

Ю. Б. ГОЛОВКИН, А. С. ГУСАРЕНКО

ПРИМЕНЕНИЕ НЕЧЕТКИХ ГИПЕРГРАФОВ В МОДЕЛЯХ ГЕНЕРАЦИИ WEB-КОМПОНЕНТОВ

Рассматриваются задачи формирования web-компонентов с использованием моделей генераторов кода на основе управляющих нечетких гиперграфов. Обсуждается описание динамических моделей гиперграфов при программировании web-систем, порталов и сайтов.

Ключевые слова: гиперграфы, XML-технологии, генерация кода, модель web-приложения, сервис, информационно-коммуникационные технологии.

Введение. Использование на предприятиях сервиса современных информационно-коммуникационных технологий позволяет существенно оптимизировать взаимоотношения бизнес-партнеров, упростить процесс их поиска и выбора, способствует заключению выгодных контрактов и повышению уровня обслуживания клиентов.

В современном мире практически любая известная компания имеет в сети Интернет свою web-страницу или сайт с подробной информацией о своей деятельности. Для повышения эффективности использования сайтов в интересах бизнеса первостепенной задачей становится управление web-контентом. Динамическое формирование web-компонентов позволяет обеспечить своевременный обмен информацией между бизнес-партнерами и совместное использование информационных ресурсов, способствует эффективному поиску нужной информации и проведению оперативного анализа в целях повышения эффективности управления предприятиями сервиса. Поэтому разработка моделей и программных средств для динамического формирования web-компонентов является важной задачей.

Генерация кода программ. В настоящее время глобальное информационное пространство обеспечивается множеством Интернет-приложений, инфраструктура которых расширилась за счет динамической обработки контента и представления информационного наполнения, соответствующего потребностям пользователей при управлении различными бизнес-процессами на основании гибких сценариев на сервере. Существенным недостатком моделей, которые используются для формирования web-компонентов, является отсутствие учета информации о текущем состоянии процесса, в результате чего каждая транзакция изолирована от предыдущих и последующих [1, 2]. Для устранения этого недостатка предлагается принцип построения web-систем, в основе которого лежит модель приложения, что позволяет рассматривать поведение системы при взаимодействии с пользователями как класс „ситуация — действие“, а также повышает скорость формирования компонентов [3—7].

В настоящей статье предлагается способ формирования настроек приложения и его архитектуры, а также генераторов кода в виде модели.

Модель представляет собой иерархическую конструкцию по управлению процессом создания приложения и отдельных его частей (форм, моделей, модулей). Для реализации моделей используется формат представления данных XML [8], который как нельзя лучше подходит для решения задач программирования управляющих моделей, поскольку легко описывает сложные структурные части систем. Для обработки программного кода XML-модели целесообразно использовать стандарты DOM всех уровней [8], поддержка которых реализована как на серверной, так и на клиентской стороне для создания асинхронных приложений. С использованием данной модели осуществляется поддержка приложений и их компонентов, так как код генерируется в соответствии с моделью, и для каких-либо изменений не требуется

менять строки кода самого компонента. Такой подход не требует высоких профессиональных качеств от системных аналитиков и проектировщиков.

Иерархическая модель. Иерархическая модель в графической нотации подобна схеме алгоритма. Модель может содержать источники данных, документы, динамические DOM-объекты, переходы в секторы и подчиненные модели. Для перехода по секторам и динамического подключения отдельных модулей, а также установки вероятностных нечетких отношений в гиперграфе предлагается специфицировать вероятностные характеристики как атрибуты какого-либо элемента гиперграфа.

Данная модель управления генерированием кода основана на применении графов в программировании [9]. С помощью графов модель оптимизируется, для того чтобы сократить количество дуг в управляющем графе. Управляющий граф несложно выделяется из основной модели (рис. 1, где *a* — глобальная динамическая модель; *б* — сектор „Компонент“).

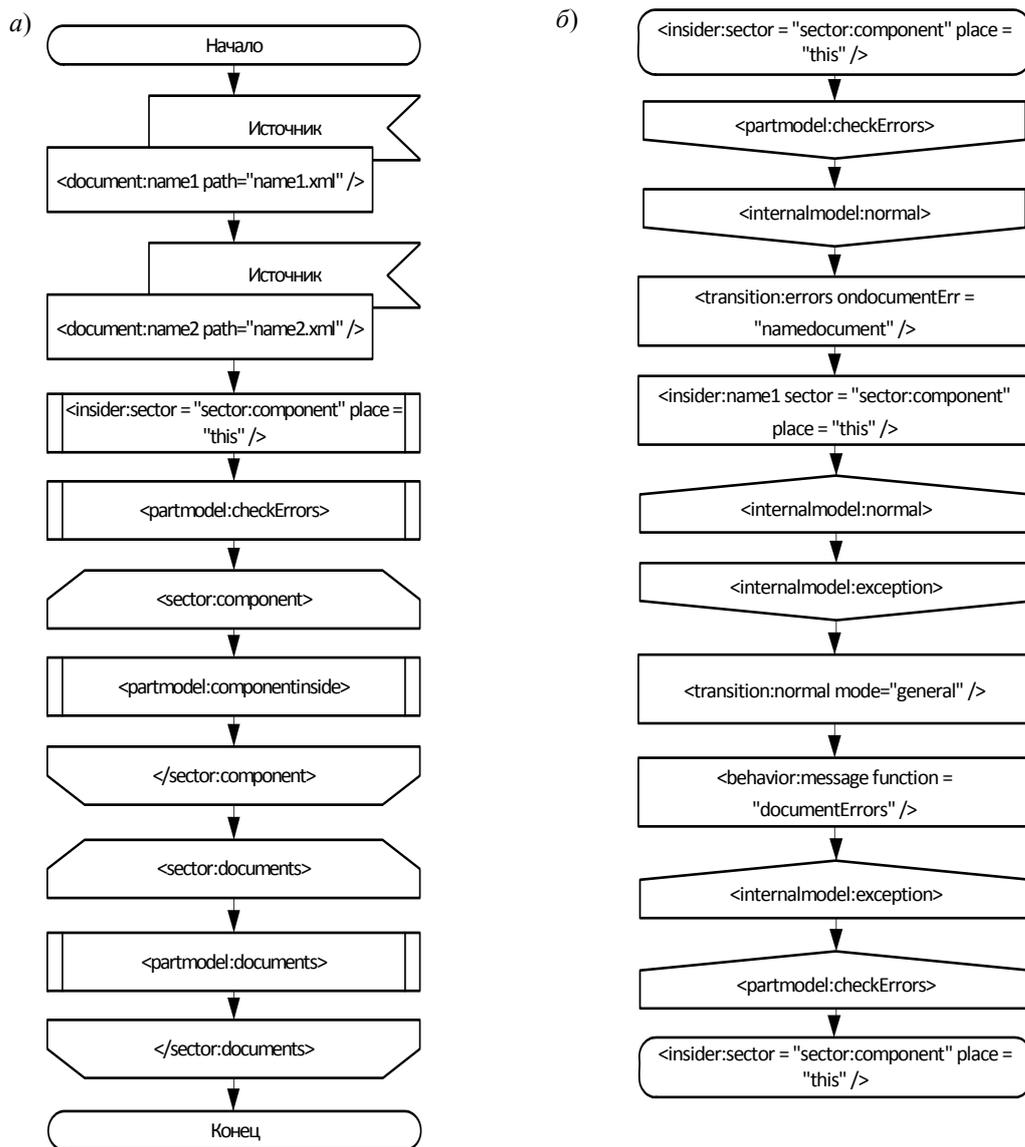


Рис. 1

Предлагается усовершенствовать и описать полученную модель в виде нечеткого гиперграфа (рис. 2, здесь *a* — глобальная модель web-приложения; *б* — сектор „Компонент“). Модель включает несколько составных частей, которые анализируются в процессе выполнения сценариев генерации кода с учетом влияния пользователя на процесс. Фрагменты моделей связаны между собой множественными отношениями и их совокупностями, загружать такие

модели выгоднее поэтапно. Инцидентность между моделями может быть нечеткой — это позволяет строить более сложные отношения между объектами модели по сравнению с моделями четкой инцидентности. Нечеткая инцидентность проявляется, когда с системой генерации кода работает пользователь или когда возникают факторы, которые невозможно учесть.

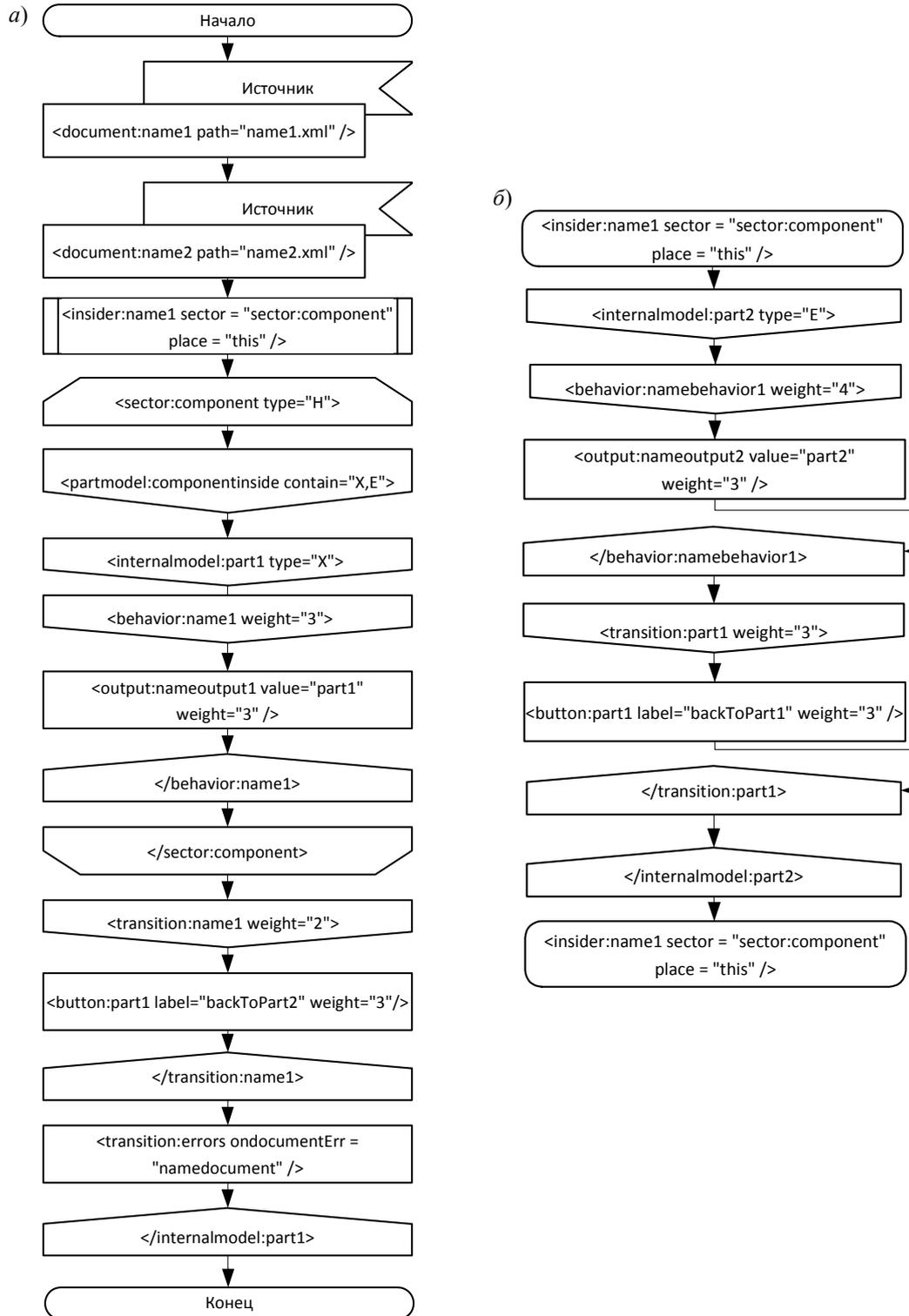


Рис. 2

Нечеткие гиперграфы в модели. Для того чтобы расширить модель нечеткими гиперграфами [10], следует определить способ представления системы с помощью нечетких гиперграфов, а также рассмотреть вопрос их эквивалентного представления. Далее необходимо выделить части нечеткого гиперграфа.

Сформулируем способ задания нечеткого гиперграфа. Нечеткий гиперграф $\tilde{H} = (X, U, \tilde{P})$ задан, если заданы: множество $X = \{x_i\}, i \in I = \{1, 2, \dots, n\}$, называемое множеством вершин; множество $U = \{u_j\}, j \in J = \{1, 2, \dots, m\}$, называемое множеством ребер; двуместный нечеткий предикат, называемый нечетким инцидентором, который определяется для всех пар как $(x, u), x \in X, u \in U$, и принимает значения из интервала $[0; 1]$. Нечеткий инцидентор \tilde{P} порождает нечеткое множество $F(\tilde{P}) = \{\mu_{F(P)}(x, u) / (x, u)\}, x \in X, u \in U$, во множестве $X \times U$, где $\mu_{F(P)}$ — функция принадлежности, определяющая для каждой пары (x, u) степень инцидентности $\mu_{F(P)}(x, u)$ входящих в нее элементов гиперграфа. Множество $F(\tilde{P})$ называется областью нечеткой истинности нечеткого инцидентора \tilde{P} .

Эквивалентным способом задания нечеткого гиперграфа $\tilde{H} = (X, U, \tilde{P})$ является матрица $\tilde{R}_H = \|r_{ij}\|_{n \times m}$, где $r_{ij} = \mu_{F(P)}, x_i \in X, u_j \in U$, называемая матрицей нечеткой инцидентности.

Представление систем нечеткими гиперграфами. Нечеткий гиперграф $\tilde{H} = (X, U, \tilde{P})$ представляет собой совокупность нечетких ситуаций, описывающих объект управления при решении задач вида „класс ситуаций — действие“, если множество этих ситуаций обозначить как $X = \{x_i\}, i \in I = \{1, 2, \dots, n\}$; множество значений лингвистических переменных, характеризующих базовое множество признаков, которыми ситуации $x \in X$ обладают в той или иной степени, обозначить как $U = \{u_j\}, j \in J = \{1, 2, \dots, m\}$, и положить, что двуместный нечеткий инцидентор \tilde{P} определяет степень инцидентности для всех пар $(x, u), x \in X, u \in U$, т.е. между каждой ситуацией и каждым значением. Таким образом, в матрице инцидентности \tilde{R}_H каждая нечеткая ситуация $x_i \in X$ отображается строкой x_i .

В случае если необходимо проанализировать систему, заданную в виде совокупности нечетких ситуаций, время от времени возникают задачи их классификации, а также построения отношений между ними и анализа структуры гиперграфа в целях планирования поиска требуемых ситуаций и декомпозиции множества ситуаций на группы.

При принятии решений в данных условиях естественно представляется равнозначность по отношению к целям и ограничениям, которая устраняет различия между ними. При этом нечеткий гиперграф $\tilde{H} = (X, U, \tilde{P})$ представляет собой совокупность целей и ограничений, если положить, что $X = \{x_i\}, i \in I$, определяет заданное множество альтернатив, а множество $U = U_1 \cup U_2$ определяет совокупность целей и ограничений, где $U_1 = \{u_j\}, j \in J_1$, — множество нечетких целей, $U_2 = \{u_k\}, k \in J_2$, — множество нечетких ограничений. Нечеткий инцидентор \tilde{P} задает степень инцидентности между $x_i \in X$ (альтернативой) и $u_i \in U$ (целью или ограничением). В матрице \tilde{R}_H каждая нечеткая цель $u_j \in U_1$ и каждое ограничение $u_k \in U_2$ представляются соответствующим столбцом. Принятие решения в этом случае сводится к выполнению теоретико-множественных операций на множестве элементов гиперграфа и, при необходимости, к поиску некоторых его экстремальных характеристик. В случае когда цели и ограничения — нечеткие множества в разных пространствах X и Y , при помощи соответствия $f: X \rightarrow Y$, которое также может быть нечетким, они могут быть сведены в одно пространство и заданы нечетким гиперграфом. При этом следует заметить, что нечеткое со-

ответствие $\tilde{f}: X \rightarrow Y$ представляется нечетким гиперграфом $\tilde{H} = (X, U, \tilde{P})$, если положить, что нечеткий граф нечеткого соответствия и область истинности нечеткого инцидентора \tilde{P} совпадают. Отсюда следует, что различные нечеткие системы инцидентностей, особенно с участием пользователей, такие как „пользователи — задачи“, иерархические системы управления и операционно-технологические системы с участием человека, могут быть представимы неориентированными нечеткими гиперграфами. Анализ и синтез таких систем может быть сведен к выполнению последовательностей теоретико-множественных алгебраических операций, исследованию их структурных свойств, выявлению экстремальных характеристик.

Пример определения нечеткого гиперграфа. Пусть

$$X = \{x_1, x_2, x_3, x_4, x_5, x_6\}, U = \{u_1, u_2, u_3, u_4, u_5\},$$

$$F(\tilde{P}) = \{ \langle 0,4 / (x_1, u_3) \rangle, \langle 0,1 / (x_1, u_4) \rangle, \langle 1 / (x_2, u_2) \rangle, \langle 0,7 / (x_2, u_3) \rangle, \langle 1 / (x_2, u_4) \rangle, \langle 0,8 / (x_2, u_5) \rangle, \langle 0,2 / (x_3, u_1) \rangle, \langle 1 / (x_3, u_2) \rangle, \langle 0,7 / (x_3, u_5) \rangle, \langle 0,7 / (x_4, u_1) \rangle, \langle 1 / (x_5, u_1) \rangle, \langle 0,6 / (x_5, u_4) \rangle, \langle 0,3 / (x_5, u_5) \rangle, \langle 0,8 / (x_6, u_3) \rangle, \langle 1 / (x_6, u_4) \rangle \}.$$

При этом $\tilde{H} = (X, U, \tilde{P})$ является нечетким гиперграфом с матрицей нечеткой инцидентности

$$\begin{array}{c} u_1 \quad u_2 \quad u_3 \quad u_4 \quad u_5 \\ \begin{array}{c} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{array} \left\| \begin{array}{ccccc} 0 & 0 & 0,4 & 0,1 & 0 \\ 0 & 1 & 0,7 & 1 & 0,8 \\ 0,2 & 1 & 0 & 0 & 0,7 \\ 0,7 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0,6 & 0,3 \\ 0 & 0 & 0,8 & 1 & 0 \end{array} \right\| \end{array}.$$

Пример альтернативного определения нечеткого гиперграфа. Нечеткий гиперграф, рассмотренный выше, может быть задан в виде

$$\tilde{H} = (X, E),$$

где

$$\begin{aligned} X &= \{x_1, x_2, x_3, x_4, x_5, x_6\}, E = \{\tilde{e}_1, \tilde{e}_2, \tilde{e}_3, \tilde{e}_4, \tilde{e}_5\}, \\ \tilde{e}_1 &= \{ \langle 0,2 / x_3 \rangle, \langle 0,7 / x_4 \rangle, \langle 1 / x_5 \rangle \}, \\ \tilde{e}_2 &= \{ \langle 1 / x_2 \rangle, \langle 1 / x_3 \rangle \}, \\ \tilde{e}_3 &= \{ \langle 0,4 / x_1 \rangle, \langle 0,7 / x_2 \rangle, \langle 0,8 / x_6 \rangle \}, \\ \tilde{e}_4 &= \{ \langle 0,1 / x_1 \rangle, \langle 1 / x_2 \rangle, \langle 0,6 / x_5 \rangle, \langle 1 / x_6 \rangle \}, \\ \tilde{e}_5 &= \{ \langle 0,8 / x_2 \rangle, \langle 0,7 / x_3 \rangle, \langle 0,3 / x_5 \rangle \}. \end{aligned}$$

Нечеткий гиперграф

$$\tilde{H}^* = (U, L),$$

где

$$\begin{aligned} U &= \{u_1, u_2, u_3, u_4, u_5\}, L = \{ \tilde{l}_1, \tilde{l}_2, \tilde{l}_3, \tilde{l}_4, \tilde{l}_5, \tilde{l}_6 \}, \\ \tilde{l}_1 &= \{ \langle 0,4 / u_3 \rangle, \langle 0,1 / u_4 \rangle \}, \\ \tilde{l}_2 &= \{ \langle 1 / u_2 \rangle, \langle 0,7 / u_3 \rangle, \langle 1 / u_4 \rangle, \langle 0,8 / u_5 \rangle \}, \end{aligned}$$

$$\tilde{l}_3 = \{\langle 0, 2 / u_1 \rangle, \langle 1 / u_2 \rangle, \langle 0, 7 / u_5 \rangle\},$$

$$\tilde{l}_4 = \{\langle 0, 7 / u_1 \rangle\},$$

$$\tilde{l}_5 = \{\langle 1 / u_1 \rangle, \langle 0, 6 / u_4 \rangle, \langle 0, 3 / u_5 \rangle\},$$

$$\tilde{l}_6 = \{\langle 0, 8 / u_3 \rangle, \langle 1 / u_4 \rangle\},$$

является двойственным гиперграфу \tilde{H} .

Для гиперграфа \tilde{H} степени вершин и ребер составляют:

$$p(x_1) = 2, p(x_2) = 4, p(x_3) = 3, p(x_4) = 1, p(x_5) = 3, p(x_6) = 2,$$

$$p(u_1) = 3, p(u_2) = 2, p(u_3) = 3, p(u_4) = 4, p(u_5) = 3.$$

Максимальные степени вершин и ребер соответственно равны

$$\mu(x_1) = 0, 4, \mu(x_2) = 1, \mu(x_3) = 1, \mu(x_4) = 0, 7, \mu(x_5) = 1, \mu(x_6) = 1,$$

$$\mu(u_1) = 1, \mu(u_2) = 1, \mu(u_3) = 0, 8, \mu(u_4) = 1, \mu(u_5) = 0, 8.$$

Определение нечеткого гиперграфа в модели генерации программного кода. Для определения нечеткого гиперграфа в модели генерирования программного кода и секторах необходимо учитывать навигацию и обозначения секторов, а также указывать, является ли часть данного конкретного участка модели ее составным фрагментом или фрагментом дочерней модели. Для этого определяются сами гиперграфы с помощью меток $H(\text{type}=\text{“H”})$. При определении подчиненных моделей для вышестоящих модульных моделей отмечаются множества, которые входят в данный гиперграф, это также выполняется с помощью атрибутов модели.

Вероятностные характеристики задаются с помощью атрибутов “weight”. Любые другие настройки, которые необходимы проектировщику, следует специфицировать в атрибутах. Сами по себе вероятностные характеристики ничего не определяют, если, конечно, отсутствуют инструментарий и метод для анализа модели с учетом атрибутивных спецификаций и классификаций ситуаций. Для анализа модели создается устройство, к которому предъявляются требования по обработке древовидных структур. Для обработки древовидных структур [11] создана спецификация DOM, позволяющая анализировать и генерировать код.

Заключение. Итак, установлено, что модель генерации web-компонентов целесообразно строить в древовидном виде и определять нечеткие гиперграфы, что вполне совместимо со структурной нотацией, описываемой форматом XML. Для учета случайной составляющей системы, которая будет непосредственно взаимодействовать с пользователем, следует задавать модели гиперграфов. Это является важной задачей, так как пользователь в одной и той же ситуации может вести себя по-разному. Для динамического построения и обработки модели проектировщику может потребоваться часть системы, например контроллер. Контроллер реализуется на том же языке программирования, что и сама система. Функцией контроллера является генерирование кода и переходов по модели. В случае если модель предназначена для работы с потоком информации, то данные в формате XML могут ранжироваться в соответствии с нечетким графом, а данные, запрошенные наибольшее количество раз, могут дополнительно загружаться в память автоматически при срабатывании перехода к очередному состоянию.

СПИСОК ЛИТЕРАТУРЫ

1. Миронов В. В., Маликова К. Э. Интернет-приложения на основе встроенных динамических моделей // Вестн. УГАТУ: науч. журн. Уфимск. гос. авиац. техн. ун-та. 2009. Т. 13, № 2.
2. Миронов В. В., Юсупова Н. И., Шакирова Г. Р. Иерархические модели данных: концепции и реализация на основе XML. М.: Машиностроение, 2011.

3. Головкин Ю. Б., Гусаренко А. С. Информационная система кредитования юридических лиц в банке // Актуальные инновационные исследования: наука и практика. 2010. № 3.
4. Головкин Ю. Б., Миронов В. В., Юсупова Н. И. Об алгоритмах управления по состоянию ситуации // Управление сложными техническими системами: Межвуз. науч. сб. Уфа: УАИ, 1985. № 8.
5. Головкин Ю. Б., Миронов В. В., Юсупова Н. И. Об автоматной модели ситуационного управления // Управление сложными техническими системами: Межвуз. науч. сб. Уфа: УАИ, 1987. № 10.
6. Tatarnikova T., Kolbanev M. Statement of a task corporate information networks interface centers structural synthesis // IEEE EUROCON 2009. St. Petersburg, 2009. P. 1883—1887.
7. Богатырев В. А. Надежность и эффективность резервированных компьютерных сетей // Информационные технологии. 2006. № 9. С. 25—30.
8. Гусаренко А. С. Управление XML-данными на основе динамических DOM-объектов // Перспективы развития информационных технологий: Сб. материалов 6-й Междунар. науч.-практ. конф. Новосибирск: Сибпринт, 2011. С. 103—108.
9. Пуха Г. П. Современное высокоуровневое и объектно-ориентированное программирование. СПб: СПбГУСЭ, 2013.
10. Зоболотский В. П., Оводенко А. А., Степанов А. Г. Математические модели в управлении: Учеб. пособие. СПб: СПбГУАП, 2001. 196 с.
11. Богатырев В. А., Богатырев С. В., Богатырев А. В. Оптимизация древовидной сети с резервированием коммутационных узлов и связей // Телекоммуникации. 2013. № 2. С. 42—48.

Сведения об авторах

- Юрий Борисович Головкин** — канд. техн. наук, доцент; Санкт-Петербургский государственный экономический университет, кафедра прикладных информационных технологий;
E-mail: comparif@rambler.ru
- Артём Сергеевич Гусаренко** — канд. техн. наук; Уфимский государственный авиационный технический университет, кафедра автоматизированных систем управления; ассистент;
E-mail: artyomgusarenko@gmail.com

Рекомендована кафедрой
прикладных информационных
технологий

Поступила в редакцию
28.04.14 г.