помещения должны немедленно вноситься в электронную модель строительного объекта – в графическую и информационную базу. Это обеспечит структуры управления актуальной информацией по объекту в любой момент времени.

Функционально управление площадями может включать в себя следующие блоки и модули:

определение объемов и площадей

анализ функционального назначения

диспетчерская служба (составление расписания, оперативное выделение помещений, смена аудиторий и т.п.)

управление пользователями помещений

менеджмент арендаторов

управление перепрофилированием и перепланировкой площадей

Управление перепрофилированием и перепланировкой площадей

Является составной частью модуля управления площадями. В процессе эксплуатации объектов недвижимости может возникнуть потребность в перепланировке и изменении функционального назначения площадей, которые не требуют привлечения специалистов-проектировщиков и архитекторов. Наличие архивной проектной документации, данных о встроенных коммуникациях, позволяет ответственному за изменения объекта недвижимости определить, насколько возможны подобные изменения. В случае положительного решения создать необходимую документацию и тем самым сократить издержки и время на проведение проектных работ. В случае серьёзных изменений оперативно представить подрядчикам или подготовить к тендерным торгам необходимые исходные данные. Кроме того, за счёт автоматизации работ при вариантном проектировании можно оптимизировать распределение площадей по различным критериям.

Управление услугами

Управление услугам аккумулирует все действия, которые имеют значение для сохранения материальных ценностей, использования и надежности здания. Ими являются:

- Уборка здания
- Общее управление продовольственным снабжением
- Санитарно - гигиенические услуги
- Социальные услуги / социальное обслуживание
- Специфические услуги по обслуживанию
- Службы безопасности
- Курьерские услуги
- Услуги бюро
- Внутренняя обработка почты
- Услуги по переезду
- Уход за земельными участками
- Услуги по сбору и утилизации отходов
- Медицинские услуги
- Экспертное обслуживание
- Услуги прачечной
- Прочие услуги

#### Финансовый менеджмент

Финансовый менеджмент охватывает все отношения между владельцами здания и пользователями здания. В частности, к этой сфере относятся:

- Рынок недвижимости
- Права на земельные участки
- Таксация застроенных и незастроенных земельных участков
- Приобретение земельных участков
- Управление общежитиями, профилакториями, базами отдыха, жилым

фондом

- Администрирование арендных отношений
- Управление договорами

- Управление гарантийными обязательствами
- Бухгалтерский учёт объектов
- Инвестиция и финансирование
- Оперативный мониторинг выполнения договорных обязательств

#### Управление персоналом

В управлении персоналом рассматриваются все вопросы, связанные с персоналом:

- Администрирование персонала
- Подготовка и переподготовка
- Квалификация
- Расчеты заработной платы
- Социальные услуги

#### Управление коммуникациями

Управление коммуникациями направлено на организацию и оптимизацию внутренних и внешних информационных каналов:

- Селекторная связь
- Почтовое сообщение
- Телефония
- Факсимильная связь
- Intranet
- Internet
- ISDN
- Мобильная связь
- Радиосвязь
- Специальная техника

#### Планирование переездов.

Оптимальная загрузка площадей зданий и сооружений с целью максимального удовлетворения по количеству, качеству, функциональному соответствию потребностям пользователей позволяет рационально планировать загрузку площадей. Цепочки переездов связаны с дополнительными проблемами по согласованию сроков запланированных мероприятий, составлением новых расписаний, заказам оборудования, инвентаря, составлением договоров подряда.

Автоматизация учета взаимосвязей при переездах позволит эффективно планировать и управлять этими процессами.

#### Инвентаризация.

Проведение этих мероприятий может быть упрощено применением оборудования для распечатки и считывания штрих-кода. Информация о наличии и состоянии подлежащих инвентаризации объектов позволяет более точно контролировать затраты и предотвращать факты недостаточной обеспеченности ресурсами при выполнении проектных работ.

#### Управление средствами доступа к объектам и охраной

Для многих заведений важно обеспечение простой и удобной системы доступа сотрудников в необходимые помещения, к рабочим местам. Система управления правами доступа позволяет организовать автоматизированный учёт выданных ключей с учётом прав доступа определённого сотрудника в соответствующее помещение. Многие учреждения обладают определенными нематериальными (но представляющими большую ценность) активами, что обуславливает специфику охраны. В этой ситуации актуальной является проблема разграничения полномочий и чёткое их выполнение.

#### Управление договорами

Обычно учебные учреждения имеют обширные связи в различных областях. Долговременные договорные отношения целесообразно

контролировать автоматизировано. Ведение архивов договоров и сопутствующих документов в электронной форме позволит оперативно выполнять их анализ, отслеживать динамику развития договорных отношений со сторонними организациями, накапливать и обновлять базу подрядчиков и заказчиков.

#### Управление уборкой помещений и территорий

За счёт эффективной организации труда по уборке помещений и территорий достигается экономия до 10%4 средств. Определяется определение интервалов уборки, применение различных технических, химических средств и материалов для уборки. При этом учитывается тот факт, что площади в связи с перепланировками и изменением функционального назначения могут изменяться. Автоматизация позволяет точнее учитывать реальные объёмы работ, их стоимость, качество исполнения, вести статистические наблюдения с целью установления цикличности дополнительных видов уборки.

#### Управление персоналом

Эта область управления включает в себя текущее управление персоналом, мероприятия по повышению квалификации и переобучению сотрудников, расчёты заработной платы, решение социальных вопросов. Отчасти управленческие системы в области расчёта заработной платы уже интегрированы с профессиональными системами бухгалтерского учёта. Поэтому необходимо установить возможность конвертации выходных данных в эти системы.

#### Архив планов

Электронная архивация проектной документации позволяет быстро находить любые данные по проекту, изменениям в проекте на любую дату. В любой момент времени можно получить актуальную информацию как по

текущему состоянию объекта, коммуникациям, оборудованию, инвентарю, так и по вариантам проектирования и по истории изменения любой составной части объекта.

Категории информационных потоков

Информационные потоки можно разделить на три категории

Техническая информация

Эксплуатация, ремонт и обслуживание объектов недвижимости

Эксплуатация систем канализации, водоснабжения, вентиляции, энергоснабжения

Эксплуатация контрольно-измерительных приборов и технических средств

Эксплуатация охранной и пожарной сигнализации

Об обслуживающих системах и технике безопасности

Коммерческая информация

Издержки по эксплуатации зданий

Прямые издержки по производственным объектам учёта

Контроль бюджета

Контроль выполнения договоров по условиям, срокам, суммам и санкциям.

Страхование

Контроль выполнения текущих платежей

Инвентаризация

Информация об инфраструктуре

Использование площадей

Изменение назначения используемых площадей

Архивы данных проектной документации

Обеспечение надёжности объекта

---

<sup>4</sup> По данным немецких специалистов.



Работа служб

Расположение и внутренние переезды подразделений

Состояние прилегающих территорий

Связь информационных потоков обеспечивается на основе неразрывной связи между представлением информации в графической и смысловой формах (таблицы, формы, отчеты). При этом обновление данных при работе в графическом редакторе немедленно отражается в базе данных, описывающей сущность объекта. Наоборот, изменение данных в базах вызывает соответствующее изменение в графическом представлении информации.