

мають різні граматичні рамки) будуть позначатися однаково, і, відповідно, мати ідентичний зміст.

4. Логічний вивід. На останньому етапі алгоритму порівняння логіко-лінгвістичних моделей після ідентифікації та заміни тотожних змінних застосовується система продукцій, що містить правила порівняння логіко-лінгвістичних моделей. Для цього використовується база знань у вигляді продукційної моделі, сформована на основі правил формування речень флективних мов. Такі правила дозволяють через встановлені зв'язки між словами переходити до представлення значень слів у вигляді комбінацій елементарних компонентів змісту.

Формалізація текстової інформації дає змогу знаходити компоненти змісту без звернення до референта та екстенціонала кожного зі слів [3], а порівняння формалізованих логічних моделей з використанням правил формування речень флективних мов – доводити істинність висловлювань в залежності від контексту певного текстового документу.

Література

1. Башмаков А.И., Башмаков И.А. Интеллектуальные информационные технологии. – М.: Издательство МГТУ им. Н.Э. Баумана, 2005. – 302с.
2. Звегинцев В.А. Новое в зарубежной лингвистике. Лингвистическая семантика / Звегинцев В.А. – М.: Прогресс, 1981. – 566 с.
3. Мельников Г.П. Системология и языковые аспекты кибернетики/ Под ред. Ю.Г. Косарева. – М.: Сов. Радио, 1978. – 368с.

УДК 004.8

ЯЗЫК МОДЕЛИРОВАНИЯ ПРОГРАММНЫХ АРХИТЕКТУР NOARE_SYSMOD. ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ, ПОСТРОЕНИЕ ДИАГРАММЫ ВЗАИМОДЕЙСТВИЙ

Е.Н. Гайтан

Полтавский национальный технический университет имени Юрия Кондратюка, Украина

Существенным этапом разработки информационных систем является проектирование программной архитектуры, которое часто нуждается в больших ресурсах и может быть ошибочным. Архитектурное проектирование идентифицирует ключевые стратегии, которые позволяют определить программные структуры, такие как подсистемы, пакеты и задачи.

Однако при построении программной архитектуры выделяют проблемы моделирования, проверки корректности и непротиворечивости построенной архитектуры, то есть ее верификации. Поэтому совершенствование методов и программных средств моделирования и верификации программной архитектуры является актуальным.

Существует большое количество разработок и исследований в области описания архитектур программных систем. Наиболее эффективным сред-

ством создания имитационных моделей являются специальные имитационные языки, языки описания интерфейсов (IDL) и языки описания архитектур (ADL). Однако большинство систем, которые используются при описании и моделировании программных архитектур, имеют недостатки, устранение которых возможно за счет использования математического аппарата, например CSP (Communicating sequential processes) [1], теории взаимодействующих последовательных процессов, разработанной Хоаром (Великобритания).

В данной работе рассматриваются принципы функционирования визуальной, объектно-ориентированной системы HOARE_SYSMOD, основанной на теории взаимодействующих процессов Ч. Хоара, для моделирования и верификации архитектуры программных систем.

1. Сборка системы.

а) сборка модели, используя графические компоненты, на экране.

Графическое построение модели начинается с контекстной диаграммы, отображающей контекст функционирования моделируемой системы как единого целого. Детализация системы осуществляется с помощью диаграмм декомпозиции, которые отражают разбиение системы на подсистемы или элементы.

Для каждого блока обязательно указывается количество входов и выходов, характеристическое действие в терминах Хоара. Возможно также определение этих характеристик блока из диаграмм декомпозиции.

Таким образом, в одной системе содержится один блок, внутри которого находятся все остальные блоки. Описание целой системы в терминах Хоара строится рекурсивно от самых мелких блоков до системы.

Рассмотрим пример модели простой системы, которая подсчитывает количество правильных и неправильных ответов. Пусть Check – компонент, определяющий правильность ответов сравнением с эталоном, Right и Wrong – компоненты-счетчики правильных/неправильных ответов соответственно. Компоненты производят вычисления. Каждый компонент имеет входы и выходы. Например, компонент Check имеет один вход In и два выхода – Out1 и Out2. Если сигнал является входным для данного элемента, он обозначается с помощью вопросительного знака, например In?x. В противном случае сигнал обозначается с помощью восклицательного знака Out!x. Тогда компонент Check описывается как

Check:

// Входные и выходные каналы:

In // вход

Out 1 // выводит данных при правильном ответе

Out 2 // выводит данных при не правильном ответе

// Действие, которое выполняет компонент Check, в терминах Хоара можно описать как:

Check = $\mu X.vx?x \rightarrow (\text{if } x=x\text{эталон then } (\text{вых1!(true)} \rightarrow \text{вых2!(false)} \rightarrow X$
)
else (вых2!(true) \rightarrow вых1!(false) \rightarrow X)

б) загрузка модели системы из текстового файла.

2. Сохранение модели

После описания (загрузки системы из файла) необходимо сохранить модель (команда SAVE для сохранения структуры) и запустить процесс моделирования или процесс верификации модели.

3. Запуск системы для моделирования / верификации.

Моделирование – это цепь расчетов и действий, которые проходят по пути соединений блоков (команда RUN или ► на панели инструментов).

Верификация – проверка правильности модели с помощью аксиоматики Хоара, заложенной в программу (команда VERIFY или V на панели инструментов).

Система выдает пользователю текстовое сообщение с результатами верификации.

Для данной системы можно построить диаграмму взаимодействий, которая приведена в работе [2].

Система моделирования HOARE_SYSMOD, основанная на теории Хоара, является эффективным инструментом моделирования и верификации программных архитектур.

Литература

1. Хоар, Ч. Взаимодействующие последовательные процессы / Ч. Хоар. – М.: Мир, 1989, 264 с.
2. Матеріали всеукраїнської науково-практичної конференції «Проблеми й перспективи розвитку академічної та університетської науки», м. Полтава, 2012.

УДК 681.3.06

ПОРІВНЯЛЬНА ХАРАКТЕРИСТИКА ЗАСОБІВ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ ARENA ROCKWELL SOFTWARE І GPSS WORLD STUDENT VERSION

Е.Н. Гайтан, С.М. Перетяцько

Полтавський національний технічний університет імені Юрія Кондратюка, Україна

Імітаційне моделювання представляє собою метод дослідження, при якому система, що досліджується, замінюється моделлю, що з достатньою