

ни являются специалистами по проектированию и оптимизации документооборота и работают с объектной моделью. С другой стороны, им не надо быть специалистами по бизнес-процессам. На последнем этапе подключаются к работе люди из группы реализации. На этом этапе можно не только реализовать систему документооборота с нуля, но и настроить какую-либо существующую, так как все настройки и документы уже известны и описаны. В третьей группе могут быть просто специалисты по внедрению какой-то готовой (но гибко настраиваемой) системы. Благодаря предлагаемому методу постепенного перехода от процессов к документам и их схемам работы, концепция «что должно делать предприятие» в итоге перейдет в конкретные настройки, а на каждом этапе будут работать специалисты в своей области.

Использование нескольких моделей позволяет посмотреть на систему с разных точек зрения и получить одновременно разные варианты одних и тех же требований. Таким образом, на некоторых этапах появляется альтернатива, являющаяся сигналом к проведению дополнительного анализа и выбору более подходящего решения. Множество моделей позволяет выявить больше противоречий в требованиях на более ранних этапах проектирования.

Есть и недостаток рассматриваемого подхода: надо вести параллельно фактически два процесса моделирования. Но большее число моделей дает

лучшее понимание сути системы разным группам разработчиков.

Авторами было разработано и реализовано программное средство, упрощающее применение описанных методов и позволяющее частично их автоматизировать. Сами методы были успешно применены при проектировании системы документооборота для одной из компаний, специализирующихся на разработке и верификации встроенного ПО. На основе примерно 40 процессов и порядка тысячи предметов были выделены основные типы документов и построены их ЖЦ. Эксплуатация электронной системы документооборота, в основу которой были положены выделенные типы документов, показала применимость описываемых методов и алгоритмов. Кроме того, внедрение системы документооборота позволило успешно пройти сертификацию по стандартам ISO 9001:2000 и AS 9100A.

Список литературы

1. Фаулер М., Скотт К. UML основы. Краткое руководство по унифицированному языку моделирования. – М.: Символ-Плюс, 2002. – 192 с.
2. Марка Д.А., МакГоуэн К. Методология структурного системного анализа и проектирования SADT. – М.: Метатехнология, 1993. – 240 с.
3. ISO 9001:2000 Quality Management System – Requirements, ISO, 2000. – 33 p.
4. Синицын С.В., Хлытчиев О.И. Переход от процессного к объектному описанию системы документооборота предприятия. / В сб.: Современные технологии в задачах управления автоматикой и обработки информации: Тр. XVI Междунар. науч.-технич. сем. – Тула: Изд-во ТулГУ. – 2007. – С. 80–81.

КОМПОНЕНТНЫЙ ПОДХОД: МОДУЛЬ ПРАВДОПОДОБНОГО ВЫВОДА ПО ПРЕЦЕДЕНТАМ

(Работа выполнена при поддержке Фонда содействия отечественной науке)

А.И. Павлов, к.т.н.; А.Ю. Юрин, к.т.н.

(Институт динамики систем и теории управления СО РАН, г. Иркутск)

Компьютерная поддержка исследователя в слабоформализованных предметных областях, связанных с процессами, представляющими опасность для человека (например производство полиэтилена в нефтехимии), является актуальной проблемой [1]. Ее актуальность главным образом обусловлена необходимостью обеспечения безопасности людей и повышением надежности оборудования, задействованных в опасных процессах.

Наиболее перспективным в данной ситуации является создание программного средства, автоматизирующего исследование динамики состояний данных технических (механических) производственных систем. Создаваемое средство автоматизации исследований должно обладать высокой универсальностью (гибкостью), то есть обеспечивать возможность решения широкого круга задач исследования (идентификация, диагностирование и прогнозирование) и

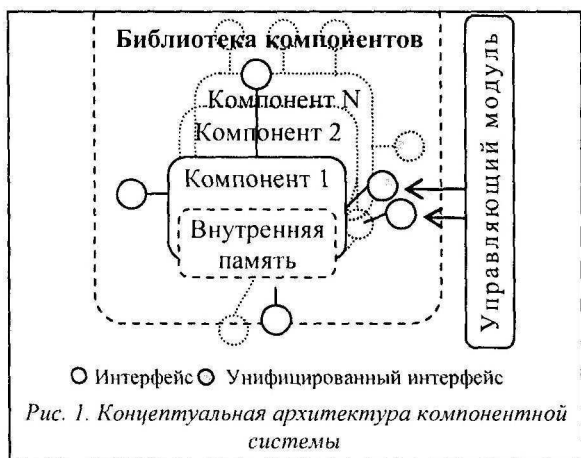
настройки на определенную предметную область [2]. Наиболее эффективным способом создания подобных программных систем является компонентная сборка требуемой системы из функциональных компонентов.

Технология сборки систем из компонентов – один из подходов объектно-ориентированного программирования, центральным понятием данной технологии является компонент – независимый модуль, который может быть реализован в виде исполняемого файла или динамической библиотеки.

Каждый компонент реализует как минимум один интерфейс, определяющий границу взаимодействия одного компонента с другими или с программами. Интерфейс строго специфицирует поведение компонента, благодаря чему компонент может быть использован на основании описания интерфейса, без внимания к подробностям его реализации.

В данной работе под компонентом системы автоматизации исследований понимается компонент, который, кроме основной функциональности, обеспечивает реализацию механизма внутренней памяти и предложенного авторами унифицированного интерфейса компонента [3].

Концептуально в архитектуре подобной системы (рис. 1) можно выделить библиотеку компонентов, содержащую компоненты с унифицированными интерфейсами, и управляющий модуль, содержащий механизм интеграции компонентов в единую систему. Использование унифицированного интерфейса позволяет перестраивать архитектуру системы исследования динамически, без внесения в код компонентов информации о межкомпонентных связях.



Таким образом, функциональность системы автоматизации исследований напрямую зависит от компонентов, составляющих ее библиотеку.

Среди множества методов, которые могут быть алгоритмизированы и программно реализованы в виде компонентов для системы автоматизации исследований, выделим метод (подход), основанный на изучении существующего в определенной предметной области опыта – это правдоподобный вывод по прецедентам (поиск решения по аналогии, *case-based reasoning*) [4]. Выбор данного метода обусловлен тем, что зачастую в областях, где требуется поддержка исследователя (например, обеспечение надежности и безопасности сложных технических систем в нефтехимии), к моменту возникновения *новой* проблемы уже накоплен значительный опыт решения *похожих* проблем, возникавших ранее на подобном оборудовании. Однако невозможность или сложность аналитической обработки этого опыта по ряду причин (например, отсутствие специалистов необходимой квалификации, экспертов и др.) приводит к невозможности эффективно использовать эти знания и реализовать их потенциал. Представление же этого опыта в виде прецедентов и его автоматизированная обработка при помощи специализированных программных систем позволяют значительно повысить эффективность его повторного использования.

Решение проблемы по прецедентам основано на распознавании текущей проблемной ситуации, информация о которой представлена в виде некоего

образа (прецедента), и поиске похожих образов, содержащихся в хранилище образов (базе прецедентов), с последующей их адаптацией и повторным использованием для решения задач исследования.

Таким образом, можно выделить основные функции, которыми должен обладать компонент, обеспечивающий решение задач с помощью правдоподобного вывода по прецедентам.

1. Представление прецедента (формирование модели прецедента). Структура прецедента (образа) может быть произвольной, но четко выделяются две части: описание проблемы и ее решение.

2. Формирование базы (библиотеки) прецедентов на основании сформированной модели. Предполагается, что база прецедентов формируется с использованием существующего хранилища информации предметной области, например, базы данных отказов оборудования.

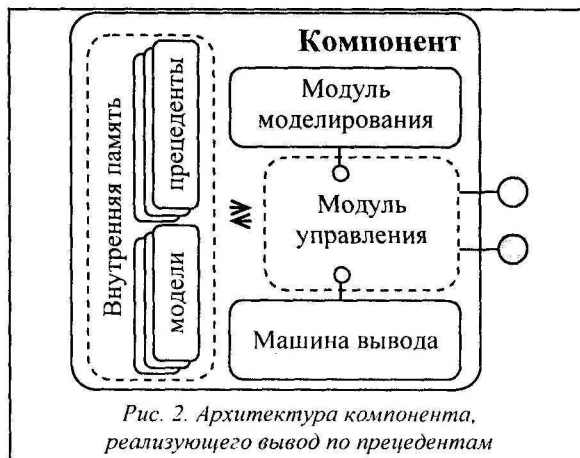
3. Поиск решения (правдоподобный вывод) по прецедентам. При поиске руководствуются полезностью того или иного образа (прецедента) для решения новой проблемы. Полезность аппроксимируется мерой схожести описаний образов, вычисляемой как расстояние между образами в многомерном признаковом пространстве.

Компонент, реализующий описанную функциональность, содержит следующие основные модули (рис. 2):

- модуль моделирования прецедентов – обеспечивает возможность создания, модификации моделей прецедентов, формирования и обновления баз прецедентов на основе существующих моделей;
- внутренняя память – обеспечивает хранение моделей прецедентов и самих прецедентов;
- прецедентная машина вывода – позволяет осуществлять поиск прецедентов по полученному описанию;
- управляющий модуль – обеспечивает взаимодействие между модулями компонента и предоставляет интерфейсы для взаимодействия с внешней по отношению к компоненту средой.

Рассмотрим модули подробнее.

Модуль моделирования. Решение задач исследования на основе прецедентов требует создания моделей прецедента под эти задачи. В случае построения базы прецедентов на основании существующего хра-



нилища данных предметной области наличие подобной модели позволит понизить избыточность этой базы путем отсека (исключения из модели прецедента) неинформативных (нерелевантных решаемой задаче) признаков (данных).

Модель прецедента базируется на классической модели прецедента: *прецедент* = *<проблема, решение>* [4]. Части модели представляют собой коллекции описаний свойств хранимых классов – классов хранилища данных. В процессе формирования модели прецедента пользователь посредством графического интерфейса, отображающего информацию о структуре хранилища данных, указывает свойства, составляющие описания проблемы и решения. При этом предполагается существование некоего хранилища данных со структурой, элементы которой можно использовать как основу для формирования прецедентов. Если подобное хранилище отсутствует и все знания эксперта содержатся в неявном виде, структуру хранилища предлагается создать и заполнить данными перед формированием модели. В этом случае структура модели будет соответствовать структуре хранилища данных.

Внутренняя память, или память компонента. Для хранения сформированных моделей предлагается использовать хранилище данных, при этом создается хранимый класс (структура), который позволяет сохранять информацию, идентифицирующую модель (идентификатор модели), и данные модели, подвергнутые сериализации, в виде потока двоичной или символической информации.

Данная структура при ее простоте позволяет сохранять модели прецедента любой степени сложности. Перед использованием модели происходит десериализация – обратное преобразование хранимых данных.

Хранение прецедентов также осуществляется в памяти компонента. Возможна синхронизация (односторонняя, в сторону базы прецедентов) хранилища данных предметной информации и базы прецедентов или формирование последней на основании анализа хранилища данных в соответствии с актуальной моделью прецедента.

В процессе работы компонента база прецедентов размещается в оперативной памяти, что, хотя и приводит к определенной потере производительности в процессе загрузки базы прецедентов, однако обеспечивает высокое быстродействие в дальнейшем.

Существует возможность сохранения информации о новых прецедентах, полученных в процессе работы компонента. Это обеспечивает возможность обучения как прецедентного компонента в отдельности, так и системы исследования в целом: опыт решения новой проблемы становится доступным для повторного использования в будущем.

Машина вывода. На основании сформированной базы прецедентов возможно построение правдоподобного вывода путем анализа описаний прецедентов, представленных в виде наборов признаков с качественными и количественными значениями, формирования локальных и глобальных оценок близости описаний прецедентов и эталона с учетом субъек-

тивных предпочтений, задаваемых пользователем (оператором), с последующим выбором наиболее близкого прецедента.

Извлечение прецедентов осуществляется в соответствии с глобальной мерой (оценкой) подобия (близости) описаний прецедентов, вычисляемой как расстояние между прецедентами в признаковом пространстве.

При вычислении расстояния каждому прецеденту ставятся в соответствие векторы (дескрипторы) описания: бинарные (...01001...) или состоящие из кортежей {..., S_{v_i} , ...} типа $S_{v_i} = \langle n, z, v, o \rangle$, где n – наименование свойства; z – его значение; v – важность или информационный вес свойства; o – ограничение на интервал допустимых значений). Ограничение определяет интервал значений, в рамках которого значение свойства может определять значение меры подобия, при выпадении значения свойства за этот интервал значение меры подобия равно 0.

Выбор вида вектора происходит в зависимости от сложности описания прецедента: типов признаков (детерминированных или логических) и наличия иерархии свойств. Бинарные векторы используются в том случае, если необходимо проанализировать отсутствие или присутствие определенных признаков в описании прецедентов, а кортежи – если признаки имеют иерархическую структуру (например, иерархия типа: *процесс – параметр процесса* с числовым значением) или важны числовые (количественные) значения этих признаков. Вычисление расстояния между прецедентами осуществляется при помощи данных векторов и метрики Миньковского [5]:

$$\text{dist}_{\text{Minkowski-p}}(\bar{x}, \bar{y}) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}.$$

Данная метрика является обобщением метрик городских кварталов (используется при обработке бинарных векторов, при этом $p=1$) и евклидова расстояния (используется при обработке множеств кортежей, при этом $p=2$).

Результатом работы машины вывода является список прецедентов с оценками близости.

Управляющий модуль обеспечивает взаимодействие между модулем моделирования, машиной вывода и внутренней памятью, публикацию результатов работы машины вывода с возможностью выбора, назначения и модификации (адаптации) решения и сохранение пересмотренного прецедента в базе прецедентов. Основная его функция – обеспечение предоставляемых компонентом интерфейсов, в том числе и унифицированного.

Сложность и многообразие задач в различных областях, связанных с опасными для человека и окружающей среды процессами, обуславливает необходимость проектирования и разработки специализированных интеллектуальных прикладных систем на основе изучения существующего опыта. Перспективность данного направления подтверждается примерами успешного решения задач планирования, диагностики, проектирования и подбора персонала в таких областях, как электронная коммерция и синтез программного обеспечения, авиация, атомная энергетика и др.

Однако в большинстве случаев для решения подобных задач в конкретной предметной области создается отдельная программная система, что является неэффективным и затратным.

Компонентный подход позволяет значительно повысить эффективность процесса разработки программных систем путем создания расширяемого и настраиваемого компонентного программного обеспечения. Система автоматизации исследований, разрабатываемая в соответствии с этим и содержащая описанный прецедентный компонент, позволит эффективно решать задачи планирования, диагностики, проектирования.

Компонент, реализующий прецедентный подход, использован при решении задачи идентификации технического состояния уникальных машин и конструкций в интеллектуальной системе поддержки принятия решений при определении причин отказов и аварий в нефтехимической промышленности [6].

Список литературы

1. Берман А.Ф. Деградация механических систем. – Новосибирск: Наука, 1998.
2. Берман А.Ф., Николайчук О.А., Павлов А.И., Юрин А.Ю. Моделирование свойств адаптивности программной системы. // Матер. Всерос. конф.: Инфокоммуникационные и вычислительные технологии и системы. – Улан-Удэ, 2003.
3. Павлов А.И. Подход к автоматизации исследования надежности механических систем. // Матер. V Междунар. науч.-технич. конф.: Информационно-вычислительные технологии и их приложения. – Пенза, 2007.
4. Aamodt A., Plaza E. Case-Based reasoning: Foundational issues, methodological variations, and system approaches. AI Communications, vol. 7, no. 1, 1994.
5. Bergmann R. Experience Management. Lecture notes on artificial intelligence, vol. 2432, 2002.
6. Берман А.Ф., Николайчук О.А., Павлов А.И., Юрин А.Ю. Интеллектуальная система поддержки принятия решений при определении причин отказов и аварий в нефтехимической промышленности. // Автоматизация в промышленности. – 2006. – № 6.

ПРОГРАММА С ЭЛЕМЕНТАМИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ОЦЕНКИ РЕАКЦИОННОЙ СПОСОБНОСТИ РАДИКАЛЬНЫХ РЕАКЦИЙ

(Работа выполнена при поддержке гранта РФФИ 07-07-00343а)

В.Е. Туманов, к.х.н. (Институт проблем химической физики РАН, г. Черноголовка)

В результате научных исследований накоплены огромные массивы научных данных о характеристиках химических реакций (константах скорости, энергиях активации, энтальпиях реакции) и о свойствах реагентов (энтальпиях образования молекул и радикалов, потенциалах ионизации и т.д.). Эти данные размещаются в электронных коллекциях данных и часто публикуются в Интернете. Несмотря на доступность данных о реакционной способности реагентов в химических реакциях, при конструировании технологических процессов и моделировании явлений в экологии и медицине химии принимают решения о скорости протекания реакций по аналогии.

Интеллектуальный анализ накопленных электронных данных о характеристиках химических реакций позволяет во многих случаях находить эмпирические зависимости и строить на их основе классификацию реакций, которую можно использовать для предсказания скорости протекания химических реакций. Это даст возможность в ряде случаев получить надежные оценки констант скорости и энергий активации химических реакций.

Константа скорости радикальной элементарной реакции характеризует скорость взаимодействия реагентов (радикала и молекулы) в жидкой или газовой среде. Как правило, константа скорости реакции связана с экспериментальной энергией активации E и предэкспоненциальным множителем A_0 в расчете на одну эквивалентную связь в молекуле соотношением Аррениуса [1]:

$$k = n \times A_0 \exp(-E/RT), \quad (1)$$

где n – число эквивалентных связей в молекуле-реагенте; T – температура в К; R – универсальная газовая постоянная.

Экспериментальное определение констант скорости сложной экспериментальной задачей, и во многих практических случаях ее теоретическая оценка является актуальной.

Эмпирическая классификация радикальных реакций. В начале 90-х годов прошлого века профессор Е.Т. Денисов [2] эмпирически установил наличие параболического тренда между корнем квадратным из классического потенциального барьера радикальной реакции ($\sqrt{E_e}$) и ее энтальпией (ΔH_e):

$$\Delta H_e = a(\sqrt{E_e})^2 + b\sqrt{E_e} + c, \quad (2)$$

где a , b , c – некоторые эмпирические коэффициенты.

Интерпретация тренда (2) была дана Е.Т. Денисовым в предложенной им модели элементарной бимолекулярной реакции – модели пересекающихся парабол (МПП) [2]. В рамках этой эмпирической модели реакция радикального отрыва рассматривается как результат пересечения двух потенциальных кривых, одна из которых описывает потенциальную энергию колебания атома H вдоль разрываемой связи в исходной молекуле, а другая – потенциальную энергию колебания того же атома вдоль образующейся связи в продукте