

ПРАКТИЧЕСКАЯ РАБОТА № 2

АНАЛИЗ ПОДХОДОВ К ФОРМАЛИЗАЦИИ ОПИСАНИЯ СЛОЖНЫХ КОРПОРАТИВНЫХ СИСТЕМ

1. Цель работы

- Получить практические навыки формализации в задачах системной инженерии.
- Получить практические навыки в проведении анализа и формализации функции цели ИС (корпоративной ИС).

2. Теоретические сведения

Задача планирования и эффективного управления предприятиями – одна из основных областей применения информационных технологий, являющимися базой автоматизированных систем управления (АСУ). АСУ может быть представлено в виде совокупности автоматизированных систем взаимодействующих уровней, условно называемых «управление предприятием» (уровень ERP, MRP), управлением производством» (уровень MES) и «управление технологическими процессами и оборудованием» (уровень DCS).

Уровень ERP реализуется автоматизированными системами управления предприятием (АСУП), уровень DCS – автоматизированными системами управления технологическими процессами (АСУ ТП), а важнейшей функцией уровня MES является сопряжение между АСУП и АСУ ТП [7].

Важную роль при анализе и реализации требований к системе играет их специфицирование, для создания систем программного уровня – Software Requirements Specification (SRS). Следует отметить, что часто под термином "система" подразумевается программная система или приложение в разработке. Это может быть крупная коллекция множества компонентов программного и аппаратного обеспечения, одно приложение или компонент программного обеспечения внутри более крупной системы. Во всех этих случаях модель требований описывает поведение, видимое вне системы (через пользовательский интерфейс или API) [1, 2, 3].

Модель описания требований уровня UML во многих случаях не отражает особенности построения узкоспециализированных классов систем в широкой трактовке. Исходя из методики, предложенной А.В. Дабагяном [4, 5], требования характеризуются физическими параметрами, а требования, отличающиеся хотя бы одним признаком или одним значением его параметра, должны быть отнесены к различным классам.

Поле заявок представлено в многомерном евклидовом пространстве. Размерность евклидова пространства равна m , что соответствует отдельным признакам заявки. В поле вектора Z некоторую заявку можно представить в виде вектора Z_i с координатами Z_{ij} , $j \in \{1, \dots, m^1\}$. Полем требований в m^1 -мерном пространстве вектора Z называется замкнутая m^1 – 1-мерная область, границы которой ортогональны осям евклидова пространства, где заданы векторы заявок Z . В той же работе введены определения минимальной и максимальной j -ых границ поля требований.

$$T_i^\chi = [\tau_{1j}^\chi, \dots, \tau_{ij}^\chi, \dots, \tau_{i(m^1-1)}^\chi]^T, Z^\chi = [z_1^\chi, \dots, z_j^\chi, \dots, z_{m^1}^\chi]^T, \quad (1)$$

где i – номер требования; j – номер параметра требования; $(m^1 - 1)$ – размерность вектора требований; T – знак транспонирования; Z^χ – заявка класса χ ; m^1 – число параметров.

$$Z^\chi = \bigcup_{\forall i} T_i^\chi,$$

где i – номер требования класса χ в заявке Z^χ .

Ограничениями данных определений является то, что обслуживание требований одного класса предполагается по одинаковой технологии, а обслуживание требований, входящих в заявки различных классов – по различным технологиям. Однако, реализация *on-line* требований, связанная, в том числе с новыми заявками, не всегда может быть отнесена к существующим технологиям. Следовательно, целесообразной является предварительная оценка времени реализации требования, а также классификация требований и соотнесение их с возможными реализациями.

Фаза анализа и планирования требований в рамках технологии RAD - быстрой разработки приложений [6] связана со следующими работами: 1) определяются функции, которые должна выполнять разрабатываемая информационная система; 2) определяются наиболее приоритетные функции, требующие разработки в первую очередь; 3) проводится описание информационных потребностей; 4) ограничивается масштаб проекта; определяются временные рамки для каждой из последующих фаз; в заключение, 5) определяется сама возможность реализации данного проекта в установленных рамках финансирования, на имеющихся аппаратных и программных средствах.

При использовании методологии быстрой разработки приложений жизненный цикл информационной системы состоит из четырех фаз: • фаза анализа и планирования требований; • фаза проектирования; • фаза построения; • фаза внедрения.

Если реализация проекта принципиально возможна, то результатом фазы анализа и планирования требований будет список функций разрабатываемой информационной системы с указанием их приоритетов и предварительные функциональные и информационные модели системы.

Таким образом, анализ и формализация требований играет важную роль при анализе функционирования системы в целом.

Пример выполнения работы

Пусть требуется построить ИС по следующему описанию: конструкторское бюро по проектированию сложных технических объектов. Предполагается использование большого количества программных средств дизайнерского назначения и средств САПР¹. Предполагаются активная передача информации по

¹ Подготовленная в САПР полная схема является не просто набором чертежей, но и содержит информацию о соединениях всех элементов. С перечнем аппаратуры связаны данные о размещении, габаритах и схемах аппаратов. На их основе формируется база данных проекта, позволяющая использовать ее для создания других документов. Использование именно базы данных проекта, а не отдельных чертежей, позволяет значительно повысить эффективность выпуска рабочей документации проектными организациями.

локальной сети между конструкторами, имеется необходимость защиты особых данных. Особо важна скорость реализации процесса обработки изображений.

Задание: 1. Формализовать требования к системе на основе технологии RAD,
2. Формализовать описание системы в целом.

Часть 1. Исходя из технологии RAD, проводится анализ требований:

1) определяются функции, которые должна выполнять разрабатываемая информационная система;

Из описания системы, определяется множество функций ИС. Глобальная функция: Φ – получение комплексного проекта в виде совокупности изображений.

Поддерживающие и обеспечивающие функции: $\Phi 1$ – обеспечение взаимодействия многих программ дизайнерского назначения и средств САПР; $\Phi 2$ – поддержка интенсивного обмена информацией по локальной сети между конструкторами; $\Phi 3$ – функция защиты особых данных;

$\Phi 4$ – поддержка высокой скорости реализации процесса обработки изображений.

Φ : $\Phi 1, \Phi 2, \Phi 3, \Phi 4$ (2)

2) определяются наиболее приоритетные функции, требующие разработки в первую очередь;

$\Phi 1$ – обеспечение взаимодействия многих программ дизайнерского назначения и средств САПР можно считать базовой функций разрабатываемой ИС, поскольку исходя из способа реализации $\Phi 1$ будут обеспечиваться остальные функции.

3) проводится описание информационных потребностей;

Информационными потребностями пользователя являются конкретные (фактические) значения требуемых параметров системы, которые задаются заказчиком проекта. Например, объемы текущих данных, требуемая скорость обмена данными, или время, необходимое на передачу данных.

Эти данные выбираются либо по согласованию с преподавателем, либо выбираются самостоятельно, исходя из среднестатистических данных, например, размера некоторого изображения САПР в байтах (формат чертежа A1), скорость обмена порядка 2Мбайт/сек и т.п.

4) ограничивается масштаб проекта;

Ограничения могут быть связаны с уточнениями описания модели. Для приведенного примера это могут быть ответы на вопросы: какой тип САПР используется? (От этого зависит размер картинки, стоимость разработки). Например:

1. САПР двумерного проектирования — «2D-3D Легкие — Нижний уровень»²

Эти САПР служат для выполнения почти всех работ с двумерными чертежами и имеют ограниченный набор функций по трехмерному моделированию. С помощью этих систем выполняются порядка 90% всех работ по проектированию. Область их работы — создание чертежей отдельных деталей и сборок. Платой за возросшие возможности является усложнение интерфейса и меньшее удобство в работе. Характерные

² <http://www.automationlabs.ru/>

представители таких САПР — AutoCAD, CADdy, CADMECH Desktop, MasterCAM, T-FlexCAD, OmniCAD, Компас-График.

2. САПР объемного моделирования «3D — Средний уровень»

По своим возможностям они полностью охватывают САПР «легкого веса», а также позволяют работать со сборками, по некоторым параметрам они уже не уступают тяжелым САПР, а в удобстве работы даже превосходят. Обязательным условием является наличие функции обмена данными (или интеграции). Это не просто программы, а программные комплексы, в частности, SolidWorks SolidEdge, Cimatron, Form-Z, Autodesk Inventor, CAD SolidMaster, и все еще продолжающий развиваться, Mechanical Desktop, DesignSpace.

3. САПР объемного моделирования «3D Тяжелые — Верхний уровень»

Эти системы применяются для решения наиболее трудоемких задач - моделирования поведения сложных механических систем в реальном масштабе времени, оптимизирующих расчетов с визуализацией результатов, расчетов температурных полей и теплообмена и т.д. Обычно в состав системы входят графические, модули для проведения расчетов и моделирования, постпроцессоры для станков с ЧПУ. Эти самые мощные САПР наиболее громоздки и сложны в работе, а также имеют значительную стоимость. Примерами «тяжелых» САПР могут служить такие продукты, как ADAMS, ANSYS, CATIA, EUCLID3, Pro/ENGINEER, UniGraphics.

Стоимость всех САПР соотносятся по уровням следующим образом:

- Нижний: \$500-\$2000 за рабочее место (AutoCAD, AutoCAD LT, Компас);
- Средний: \$2000-\$20000 (Inventor, Mechanical Desktop, SolidWorks);
- Верхний: более \$20000 (ProEngineer, Unigraphics).

Кроме того, ограничениями может быть уточнение требуемой скорости обмена данными и т.п.

5) определяется возможность реализации данного проекта в установленных рамках ограничений.

Этот этап служит для анализа требований на совместимость и непротиворечивость.

В примере: при ограничении ресурсов по финансированию проекта применение САПР верхнего уровня может привести к полной невозможности его реализации. Или ограничение на скорость обмена данными вследствие применения определенного типа аппаратуры может сказаться на реализации соответствующего требования.

Таким образом, анализируются функции и определяются количественные требования.

Формализация требований осуществляется посредством их символьного представления согласно (1) и (2). **Функции системы:**

$\Phi: \Phi_1, \Phi_2, \Phi_3, \Phi_4$,

где Φ – получение комплексного проекта в виде совокупности изображений.

Φ_1 – обеспечение взаимодействия многих программ дизайнерского назначения и средств САПР; Φ_2 – поддержка интенсивного обмена информацией по локальной сети между конструкторами; Φ_3 – функция защиты особых данных;

Φ_4 – поддержка высокой скорости реализации процесса обработки изображений.

Требования к системе. Определим поле требований как вектор требований,

их шкал и ограничений:

$$T_1^{\Phi 1} = [\tau_{11}^{\Phi 1}, \tau_{12}^{\Phi 1}],$$

где $T_1^{\Phi 1}$ – требования класса $\Phi 1$: $\tau_{11}^{\Phi 1}$ – наличие гетерогенной информационной среды; $\tau_{12}^{\Phi 1}$ – наличие интерфейса взаимодействия различных элементов гетерогенной информационной среды.

При этом, если $T_1^{\Phi 1} \in \overline{0,1}$, то $\tau_{11}^{\Phi 1}$, $\tau_{12}^{\Phi 1}$ могут быть трактованы как степени принадлежности к нечеткому множеству соответствия, а если $T_1^{\Phi 1} = \overline{0,N}$, где N – общее кол-во возможных значений $T_1^{\Phi 1}$, то $\tau_{11}^{\Phi 1}$, $\tau_{12}^{\Phi 1}$ имеют смысл цифровых количественных оценок; при $T_1^{\Phi 1} \in \overline{0,1}$, то $\tau_{11}^{\Phi 1}$, $\tau_{12}^{\Phi 1}$ – бинарные определители наличия или отсутствия признака требования: есть гетерогенная среда или нет.

$$T_1^{\Phi 2} = [\tau_{11}^{\Phi 2}, \tau_{12}^{\Phi 2}],$$

где $T_1^{\Phi 2}$ – требования класса $\Phi 2$: $\tau_{11}^{\Phi 2}$ – обеспечение заданной скорости обмена данными; $\tau_{12}^{\Phi 2}$ – обеспечение заданной надежности обмена данными. При ограничениях: $\tau_{11}^{\Phi 2} \geq 2 \text{ Мб/сек.}$, $0.8 \leq \tau_{12}^{\Phi 2} \leq 1$.

Аналогичным образом строятся множества $T_1^{\Phi 3} \dots T_K^{\Phi 3} = [\dots]$, $T_1^{\Phi 4} \dots T_L^{\Phi 4} = [\dots]$, где K , L – количество типов требований каждого класса Φi .

Часть 2. Формализация описания системы в целом.

Следует разграничивать **формализованное описание процесса проектирования системы, процесса функционирования системы и общее описание системы**. Очевидно, что формализация в первом и втором случаях будет связана с динамическими процессами, а описание системы в целом, в основном, – статическая или информационная ее интерпретация.

В Таблице 1 приведены основные способы представления формального описания сложных систем. Поскольку глобальная функция системы представляет собой формирование нового информационного объекта (Φ – получение комплексного проекта в виде совокупности изображений), можно использовать при описании сочетание п.4 и п.2.

Таблица 1. Основные способы описания сложных систем

Определение	Формальное описание	Описание
Система есть множество элементов, свойств и отношений	$S = (\{m\}, \{n\}, \{r\})$, где m – элементы, n – свойства, r – отношения	Знаково-графовый подход
Система есть множество элементов, образующих структуру и обеспечивающих определенное поведение в условиях окружающей среды	$S = (\varepsilon, ST, BE, E)$, где ε – элементы, ST – структура, BE – поведение, E – среда	Структурно-поведенческий подход
Система есть множество входов, множество выходов, множество состояний, характеризующихся	$S = (X, Y, Z, H, G)$, где X – входы, Y – выходы, Z – состояния, H – оператор	Учитываются все основные компоненты,

Определение	Формальное описание	Описание
оператором переходов и оператором выходов	переходов, G - оператор выходов	рассматриваемые в автоматике.
Понятия модели F , связи SC , пересчета R , самообучения FL , самоорганизации FQ , проводимости связей CO и возбуждения моделей JN	$S=(F, SC, R, FL, FO, CO, JN)$	Определение применяется для описания нейрокибернетических систем.
Введен фактор времени и функциональных связей	$S=(T, X, Y, Z, \Omega, V, \eta, \varphi)$, где T - время, X - входы, Y - выходы, Z - состояния, Ω - класс операторов на выходе, V - значения операторов на выходе, η - функциональная связь в уравнении $y(t_2) = \eta(x(t_1), z(t_1), t_2)$, φ - функциональная связь в уравнении $z(t_2) = \varphi(x(t_1), z(t_1), t_2)$.	Определение системы, которым оперируют в теории автоматического управления
Организационные системы	$S9=(PL, RO, RJ, EX, PR, DT, SV, RD, EF)$, где PL - цели и планы, RO - внешние ресурсы, RJ - внутренние ресурсы, EX - исполнители, PR - процесс, DT - помехи, SV - контроль, RD - управление, EF - эффект.	Организационное управление

Таким образом, возможно применить следующее обобщенное описание

$$S = (X, Y, Z, H, G, \{m\}, \{n\}, \{r\}),$$

где X - входы, Y - выходы, Z - состояния, H - оператор переходов, G - оператор выходов, m - элементы, n - свойства, r - отношения между элементами.

Пример. Пусть дана гидрометеорологическая система (ГМИС) с возможностью сбора, анализа и хранения данных. Опишем функционирование этой ГМИС с требованиями.

Проектируемая система производит преобразование

$$Y_t = F(X_t, T_t),$$

где X_t – текущее состояние входного объекта системы в момент времени t ; Y_t – текущее состояние выходного объекта системы в момент времени t ; T_t – текущее требование к системе; F – функция преобразования входного объекта системы.

Система также может быть описана как совокупность элементов и ограничений

$$S = \left[\bigcup_{n=1}^N \{X_i^{(n)} \mid (\delta_i \leq \delta^{X_i})\}, \{Y_j^{(k)} \mid (\theta_j \leq \theta^{Y_j})\}, \{T_t \mid T_i^{\Phi_j} = [\tau_{i1}^{\Phi_j}, \dots, \tau_{iN}^{\Phi_j}]\}, \Phi, (Z : z_i \geq z^*) \right],$$

где $\bigcup_{n=1}^N \{X_i^{(n)}\}$ – объединение подмножеств входных объектов (данных) системы

(если входные данные возможно классифицировать по подмножествам), $n = \overline{1, N}$;

δ_i – требуемое качество входных данных;

δ^{X_i} – допустимое качество входных данных (нижняя граница);

θ_j – качество выходных данных;

θ_j^Y – допустимое качество выходных данных (*нижняя граница*);

T_i – текущее требование к системе;

I, J – общее кол-во i -требований класса j ;

$\Phi: \Phi_1, \Phi_2, \Phi_3, \Phi_4$ – множество функций системы;

Z – множество состояний системы, при которых достигается требуемый уровень защиты данных: обеспечение Φ_3 – функции защиты особых данных (z_i – i -е состояние системы;

z^* – необходимое состояние системы после реализации i -го требования к системе в текущем состоянии;

Z_w – множество верхних граней ($\sup(W)$) некоторого множества W , равных или больших всех элементов W (*это утверждение может быть записано в следующей форме: $z^* = \sup(W) \mid z_i \in Z_w, \forall s \in Z_w : z^* \leq s$*).

Описание функции цели может быть определено по следующей схеме.

Пусть $F(T_j^{\Phi_i})$ – функционал (*реализация требований к системе*), определенный на множестве $T_j^{\Phi_i}(X)$ (*требования в рамках входных данных*). Рассматривается следующая задача нахождения варианта системы X :

$$x \in \arg \max_{x \in D} F(x); \arg \max_{x \in D} F(x) = \{x \in D \mid F(x) = \max_{x \in D} F(x)\}, \quad (3)$$

где $F(x)$ – искомый функционал.

Вариант $x \in D$, при котором достигается максимум функционала $F(x)$ на некотором множестве $X^{(s_0)} \subseteq X$, называют максимальным. Полученные максимальные варианты исследуются с помощью критериев оптимальности.

Таким образом ставится задача нахождения такого варианта системы из области допустимых (D), который обеспечит реализацию текущего требования с достижением верхней границы его качества.

Примечание: Приведенные элементы примера формализации требований и функции цели не обладают высокой степенью общности, а представляют собой конкретную реализацию определенной задачи.

3. ХОД РАБОТЫ

1. Определить вариант задания из списка (либо согласовать с ведущим преподавателем или руководителем НИР).
2. Изучить литературу и теоретические сведения к выполнению лабораторной работы.
3. Привести формальное описание системы в целом.
5. Предложить функцию цели для проектирования системы, сделать выводы.
6. Сформировать и защитить отчет, содержащий все этапы исследования.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Описать основные задачи технологии RAD.
2. Каким образом учитываются требования к системе в ее формализованном описании?
3. Какие существуют подходы к формальному описанию систем?
4. Привести примеры отличающихся подходов формального описания сложных систем.
5. Что такое **H** - оператор переходов, **G** - оператор выходов?
6. Привести пример описания реальной системы, когда должны быть учтены обратные связи в операторах преобразования объектов.
7. Прочесте и объясните выражение: $x \in \arg \max_{x \in D} F(x)$.
8. Каким образом можно описать требование обеспечения верхней (нижней) грани множества результатов? Привести пример записи.
9. Для чего требуется формализовать (описать) полный перечень функций системы?
10. Применительно к теме своей НИР охарактеризовать функцию цели.

5. ВАРИАНТЫ ЗАДАНИЙ

Вариант задания может быть выбран из предложенного списка по последней цифре номера зачетной книжки либо выбирается студентом индивидуально, в соответствии с тематикой выпускной (квалификационной) магистерской работы по согласованию с её руководителем.

Вариант 0. Конструкторское бюро по проектированию сложных технических объектов. Использование унифицированной САПР. Данные сохраняются в БД. Важность формирования целостного изображения.

Вариант 1. Станция экологического контроля параметров на химическом предприятии. ИС должна функционировать в оперативном режиме работы, решения об уровне загрязнения принимает система, подтверждает оператор. Необходимость защиты данных для хранения и при передаче на другие объекты.

Вариант 2. Предприятие по выпуску пищевой продукции обеспечено автоматизированной системой управления. Основные этапы технологического процесса: передача объектов по конвейеру, при котором необходим контроль качества выполненных операций. ИС собирает информацию со специальных датчиков и обрабатывает ее, сравнивая показатели объектов на конвейере с контрольными, при несовпадении отбраковывает, снимая с конвейера. Данные сохраняются в БД.

Вариант 3. Предприятие интернет торговли. Информационная система управляет пользователями, обеспечивает доступ, регистрацию и возможность скидок. Наличие конкурентов должно быть сопоставлено с возможностью защиты данных в режиме реального времени и их сохранности.

Вариант 4. Юридическая контора обслуживается ИС с двумя БД различного назначения (клиенты, их дела и нормативно-правовая информация). При этом БД связаны. БД клиент-дело особо секретна, БД справочная открыта и имеет доступ по всей юридической компании. Юристы, ведущие дела, могут оперативно добавлять информацию по делам по мобильной связи.

Вариант 5. Предприятие лечебно-диагностического характера. ИС клиентов, где реляты процедуры, описание диагнозов и процесса лечения. Поступление данных в БД из различных источников: датчики и устройства (напрямую с ультразвукового аппарата, от магнитно - резонансной диагностики, носимых устройств: например, пульс, частота дыхания, аритмия). Необходимость быстрого реагирования на особые ситуации в текущем режиме (информация с носимых устройств) и при диагностике.

Вариант 6. Администрация учебного заведения. ИС «школьный дневник» с возможностью обмена сообщениями с классным руководителем родителями учащегося при нестандартных ситуациях в оперативном режиме. Поступление данных с различных источников: проходная школы, журнал оценок, столовая, медицинский кабинет. Возможность сохранения и анализа данных за неделю, месяц, год.

Вариант 7. Предприятие контроля параметров технических устройств. Датчики и средства контроля передают комплекс информации о контролируемом устройстве в БД, откуда происходит их извлечение и анализ. Далее, на основании анализа формируется отчет о состоянии устройства. Возможность реализации контроля в скоростном режиме.

Вариант 8. Бюро технической инвентаризации (БТИ). Данные о состоянии жилого фонда вносятся как вручную, так и с помощью устройств фотосъемки, аудиозаписи (например, при определении шума при изоляции квартиры). Возможность передачи данных по локальной сети между подразделениями БТИ. Совместимость с другими БД.

Вариант 9. Станция гидрологического контроля параметров в морском порту. ИС должна функционировать в оперативном режиме работы, требуется наличие совместимых форматов с другими БД подобного профиля. Контроль параметров осуществляется как автоматически, так и вручную. Возможность предупреждения об опасных явлениях (резкое повышение уровня моря, увеличение скорости течения).

6. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Правила составления Software requirements specification [Электронный ресурс] / Режим доступа: <http://habrahabr.ru/blogs/development/52681/>. – Загл.с экрана.
2. Рамбо Дж. UML 2.0. Объектно-ориентированное моделирование и разработка / Дж. Рамбо. – McMillan, 2007.
3. Резник А. Развивающиеся системы [Электронный ресурс] / А. Резник//

International Conference «Knowledge-Dialogue-Solutions», 2007. – Режим доступа: <http://www.foibg.com/conf/ITA2007/KDS2007/PDF/KDS07-Reznik1.pdf>. – Загл. с экрана.

4. Дабагян А.В. Принципы автоматизированного проектирования систем машин и техноологических процессов: Учеб.пособие. – Харьков:ХПИ, 1987. – 66 с.

5. Дагабян А.В. Проектирование технических систем. – М.: Машиностроение, 1986. – 256 с.

6. Петров В.Н. Информационные системы / В.Н. Петров. – СПб.: Питер, 2003. – 688 с.

7. Яковлев В.П. Корпоративные информационные системы: конспект лекций / В.П. Яковлев; СПбГТУРП. – СПб., 2015. – 117 с.