

УДК 005.004

В. Б. Задоров, А. А. Васильев

*Київський національний університет будівництва і архітектури, Київ*

## К РАЗВИТИЮ КОНЦЕПЦИИ «КОНФИГУРАТОРОВ» ДЛЯ ПОСТРОЕНИЯ АРХИТЕКТУРЫ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ОРГАНИЗАЦИОННЫХ АНТРОПОГЕННЫХ СИСТЕМ

*Рассмотрены пути создания и развития современных эффективных интеллектуальных информационных технологий на основе развития концепции «конфигураторов». Предложена принципиальная структура полноценного конфигулятора ИТ для сложных меняющихся в процессе своей деятельности бизнес-систем. Разработана обобщенная архитектура КИС бизнес-систем на основе такого конфигулятора.*

**Ключові слова:** *компьютерные информационные системы (КИС); конфигурирующая система; аналитическая система модельной конфигурации; конфигурированная система; сфера конфигурирующей системы; сфера аналитической системы модельной конфигурации; сфера конфигурированной системы; библиотека прикладных программных средств (БППС)*

### Постановка проблемы

Организационные антропогенные системы (ОАГС) относятся к классу сложных создаваемых человеком искусственных динамических систем. Наиболее сложными из них являются бизнес-системы: от отдельных предприятий, корпораций до социально-экономических систем различных территориальных объединений. Сегодня уже нельзя представить их эффективное функционирование без информационных технологий, реализующих задачи анализа деятельности, синтеза построения и принятия решений, проектирования и управления. Информационные технологии становятся неотъемлемой частью бизнес-систем, а в некоторых случаях - самой весомой частью, например, в оценке стоимости всей бизнес-системы [5;6;7].

Современная индустрия информационных технологий, обеспечивающая сегодня функционирование бизнес-систем на разных стратифицированных уровнях, накопила богатый опыт. Все больше наблюдается переход от разработки индивидуальных уникальных комплексов программ для компьютерных проектирующих и управляющих информационных систем (КИС) предприятий и корпораций к интеграции функционально-ориентированных типовых прикладных промышленных инструментальных программных продуктов для таких КИС [2;10;14].

С середины девяностых годов двадцатого столетия развивается концепция реинжиниринга бизнес-систем, в основе которой лежит утверждение

о том, что изменения в бизнес-системах (в их структурах и в деятельности) в современных условиях имеют не эволюционный характер, а радикальный, т.е. изменения функций бизнес-систем, бизнес-процессов, организационных структур носят качественный характер [5;6;7;9;11;14].

Учет этого свойства современных бизнес-систем усложняет работу с требованиями к разработке и развитию информационных технологий в составе бизнес-систем [2;10]. Традиционные КИС предприятий, корпораций и пр., как правило, ориентированы на реализацию требований, обеспечивающих деятельность предприятий в стабильных многократно повторяющихся стандартных ситуациях. Эффективный учет возникающих изменений в структуре и деятельности бизнес-систем требует новых подходов, как к итерационному процессу создания КИС, так и к самой функциональной структуре таких средств поддержки жизнеобеспечения бизнес-систем. Одним из таких подходов, зародившихся в последние десятилетия в рамках реинжиниринга бизнес-систем, является моделирование бизнес-процессов. Программные продукты сложных корпоративных КИС при таком подходе базируются на референтных моделях бизнес-процессов, которые адаптируются к условиям конкретных бизнес-систем.

С позиций разработчика потребность в учете индивидуальной специфики конкретных бизнес-систем на этапах разработки требований, проектирования и реализации КИС достаточно

технологічно реалізується засобами комплексно проработаних об'єктно-орієнтованих методологій проектування програмних продуктів [1;2;3]. Однак ці засоби професійно не орієнтовані на менеджерів замовника і його системних аналітиків (постановщиків проблеми від замовника). Тому на етапі розробки вимог до створюваної КИС спеціалістам з боку замовника і з боку розробника доводиться прикладати більші зусилля для забезпечення формалізованого представлення моделей бізнес-процесів предметної області (ПО), що є необхідним при переході до наступним етапам проектування і розробки КИС. Таке положення пояснимо, з однієї сторони, складністю формалізації творчих устремлень і побажань замовника, а з іншої сторони, розробаністю моделей представлення ПО в методологіях ООП тільки з позицій розробника програмного продукту і, як правило, не затрагиваючих цілі і задачі попереднього моделювання ПО замовником.

Відсутність єдиного мови представлення моделей ПО, зрозумілого представителю замовника і розробника на етапі попереднього моделювання ПО, ускладнює рішення складних питань, пов'язаних з аналізом цілей, функцій і ситуацій в бізнес-системі замовника. В той же час він повинен вирішити ці питання раніше, ніж почне виконуватися розробка і аналіз вимог до інформаційної системи. Замовнику тут дуже важливо спочатку зрозуміти для себе на модельному рівні, що і як буде робити майбутня система в комплексі.

Якщо сьогодні досвід такого моделювання ПО в інтересах розробників КИС вже накоплено достатньо великим [1;3;13], а для моделювання ПО в інтересах замовника теж існує певний досвід [11;15;22], то єдиною технологією багатогранного аналізу і синтезу ОАГС на ранніх етапах створення і розвитку ОАГС в комплексі з КИС поки немає. Тому необхідно подальше вдосконалення існуючих підходів, які дозволяють би зближати наявні різнонаправлені методології аналізу і синтезу бізнес-систем в інтересах замовників і розробників КИС.

### **Аналіз основних досліджень і публікацій**

В межах реалізації об'єктно-орієнтованого підходу існує ряд технологій розробки прикладних програмних систем, в основі яких лежить побудова і інтерпретація на комп'ютері моделей цих систем. К основним з таких технологій можна віднести

наступні об'єктно-орієнтовані методології аналізу і розробки програмних систем: ОМТ (Object Modeling Technique), SA/SD (Structured Analysis/Structured Design), JSD (Jackson Structured Development), ОSA (Object-Oriented System Analysis), Проклос (Проектування в кластерній середі) [1;3]. Ці методології підтримують моделювання ПО і майбутньої програмної системи на перших двох етапах ЖЦ програмного продукту:

- на етапі аналізу вимог і попереднього проектування програмної системи, коли розглядаються моделі бізнес-процесів функціонування бізнес-системи і КИС без урахування аспектів, пов'язаних з майбутньою реалізацією прикладного програмного продукту (ППП), т.е. тут увага приймається тільки прагматичні аспекти ПО;
- на етапі проектування (конструювання), на якому приймаються основні рішення, пов'язані з реалізацією проектуваної програмної системи, а її структура, розроблена на першому етапі, змінюється з урахуванням вимог її ефективності.

Найбільш широко відома технологія *ОМТ* (Object Modeling Techniques) [1,3]. Вона мала велике вплив на інших розробників об'єктно-орієнтованих технологій. Проектувана програмна система представляється тут в формі трьох взаємопов'язаних моделей:

- об'єктної моделі, яка відображає статичні, структурні аспекти системи, в основному пов'язані з даними;
- динамічної моделі, яка описує роботу окремих частин системи;
- функціональної моделі, в якій розглядається взаємодія окремих частин системи (як за даними, так і за керуванням) в процесі її роботи.

Ці три види моделей дозволяють отримати три взаємопов'язаних представлення системи в одній системі позначень. Ці моделі системи можуть бути проаналізовані з допомогою інструментального програмного забезпечення з метою модельної демонстрації замовнику функціонування майбутньої системи. Методологія ОМТ підтримується системою Paradigm+, однією з найбільш відомих інструментальних систем об'єктно-орієнтованої розробки. Якщо умови функціонування і розвитку бізнес-системи, т.е. вимоги замовника, в цих моделях повністю враховані, то це суттєво спрощує узгодження попереднього проекту системи.

Моделі, розроблені та опрацьовані на першій фазі життєвого циклу системи, продовжують використовуватися на всіх наступних його фазах, облягаючи програмування системи, її налагодку та тестування, супроводження та подальшу модифікацію.

Здесь следует обратить внимание на следующие позитивные моменты:

- во-первых, на комплексность информационного моделирования ПО на первых двух стадиях создания КИС, учитывающего статику и динамику, а также схемы функционирования бизнес-системы;
- во-вторых, на независимость этих моделей системы от среды программирования, в которой она будет реализована;
- в-третьих, технологичность возможной модифицируемости комплекса этих моделей на последующих стадиях разработки и реализации программного продукта.

Существенные недостатки этих методологий можно свести к следующему:

- остается вне средств методологии моделирование целей, задач бизнес-систем, а также анализ ситуативности ее функционирования, т.е. они вербально определяются менеджментом заказчика, уточняются и корректируются в процессе достаточно длительных согласований с разработчиками на первом этапе ЖЦ создания КИС;
- отсутствие интегрированности с программными средствами моделирования, с помощью которых менеджеры могли бы анализировать свои цели, задачи, функции и ситуации в условиях изменений внешних и внутренних факторов на всех этапах создания и развития бизнес-систем и КИС.

К программным средствам, с помощью которых менеджмент имеет возможность моделировать динамику функционирования бизнес-системы и анализировать свои цели, задачи, функции и ситуации, относят средства интеллектуального анализа данных, получившие общее название – Data Mining, что переводится как «добыча» или «раскопка данных» [4,11]. Нередко рядом с Data Mining встречаются слова «обнаружение знаний в базах данных» (knowledge discovery in databases) и «интеллектуальный анализ данных». Их можно считать синонимами Data Mining.

Цель Data Mining состоит в выявлении скрытых правил и закономерностей в наборах данных. Дело в том, что человеческий разум сам по себе не приспособлен для восприятия больших массивов разнородной информации. Человек к тому же не способен улавливать более двух-трех взаимосвязей даже в небольших выборках. Но и традиционная математическая статистика, долгое

время претендовавшая на роль основного инструмента анализа данных, также нередко пасует при решении задач из реальной сложной жизни. Она оперирует усредненными характеристиками выборки, которые часто являются фиктивными величинами.

В подходе, заложенном в концепцию Data Mining, в принципе, нет ничего нового в постановке задачи анализа. Специалисты на протяжении нескольких последних десятилетий решали подобные задачи, используя различные методы («поиск эмпирических закономерностей», «эвристический поиск в сложных средах», «индуктивный вывод» и т. п.). Но только сейчас с развитием индустрии информационных технологий у аналитиков появляются возможности решать практические задачи и сложные проблемы в области системного анализа и синтеза ОАГС в комплексе с разработкой КИС.

Выделяют пять стандартных типов закономерностей, которые используются в методах анализа Data Mining: *ассоциация*, *последовательность*, *классификация*, *кластеризация*, *прогнозирование*. *Ассоциация* позволяет анализировать результат, полученный при изучении событий, связанных между собой. *Последовательность* позволяет изучать влияние на результат цепочки связанных событий во времени.

С помощью *классификации* анализируются различные признаки, характеризующие группу, к которой принадлежит тот или иной объект. Это делается посредством анализа уже классифицированных объектов и формулирования некоторого набора правил.

*Кластеризация* отличается от классификации тем, что сами группы заранее не заданы. С помощью кластеризации средства Data Mining самостоятельно в процессе анализа выделяют различные однородные группы данных. Основой для всевозможных систем *прогнозирования* служит историческая информация, хранящаяся в БД в виде временных рядов. Если удастся построить математическую модель и найти шаблоны, адекватно отражающие эту динамику, есть вероятность, что с их помощью можно предсказать и поведение системы в будущем.

Наряду с включением в состав средств Data Mining статистических пакетов, усложняющих работу пользователей без специальной подготовки, в них используются новые интеллектуальные средства, базирующиеся на нейронных сетях, на системах рассуждений на основе аналогичных случаев (case based reasoning — CBR), на деревьях решения (decision trees), на генетических алгоритмах, на алгоритмах ограниченного перебора и т.п. [4;11;15].

Таким образом, перспективным направлением дальнейшего развития средств систем подготовки и принятия решений (СППР) является применение комбинированных методов принятия решений в сочетании с технологиями Data Mining, методами искусственного интеллекта и компьютерным моделированием, а также с использованием различных имитационно-оптимизационных процедур и принятия решений в сочетании с экспертными процедурами [4;11].

### Формулирование цели статьи

Создание современных интеллектуальных компьютерных информационных технологий и на их основе эффективно функционирующих бизнес-систем должно базироваться на интеграции существующих концепций:

- концепции развития средств Data Mining для использования на всех стадиях жизненного цикла бизнес-систем с целью обеспечения циклического анализа функционирования бизнес-систем и программных средств;
- концепции синтеза программных продуктов, обеспечивающих эффективное функционирование бизнес-систем.

Одним из таких подходов является концепция «конфигураторов» [6;21]. Ее анализу и развитию уделено внимание в этой работе.

### Изложение основного материала

Сравнительный анализ подходов к разработке конфигураторов в информационных технологиях показывает, что в настоящее время не существует единого определения конфигуратора вообще и конфигуратора в информационных технологиях в частности.

Слово «конфигуратор» пошло от английского «configuration» (конфигурация), которое, если говорить обобщенно, означает устройство или состав любой системы.

Приведем примеры некоторых определений, взятых из сети Интернет [18;19;20;21;22;23], которые частично описывают это понятие:

- Конфигуратор – программа, которая позволяет конфигурировать (структурировать и организовывать) какие-то инженерные системы (например, автомобили, компьютеры) с некоторой заданной целью. Например, минимизировать стоимость при выполнении заданных функциональных ограничений.
- Конфигуратор – программа, которая позволяет структурировать, т.е. конфигурировать функциональные программные продукты.

- Конфигуратор – комплекс программ, который позволяет структурировать и организовывать комплексные ИТ для инженерных, коммуникационных, экономических, организационных предметных областей с реализацией множества функциональных заданий с заданной конечной целью и адаптацией к конкретным типовым ситуациям в этих предметных областях.

Проведенный авторами анализ понятия «конфигуратора» как системы показал, что в любом конфигураторе можно выделить три части:

- система, которая поддается конфигурации (конфигурируемая или конфигурированная система);
- система, которая определяет конфигурацию (аналитическая система модельной конфигурации);
- система, которая выполняет конфигурирование (конфигурирующая система).

Принципиальная схема «конфигуратора», предложенная А.А.Васильевым представлена на рис.1.

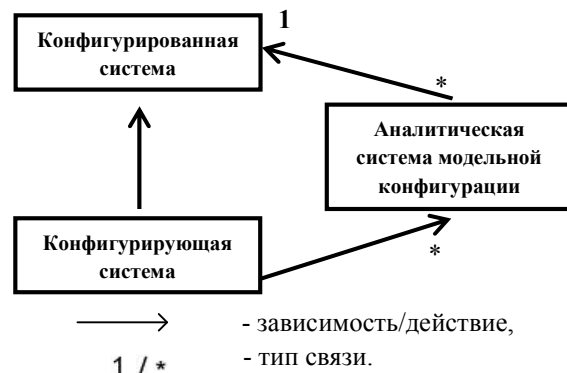


Рис.1. Принципиальная схема конфигуратора. Рассмотренная схема конфигуратора наиболее широко используется в сети Интернет

Web-страницы выполняют роль *конфигурирующей системы*, тогда как система, которая конфигурируется, чаще всего является только аналитическим отображением реальных предметов, например, конфигураторы автомобилей, компьютеров, интерьера, мебели и т.п. В данном случае *аналитической системой модельной конфигурации* является информация о том, какие составляющие, в каком количестве и с какими свойствами используются для того, чтобы был описан конкретный автомобиль, собранный ПК или оснащенная мебелью комната. Затем, как правило, эта информация (уже как *конфигурированная система*) сохраняется и отсылается менеджеру для согласования с клиентом.

Рассмотрим более сложные конфигураторы – те, которые обеспечивают конфигурирование реальных систем ИТ. В таких конфигураторах все три составляющие (конфигурированная система,

аналитическая система модельной конфигурации и конфигурирующая система) являются полноценными и явно выраженными. К ним, например, относятся:

- конфигураторы Web – приложений, такие как: конфигураторы Интернет – форумов, конфигураторы Интернет – магазинов, конфигураторы любых систем, которые базируются на Web-интерфейсе или имеют Web-интерфейс (например, багтрекинг-система LRA);
- конфигураторы систем ИТ в целом и распределенных систем: 1С-предприятие, 1С-Битрикс - Управление сайтом, конфигураторы инженерных САПР – специальные продукты Autodesk, конфигураторы систем ИТ экономических организационных систем, например, SAP R3 i BAAN 4, конфигуратор системных блоков для упрощения процесса подбора комплектующих в составе сборного изделия, Shoreline Firewall – инструмент для настройки межсетевых экранов в Linux.

Рассмотрим функции созданных и используемых конфигураторов систем ИТ более подробно.

**1С-Предприятие** представляет собой универсальный комплекс программ массового назначения для автоматизации функций управления, бухгалтерского и налогового учета в организациях, осуществляющих производственную и коммерческую деятельность [19]. Учет ведется, в основном, с использованием электронных документов – аналогов бумажных документов. При проведении проводок их корректность обеспечивается алгоритмами работы конфигурации, иногда достаточно сложными, как например, расчет события для учета НДС и налога на прибыль. Данное прикладное решение в системе можно использовать только для ведения бухгалтерского и налогового учета, а для задания автоматизации функций других служб, например, отдела продаж, решается специализированными конфигурациями или другими системами.

**1С – Битрикс: Управление сайтом** – профессиональная система управления Web – проектами. Этот универсальный программный продукт разработан для создания, поддержки и эффективного развития сайтов. Система ориентирована на корпоративные сайты, информационные и справочные порталы, социальные сети, Интернет - магазины, сайты СМИ, а также может быть пригодна для создания других видов Web– ресурсов [19].

Для сохранения данных сайта используется реляционная СУБД. Поддерживаются следующие СУБД: MySQL, Oracle, MS SQL. Продукт работает на Microsoft Windows i UNIX - подобных

платформах, включая Linux. **1С - Битрикс: Управление сайтом** реализован в виде семи типовых комплектов (*Старт, Стандарт, Эксперт, Малый бизнес, Бизнес, Портал, Большой бизнес*). Каждый из этих комплектов определяет свой набор модулей и функциональность конфигурированной системы.

Для корректной работы с системой управления пользователям без знания языков HTML и PHP необходима предварительная настройка системы (создание шаблона на основе графического дизайна, создания структуры разделов и страниц, а также подключения модулей системы).

Идеологией конфигулятора является разделение логики функций на модули и компоненты. Модули – это набор программных компонент, которые отвечают за работу с разными типами баз данных, а также предоставляют унифицированную систему API (ApplicationProgrammingInterface – интерфейс прикладного программирования – набор готовых классов, процедур, функций, структур и констант, которые предоставляются приложением, библиотекой, сервисом для использования во внешних программных продуктах). Компоненты служат для связи конечного представления информации на сайте с программным ядром системы. Они используют API, созданный модулями, для организации выборки, модификации, управления информацией в базе данных.

**Система SAP R/3** состоит из набора прикладных модулей, которые поддерживают разные бизнес-процессы компании и интегрированы в единый комплекс, функционирующий в масштабе реального времени. Такой комплекс включает: *Финансы (FI), Контроллинг (CO), Управление основными средствами (AM), Управление проектами (PS), Производственное планирование (PP), Управление материальными потоками (MM), Сбыт (SD), Управление качеством (QM), Техобслуживание и ремонт оборудования (PM), Управление персоналом (HR), Управление информационными потоками (WF), Отраслевые решения (IS), Базисная система* [22].

Даже самый краткий обзор функций системы SAP R/3 показывает ее способность решать основные задачи, которые стоят перед крупными организациями. SAP R/3 – одна из немногих систем, представляющая собой полноценный конфигуратор, имеющий в своем составе конфигурирующую систему и обеспечивающий аналитическую систему модельной конфигурации, а также создающий конфигурированную систему. Предприятие работает с индивидуальной сконфигурированной версией системы, настроенной именно под его параметры. Показателем технического уровня

конфигуратора может служить способ его настройки. Чем шире возможности конфигуратора и настройки системы без необходимости ее переписывания, тем выше уровень данной системы. С помощью специализированного инструмента бизнес-инжиниринга (BusinessEngineer) можно конфигурировать и настраивать систему SAP R/3 так, чтобы она удовлетворяла потребностям предприятия, поддерживать это соответствие на протяжении всего жизненного цикла системы. Бизнес-конфигуратор SAP R/3 поддерживает процедуры создания и ведения аналитических моделей конфигурации предприятия с автоматической генерацией соответствующих заданий и профилей настройки.

**BAAN IV** – система для разработки решений в управлении предприятиями с высокотехнологическим производством и корпоративной логистикой. Основные модули ERP-системы BAAN IV:

- *Моделирование предприятия:* в основе подсистемы лежат уникальные средства методологии внедрения Orgware, разработанной с учетом опыта внедрения продуктов BAAN более чем в 50-ти странах мира. Процесс внедрения начинается с описания или с рассмотрения соответствия типа и профиля предприятия типу и профилю референтной модели, выполняющей роль аналитической модельной системы конфигурации. На следующей стадии выполняется корректирование параметров бизнес-модели с учетом требований заказчика. Фактически здесь осуществляется уточнение модельной конфигурации. Далее система конфигурируется и для каждого конкретного пользователя создается меню, в структуру которого могут быть включены инструкции и нормативные документы, которые определяют выполнение отдельных заданий. В завершение выполняется анализ деятельности предприятия, на основе которого формируется конфигурированное решение по модернизации производства и определяются направления дальнейшего развития [23]. Фактически эта подсистема содержит средства конфигурирующей системы и средства аналитической модельной конфигурации.

- *Производство:* включает планирование потребностей, конфигуратор продукции, управление проектом, управление серийным производством и производством по отдельным заказам, управление цепочкой поставок на уровне корпоративного производства.

Также разработаны *BAAN - Процесс, BAAN - Финансы, BAAN - Сбыт, Снабжение, Склады; BAAN - Проект, BAAN - Администратор*

*деятельности предприятия, BAAN - Транспорт і BAAN - Сервис.*

#### **Конфигуратор системных блоков**

Он предназначен для упрощения процесса подбора комплектующих для сборного изделия. С его помощью можно легко и быстро рассчитать итоговую стоимость системного блока из комплектующих, которые есть на складе компании. Полученную модельную конфигурацию можно как сохранить (при этом конфигурация сборного изделия получает свой индивидуальный код, по которому дальше можно к ней вернуться), так и отправить заказ на ее сборку [18].

Предусмотрено три режима работы конфигуратора:

*Для начинающих пользователей* предлагается максимальный уровень проверки совместности компонент. Проверяется совместность компонент по разьему (для процессоров, видеокарт, жестких дисков и т.п.) и по цвету (например, корпуса и привода DVD). Допускается установка только одного вида комплектующих за исключением модулей памяти. В этом режиме можно собрать только простейшие конфигурации. Однако они обеспечивают наиболее быструю и надежную проверку совместности комплектующих в сборном изделии.

*Для продвинутых пользователей* предлагается средний уровень проверки совместности компонент.

*Для экспертов.* Этот режим рекомендуется использовать опытным пользователям, поскольку проверка на совместность в нем отключены. Также этот режим можно использовать для заказа конфигураций с частично отсутствующими на складе компонентами.

**Shorewall** или точнее **ShorelineFirewall** — инструмент для настройки межсетевого экрана в Linux, программное обеспечение под свободной лицензией GNU GPL [20]. Технически этот инструмент является надстройкой над подсистемой Netfilter (iptables/ipchains) ядра Linux и обеспечивает упрощенные методы конфигурации данной подсистемы. Программа не работает постоянно. Правила сохраняются в текстовых файлах. При запуске Shorewall считывает свои файлы конфигурации и преобразует их в настройки, понятные ipchains/iptables. После этого данные настройки могут действовать до перезапуска операционной системы. Shorewall не предусматривает GUI для конфигурации, правка конфигурационных файлов может выполняться в любом текстовом редакторе, однако есть, например, модуль к системе Webmin для настройки через Web-интерфейс.

Рассмотренные примеры показывают, что определению конфигуратора могут соответствовать практически любые системы ИТ, так как везде можно выделить три составляющие конфигуратора, пусть даже в некотором вырожденном виде. Действительно, четкой границы не существует. Однако, при создании интеллектуальных сложных развивающихся ИТ бизнес-систем целесообразно считать, что конфигуратором может быть только система ИТ с наличием явно выраженных и функционально выделенных трех составляющих: *конфигурирующей системы, аналитической системы модельной конфигурации, конфигурированной системы*. Как правило, такие системы и получают название от своих авторов, связанное с понятием «конфигуратор», например «Конфигуратор - Битрикс».

**Типы ИТ-конфигураторов.** По уровню сложности конфигураторы можно условно разделить на ряд типов:

- **Простые конфигураторы.** Они не меняют структуры и модули программного обеспечения, а меняют только настройки интерфейса, параметры выполнения конкретных функций. Сохраняются на клиентской стороне программного приложения. Этот тип конфигураторов чаще всего встречается на практике и является наиболее простым в реализации. Как правило, параметры сохраняются в \*.ini - файлах или в системном реестре. Без подобного уровня конфигурации приложение не может считаться полноценным. Любой коммерческий программный продукт содержит минимальный объем возможностей для настройки. В качестве примеров таких параметров конфигурации можно привести: «размер/стиль окон интерфейса», «тип и размер шрифта», «наличие/отсутствие панелей и строк состояния» и др.

- **Усложненные конфигураторы.** Они могут видоизменять структуру и модули программного обеспечения. Сохраняются на клиентской стороне. Встречаются в крупных многофункциональных программных продуктах, например в графических пакетах (3DMax, AutoCad). Из-за наличия большого числа функций возникает необходимость в их структуризации и настройке для конкретного пользователя.

- **Распределенные конфигураторы.** Они имеют те же свойства, что рассмотренный выше тип, однако данные конфигурации могут сохраняться на удаленном сервере и загружаться для конкретного пользователя на его локальный компьютер.

- **Конфигураторы с возможностью загрузки компонент.** До этого рассматривались конфигураторы, которые использовали только явные компоненты. Этот тип конфигураторов, в

свою очередь, позволяет по мере необходимости расширять составляющие блоки системы, которая конфигурируется. «Подгрузка» может выполняться с любого возможного источника – локально, по локальной сети от сервера, или даже из Интернет.

**Подходы к разработке конфигуратора.** Опираясь на сделанные выше обоснования и выводы, можно выделить два основных подхода к созданию конфигуратора. Здесь имеется в виду масштабный конфигуратор, удовлетворяющий требованиям современных развивающихся интеллектуальных ИТ для сложных бизнес-систем.

- **Первый подход** исходит из условия «Все блоки рассматриваются как единое целое». Этот подход предполагает распределенный тип конфигуратора. ИТ содержит полный набор всех компонент, которые в нем могут использоваться на всех этапах жизненного цикла ИТ при смене ситуаций, определяющих состав, структуру и организацию КИС бизнес-системы. Инсталляция на компьютеры пользователя заключается в копировании и установке (как через сетевой протокол, так и локально) всех необходимых компонент аналитической системы модельной конфигурации, конфигурированной системы, а также самой конфигурирующей системы. К преимуществам этого подхода можно отнести: максимальную скорость конфигурации – переконфигурация может быть проведена в любое время; максимальную скорость работы – все необходимые компоненты всегда находятся на компьютерах пользователя. К недостаткам этого подхода относятся: необходимость наличия большого (или очень большого) дискового пространства пользователя; длительный процесс инсталляции требует оперативного решения возникающих сложностей; невозможность обновления (или быстрого обновления) компонент.

На рис. 2 представлена схема реализации первого подхода в виде распределения и размещения составляющих конфигуратора на рабочих ПК бизнес-системы.

- **Второй подход** к разработке конфигуратора определяется условием «По требованию». Этот подход имеет в виду выполнение конфигурации с возможностью подгрузки компонент. В данном случае ИТ бизнес-системы включает только ядро конфигурированной системы и конфигурирующую систему. Конфигурация, например с использованием средств Data Mining, выполняется, как правило, при установке, поскольку во время работы она требует загрузки компонент.

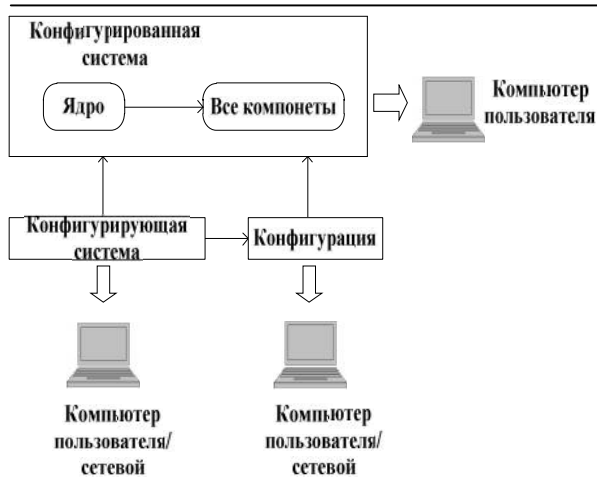


Рис.2. Принципиальная схема реализации первого подхода к разработке конфигуратора

К преимуществам этого подхода относятся: минимальный размер ИТ и отсутствие жестких требований к свободному дисковому пространству; быстрый процесс инсталляции базовой части ИТ; возможность обновления компонент. К недостаткам этого подхода можно отнести: любая переконфигурация требует подгрузки новых компонент, что может потребовать участия специалистов от разработчика. Схема реализации второго подхода представлена на рис.3.

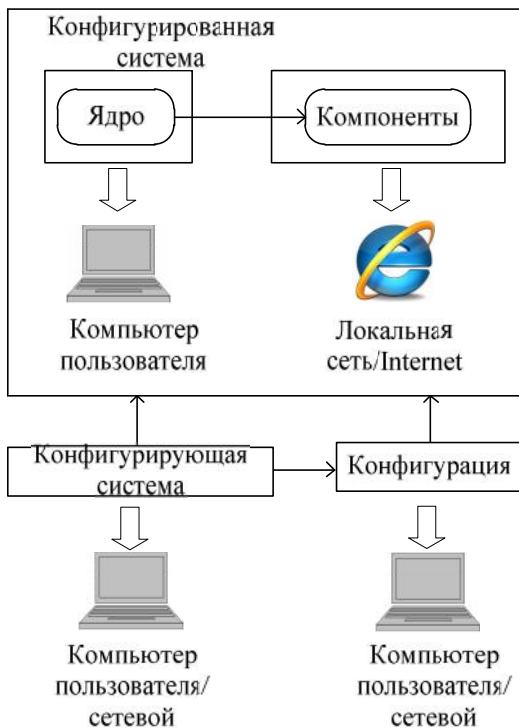


Рис.3. Принципиальная схема реализации второго подхода к разработке конфигуратора

**Выбор наиболее эффективной модели реализации конфигуратора.** Учитывая требования к созданию, эксплуатации и развитию бизнес-систем, рассмотренные выше, был выбран второй подход к созданию конфигуратора, поскольку для таких сложных меняющихся систем самым важным критерием является возможность обновления компонент.

На рис. 4 представлена схема взаимодействия компонент предлагаемого конфигуратора. В схеме в составе конфигурированной системы выделено место подсистеме «диспетчер», отвечающей за подгрузку компонент и за их использование.

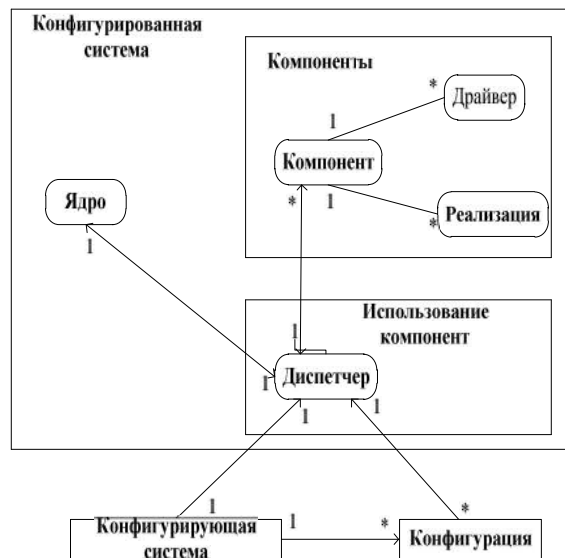


Рис. 4. Принципиальная схема предлагаемой структуры конфигуратора

Построенные на основе такой структуры конфигуратора современные комплексные КИС бизнес-систем, должны содержать три части:

- система, поддающаяся конфигурации (конфигурированная система), та, которая обеспечивает эффективное оперативное функционирование бизнес-системы в стабильных условиях и в стандартных ситуациях;
- система, определяющая конфигурацию, т.е. та, которая определяет модельные требования к конфигурации на ранних этапах проектирования КИС и на этапах ее развития (ее можно назвать средством системной аналитической модельной конфигурации бизнес-системы);
- система, выполняющая конфигурирование (конфигурирующая система), т.е. та, которая обеспечивает интеграцию разнофункциональных информационных технологий и поддерживает инструментальные средства реструктуризации как конфигурируемых систем и средств системной модельной конфигурации.



При реализации концепции «конфигураторов» меняются требования к архитектуре современных ИТ бизнес-систем на всех стадиях их жизненного цикла.

Общая архитектура КИС бизнес-системы, разработанной и функционирующей на основе концепции «конфигураторов» может иметь вид, представленный на рис.5.

Данная архитектура КИС бизнес-системы поддерживает три параллельных сферы деятельности, которые взаимодействуют между собой на уровне принятия стратегических решений на этапах разработки, эксплуатации и развития. К ним относятся:

- сфера конфигурирования КИС;
- сфера аналитической модельной конфигурации КИС;
- сфера эксплуатации конфигурированной КИС.



Рис. 5. Обобщенная схема архитектуры КИС бизнес-системы, основанная на концепции конфигураторов

Любая бизнес-система, во-первых, функционирует в изменяющейся внешней среде, во-вторых, ее внутренняя функционально-структурная организация также подвергается изменениям вследствие различных внутренних факторов. Все это требует периодической реструктуризации и реорганизации бизнес-системы. Поэтому компьютерная информационная система, которая поддерживает ее деятельность в таких меняющихся условиях, должна также периодически подвергаться реструктуризации. Такая реструктуризация требует проведения предварительного анализа состояния дел и ситуаций, переоценки целей и ограничений, оценки

функциональных возможностей, разработки новых элементов информационного и программного обеспечения, привлечения новых эффективных информационных технологий и т.п. Таким образом, появляется необходимость развивать аналитические блоки – КИС и блоки, осуществляющие синтез информационного и программного обеспечения. Так, появляется потребность в трех параллельных сферах функционирования КИС: в сфере конфигурирования КИС, в сфере модельной конфигурации КИС и в сфере эксплуатации конфигурированной КИС.

Сфера модельной конфигурации КИС с привлечением специальных средств, включая средства Data Mining, занимается аналитическим моделированием деятельности бизнес-системы, т.е. анализом ее целей, ограничений, функций и ситуаций, а сфера конфигурирования на основе полученных модельных результатов занимается построением (синтезом моделей) КИС, на основе проанализированной функционально-структурной организации. Фактически речь идет о дальнейшем развитии методологии итерационного процесса анализа и проектирования бизнес-системы и ее КИС на всех этапах их жизненного цикла.

Включение средств системной аналитической модельной конфигурации в состав КИС бизнес-систем позволит генерировать не только различные варианты комплекса программных модулей конфигурированной системы, но и альтернативные информационные целевые, функциональные и ситуационные модели отдельных бизнес-процессов и бизнес-системы в целом.

В настоящее время полученные результаты данного исследования проверяются в процессе разработки экспериментального конфигуратора информационных технологий в строительстве (КИТ – С) [8], задачей которого является конфигурирование ИТ на всех этапах ЖЦ создания объекта строительства, и прежде всего – на этапах подготовки строительства и управления строительным производством.

## Выводы

Изложенный материал позволяет сделать следующие выводы:

- проанализированы пути развития технологий создания современных КИС сложных бизнес-систем;
- обоснован и предложен новый перспективный подход к развитию концепции конфигураторов эффективных КИС для развивающихся бизнес-систем на основе предложенной принципиальной схемы конфигуратора;

- предложена обобщенная схема архитектуры современных КИС бизнес-систем на основе структуры конфигуратора с тремя параллельно функционирующими сферами деятельности (конфигурирующей системы, аналитической системы модельной конфигурации и конфигурированной системы);
- предложено использование и развитие средств Data Mining с целью их использования для решения задач в сфере аналитической системы модельной конфигурации бизнес-процессов;
- выявлена необходимость генерирования в сфере аналитической системы модельной конфигурации не только различных вариантов комплекса программных модулей конфигурированной системы, но и альтернативных информационных целевых, функциональных и ситуационных моделей бизнес-системы в целом.
- Отмечена необходимость экспериментальной проверки результатов исследования существующих конфигураторов и предложенной концепции их развития при разработке конфигуратора информационных технологий в строительстве на разных этапах жизненного цикла строительных объектов.

### Список литературы

1. Буч Г., Рамбо Дж, Джекобсон А. UML Руководство пользователя / -ДМК, Москва 2000 ISBN 5-93700-009-9
2. Вигерс Карл. Разработка требований к программному обеспечению/пер.с англ. – М.:Издательско-торговый дом «Русская редакция», 2004. – 576 с.
3. Гайсарян С.С. Объектно-ориентированные технологии проектирования прикладных программных систем/-CIT-forum <http://sure.org.ru/docs/c/oop/indexhtml>
4. Дюк В.А. 24. Data Mining – интеллектуальный анализ данных. – Санкт-Петербургский институт информатики и автоматизации РАН, 2002.
5. Задоров В.Б. системний аналіз об'єктів і процесів: технологічні основи: Навчальний посібник. – К.:КНУБА, 2003. – 276 с.
6. Задоров В.Б., Київська К.І. Концепція створення сучасних інформаційних систем маркетингу підприємств. Східно-Європейський журнал передових технологій, №3/2(39), 2009. – 18-25с.
7. Задоров В.Б. Про один підхід до створення технології попереднього системного проектування КІС підприємств // Управління розвитком складних систем. – 2010. – Вип. 01. – С. 56 – 83.
8. Задоров В.Б., Васильев О.О. Інтеграція інформаційних моделей в проектуванні та управлінні будівництва на основі узагальненого формату знань та даних // Управління розвитком складних систем. – 2011. – Вип. 05. – С. 52 – 60.
9. Калянов Г.Н., CASE – технологии. Консалтинг в автоматизации бизнес-процессов. – 3-е изд. – М.: Горячая линия –Телеком, 2002. – 320 с.
10. Леффингуэлл Дин, Уидриг Дон. Принципы работы с требованиями к программному обеспечению. Унифицированный подход.: Пер. с англ. – М.: Издательский дом "Вильямс", 2002. – 448 с.
11. Лычкина Н.Н. "Интегрированный многофункциональный комплекс имитационных моделей для стратегического и тактического управления деятельностью предприятий", – М.: МГУ, 2010
12. Прозоров А.А. Глоссарий предметной области "проектирование и разработка КИС"/- ASWL.RU <http://consult.aswl.ru/glossary/>
13. Филлип Кратчен Введение в Rational Unified Process. Пер. с англ. Второе издание. – М.: Издательский дом "Вильямс", 2002. – 235 с.
14. Хассан Гома UML Проектирование систем реального времени, распределенных и параллельных приложений/ - ДМК Пресс, 2002. -704с.: ил.
15. Шебеко Ю. Персональный компьютер помогает исследовать и анализировать поведение бизнес-процессов Руководство пользователя аналитического пакета Ithink (Прикладное пособие для менеджеров) Москва-1999г. <http://www.tora-centre.ru/library/reing/ta.htm>
16. Getting Started With Ithink : A Hands-On Experience. High Performance Systems, Inc. 1995. - 42 pp.
17. Introduction to Systems Thinking and Ithink. High Performance Systems, Inc. 1994. - 178 pp.
18. Business Applications Guide. High Performance Systems, Inc. 1994. - 170 pp.
19. Autodesk: <http://www.autodesk.ru/adsk/servlet/pc/ite/m?siteID=871736&id=15621143>.
20. <http://ru.wikipedia.org/wiki/Shorewall>.
21. <http://www.1c-bitrix.ru/products/cms>.
22. <http://www.erp-online.ru/sap>.
23. <http://www.erp-online.ru/software/baan>.

Стаття постуила в редколлегию 08.07.2011

**Рецензент:** д-р техн. наук, проф. С.В. Цюцюра, Киевский национальный университет строительства и архитектуры, Киев