

МЕТОДЫ И СРЕДСТВА КОНТРОЛЯ СЕМАНТИЧЕСКОЙ ЦЕЛОСТНОСТИ ИНФОРМАЦИОННЫХ МОДЕЛЕЙ РЕАКТОРНЫХ УСТАНОВОК

Введение

В атомной промышленности для обеспечения процессов проектирования и построения реакторных установок (РУ) используется большое количество корпоративных информационных систем (КИС), сопровождающих конструкторские, технологические, экономические и тому подобные аспекты (т. е. CAD, PLM, ERP и прочие системы). Чаще всего одному аспекту соответствует своя КИС, обеспечивающая выполнение основных операций с объектами соответствующего ей типа (графические манипуляции с моделями и чертежами, расчетные операции для определения соответствия нужным характеристикам, финансовые вычисления и прочее). И хотя объекты из этих систем отличаются по структуре и содержанию, все они описывают единое устройство — РУ.

Подобное всестороннее описание называется информационной моделью изделия (ИМ), если изделие представлено в нем как набор связанных фактов [1]. При этом изменение одного факта в системе может породить изменения других фактов. Например, изменение мощности РУ может сказаться на ее себестоимости, а потеря уникальности какого-либо объекта может произойти вследствие изменения значения одного из его свойств.

Проблема порождающих изменений

В пределах одной КИС описание ИМ может быть разнесено в том числе на программный уровень и уровень БД. В таком случае за выполнение подобных порождаемых изменений отвечает как программная модель КИС, реализуемая в виде формул, правил и пр., так и реляционная модель со стороны СУБД. К примеру, некоторые системы трехмерного моделирования (Catia, Solid Works и другие) позволяют задавать и отслеживать требования к габаритным размерам модели (предупреждают о превышении). Контроль же выполнения подобных требований возможен только из программной среды системы. В случае изменения размеров изделия на уровне БД система никак не отреагирует, пока не будет запущен необходимый программный модуль.

Чаще всего ИМ, построенные с помощью КИС, в своей основе имеют прагматические модели, т. е. такие, которые описывают реакцию системы лишь на конкретные (перечислимые) действия. При выходе же за границы этих действий система не обеспечивает адекватной реакции даже при угрозе потери данных. Между тем в ИМ могут произойти неконтролируемые порождаемые изменения (НПИ), отследить которые с помощью работающей с ней КИС не представляется возможным. Они могут произойти как в результате злоумышленных действий (подмены, уничтожения информации, внедрения паразитных подсистем [2]), так и в результате некорректного добавления/удаления информации пользователем либо информационной системой. В случае возникновения НПИ существует вероятность потери семантической целостности ИМ, т. е. приобретения таких свойств, как противоречивость, избыточность, потеря целостности и достоверность.

Обеспечение взаимосвязи между отдельными КИС, как правило, выполняют системы интеграции (подобные BizTalk и пр.), которые в дальнейшем на основе порталных решений собирают разрозненную информацию в единую структуру. При этом если в отдельных КИС для описания объектов и связей между ними заложена ИМ, которая может позволять каким-то образом контролировать их семантику (противоречия, дублирование, достоверность информации), то в случае объединения нескольких КИС даже имеющаяся семантика легко может быть потеряна при недостаточно корректном слиянии. Таким образом, фиксация НПИ в интегрированной ИМ еще менее очевидна, нежели в отдельных КИС.



Выбор формализма представления ИМ

В качестве решения проблемы обнаружения НПИ предполагается использование некоего математического аппарата, позволяющего описывать так называемую динамику предметной области [2] и отслеживать порождаемые изменения.

Однако для работы с ИМ одного математического аппарата недостаточно, поскольку также необходима возможность хранения структуры модели как исходных данных для процесса вычисления. В связи с этим возникает необходимость в использовании формализма, совмещающего описанные выше требования для работы с ИМ.

Существует множество формализмов, базирующихся на различных математических аппаратах или их комбинациях. Однако некоторые из них позволяют описывать лишь выборочные аспекты предметной области.

Так, к примеру, формализм сетей Петри, позволяющий описывать динамику поведения, не дает возможности определять иерархические связи между объектами. Реляционная модель, напротив, описывая отношения между объектами, не учитывает динамику.

Основным критерием выбора формализма для представления предметной области, безусловно, является *способность описать ее целиком с учетом всех аспектов и особенностей*.

Для оценки масштаба предметной области следует упомянуть, что РУ включает сотни тысяч простых устройств, состоящих из нескольких деталей. В процессе разработки подобного простого устройства в КИС предприятия накапливается информация о тысячах фактов (характеристиках, требованиях, связях и тому подобных объектах).

Обработка такого числа объектов возможна только с использованием средств автоматизации. Поэтому далеко не последнюю роль играет такой критерий, как *удобство автоматизации вычислений математического аппарата*.

После регистрации НПИ необходимо представить результат в виде указания на конкретный опасный объект ИМ, присутствие которого приводит к появлению подобного изменения. Далее необходимо произвести анализ ИМ, на основании которого будут предприняты те или иные действия по ее восстановлению. Подобный анализ может проводиться посредством визуального отслеживания связей опасного объекта с другими объектами системы. В связи с этим критерий *наличия у формализма графической нотации* для удобного просмотра ИМ также является существенным.

Лишь немногие формализмы для описания ИМ удовлетворяют положенным критериям выбора.

Так, например, разные виды диаграмм IDEF-, UML- и ARIS-методологий имеют различные нотации и используются для моделирования лишь выборочных аспектов предметной области. Для полного описания ИМ в методологиях подобного рода потребовалось бы использование множества видов диаграмм.

Среди подходов, предположительно имеющих возможность описания данной предметной области, можно отметить проект «СИНТЕЗ» [6]. Однако отсутствие промышленного внедрения и специфичность языка описания, позволяющего работать с ним только специалистам в области программирования, не дает использовать его для описания ИМ.

Каждый из формализмов на основе семантических сетей, таких как фреймовая нотация Руссопулоса [5] и онтологии, позволяет описать ИМ, поскольку удовлетворяет названным выше критериям выбора.

Различие данных формализмов, главным образом, состоит в математических аппаратах, лежащих в их основе. Для онтологий характерными являются логика высших порядков и теория нечетких множеств, применяемая для обеспечения необходимой меры релевантности в слабоструктурированных предметных областях. Формализм фреймов также использует логику высших порядков, но в отличие



от онтологий основан на теории неопределенных элементов, позволяющей моделировать лишь жестко структурированные предметные области. При этом теория нечетких множеств является более сложным математическим аппаратом, чем неопределенные элементы.

Применение того или иного подхода к описанию, прежде всего, зависит от специфики конкретной предметной области.

В исследуемой предметной области присутствуют как статическая, так и динамическая составляющие. Ее особенностями являются высокая сложность классов объектов, а также неполная информация о структуре отдельных конкретизаций этих классов. При этом состав атрибутов самих классов и операций над ними строго детерминирован. В силу данных причин выбор методологии на основе фреймов является оправданным.

Объекты ИМ, описанные с помощью фреймовой методологии, могут быть представлены в знаковой и графической нотациях.

Знаковая нотация описывает предметную область в виде предикатных функций высших порядков, в качестве переменных которых могут присутствовать неозначенные переменные. Динамика ИМ в этом случае может проявляться как постепенная конкретизация таких переменных. Каждое порожденное изменение ИМ производит изменение предикатной функции, являющейся в данном случае его индикатором.

В качестве удобного математического аппарата для работы с подобным представлением целесообразно использовать α -исчисление, которое адекватно поддерживается в современных языках программирования (Java, F#).

Графическая нотация представляет собой граф, состоящий из помеченных узлов, дуг, других графов, дающий наглядное представление об имеющейся ИМ.

В работах [3–5] показаны варианты трансляции модели, построенной с использованием данного формализма, в реляционные отношения базы данных для последующего размещения в БД.

Средства описания ИМ

«Ручное» моделирование даже относительно простых изделий, состоящих из нескольких простейших составляющих (материалов), при помощи предложенной нотации представляется довольно сложной и трудоемкой задачей [1]. При описании подобного количества объектов на бумаге или без специализированного CASE-средства неизбежно возникнут ошибки. Попытки проследить связи, выстроить отношения и объединить объекты вполне могут завершиться безрезультатно.

Для упрощения работы с ИМ необходимо программное средство, позволяющее комфортно описывать объекты на фреймах и производить такие операции, как:

- навигация по всем существующим в модели связям;
- поиск любых элементов и значений;
- выполнение масштабирования;
- отображение объектов в выбранном контексте;
- указание семантических ошибок (противоречивости, избыточности) в модели;
- указание объектов, вызывающих НПИ.

Необходимо отметить, что построение модели предметной области — прерогатива предметного аналитика — человека, возможно, не являющегося специалистом в сфере информационных технологий. Поэтому моделирование с использованием фреймовой и иных подобных нотаций будет для него неприемлемым. Для решения данной проблемы требуется использовать предметно-ориентированные языки (которые принято относить к классу DSL-языков), разработанные под конкретные группы специалистов и однозначно перелагающиеся в выбранную нотацию и обратно.



Информация, поступающая во фреймовую модель, может браться из различных источников, таких как:

- знания предметных специалистов;
- программные средства (с использованием специальных адаптеров).

Таким образом, данное CASE-средство может быть использовано не только для контроля семантической целостности, но и как средство семантической интеграции.

В модели на фреймах существует возможность выделения логически структурированных частей фрейма с целью их дальнейшего использования в качестве основы для создания других конфигураций, что приводит к созданию паттернов проектирования.

Выводы

В работе предложен новый подход к контролю семантической целостности программных систем. В основу подхода положен синтез фреймовых моделей с предметно-ориентированными языками и CASE-средствами специального назначения. К новым возможностям, которые открывает подход, относятся математически точное описание информационных объектов программных систем, семантический контроль безопасного (согласованного) состояния этих объектов, а также инновационное масштабирование систем на основе паттернов проектирования индустриального уровня.

СПИСОК ЛИТЕРАТУРЫ:

1. Зубков И. А. Проектирование программных средств для разработки 6D-паттернов ядерных энергетических установок // Сборник трудов «Конференция молодых специалистов ОАО ОКБ «ГИБРОПРЕСС» 2010». С. 109.
2. Вольфенгаген В. Э. Семантические вирусы и провоцирующие подстановки // Сборник трудов 2-й международной конференции ABC 2010. Институт «ЮрИнфоР-МГУ». С. 100.
3. Вольфенгаген В. Э., Горюнова И. А., Косиков С. В. Методы и средства построения систем знания. Ч. 2. Модели, обслуживающие базы данных и базы знаний. Учебное пособие. М.: МИФИ, 1992. — 136 с.
4. Зыков С. В. Проектирование корпоративных порталов. М.: МФТИ, 2005. — 258 с.
5. Roussopoulos N. D. A semantic network model of data bases. Toronto Univ., 1976.
6. Калининченко Л. А. СИНТЕЗ: язык определения, проектирования и программирования интероперабельных сред неоднородных ресурсов. М., 1993.

