

Т. А. Волянская

ВИРТУАЛЬНЫЙ МУЗЕЙ ИСТОРИИ ИНФОРМАТИКИ В СИБИРИ: МОДЕЛЬ ПРЕДМЕТНОЙ ОБЛАСТИ И МОДЕЛЬ ПОЛЬЗОВАТЕЛЯ*

ВВЕДЕНИЕ

Статья посвящена проекту создания виртуального музея истории информатики в Сибири, работа над которым ведется коллективом сотрудников ИСИ СО РАН, ИМ СО РАН и НГУ при финансовой поддержке Российского гуманитарного научного фонда (грант РГНФ N 02-05-12010). Музей создается в виде информационно-поисковой, справочной адаптивной гипермедиа-системы, доступной в интернете [1–5].

В настоящее время в интернете представлено большое число виртуальных музеев, подавляющее большинство из которых реализовано с использованием традиционных гипермедиа-технологий. Одним из ограничений традиционных гипермедиа-систем, предназначенных для использования различными категориями пользователей, является то, что они предоставляют одно и то же информационное содержание и один и тот же механизм навигации всем пользователям. Разрабатываемый виртуальный музей предназначен для различных категорий пользователей, и его посетители, имеющие различные цели, интересы, знания и предпочтения, могут нуждаться в различных частях содержащейся информации и использовать различные пути для навигации. Поэтому при создании музея особое внимание уделяется вопросам его адаптации [2, 3].

Структура статьи следующая: в первом разделе кратко представлены адаптивная гипермедиа и адаптивные гипермедиа-системы. Во втором разделе рассматриваются архитектура адаптивных гипермедиа-систем и ее основные компоненты (модель предметной области, модель пользователя и модель адаптации), а также вопросы моделирования предметной области и моделирования пользователя. Третий раздел посвящен вопросам, относящимся к виртуальному музею истории информатики в Сибири: рассматривается структура и содержимое музея, категории пользователей, а также представление знаний предметной области и моделирование пользователя.

* Работа выполнена при финансовой поддержке Российского гуманитарного научного фонда (грант № 02-05-12010) и Министерства образования РФ.

1. АДАПТИВНЫЕ ГИПЕРМЕДИА-СИСТЕМЫ

Адаптивная гипермедиа (Adaptive Hypermedia) (АГ) — альтернатива традиционному подходу разработки гипермедиа-систем. *Адаптивные гипермедиа-системы* (АГС) поддерживают модель целей, знаний, интересов, предпочтений и других особенностей индивидуального пользователя и используют ее в течение взаимодействия для адаптации к потребностям пользователя [3,6–10].

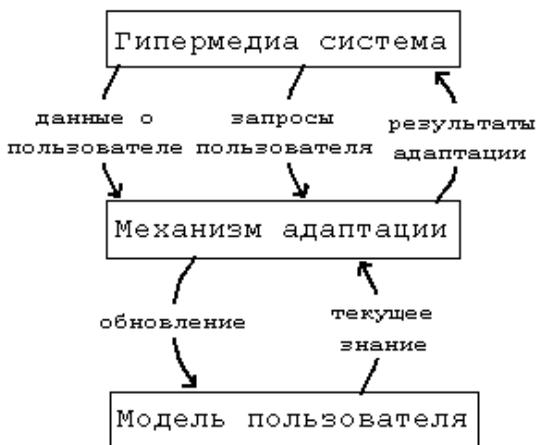


Рис. 1. Адаптивная гипермедиа-система

АГС в общем случае поддерживают следующие три вида адаптации: к данным пользователя, к рабочим характеристикам и к данным окружения. Данные пользователя включают различные характеристики пользователей: знания, цели, подготовку, опыт в гиперпространстве, предпочтения, интересы и индивидуальные особенности пользователя. Рабочие характеристики включают данные о взаимодействии пользователя с системой, которые не могут быть сведены к характеристикам пользователя, но все еще могут использоваться для принятия решений адаптации. Данные окружения включают все аспекты пользовательского окружения, которые не связаны с пользователями (например, аппаратные средства, программное обеспечение, пропускная способность сети или местонахождение пользователя) [3, 6–8].

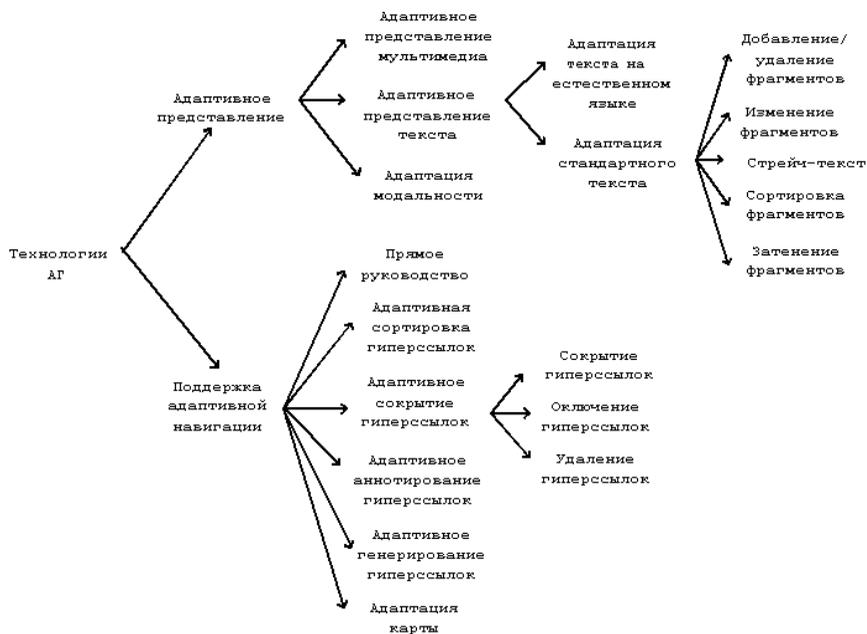


Рис. 2. Классификация технологий АГ

АГС системы обеспечивают *адаптивное представление (adaptive presentation)*, т.е. адаптацию содержания гипердокументов, и *адаптивную поддержку навигации (adaptive navigation support)*, т.е. адаптацию структуры гиперссылок. Смысл методов адаптивного представления состоит в том, чтобы адаптировать содержание страницы, к которой обращается отдельный пользователь, к текущему знанию, предпочтениям, интересам, целям и другим характеристикам пользователя. Основные методы адаптивного представления текста — это дополнительные, предварительные и сравнительные объяснения, варианты объяснения и сортировка. Следующие технологии используются для реализации вышеперечисленных методов адаптивного представления текста: условный текст, стрейч-текст, варианты фрагментов и варианты страниц, фреймовая технология. Смысл методов адаптивной навигационной поддержки состоит в том, чтобы помочь пользователям найти путь в гиперпространстве с помощью адаптации способа представления ссылок к целям, знанию и другим характеристикам индивидуального пользователя. Методы адаптивной навигационной поддержки используются для достижения нескольких целей адаптации: обеспечить

глобальное руководство, локальное руководство, глобальную ориентацию, локальную ориентацию и управление индивидуализированными представлениями в информационных пространствах. Для реализации этих методов применяются следующие технологии: полное руководство, сортировка, сокрытие, аннотирование, генерирование ссылок и адаптация карты [3, 6–8, 10, 11].

2. АРХИТЕКТУРА АГС

На абстрактном уровне АГС можно описать с помощью следующей архитектуры, базирующейся на трех компонентах [2,11,12,15–17]:

- модель предметной области (Domain Model),
- модель пользователя (User Model),
- модель адаптации (Adaptation Model).

2.1. Модель предметной области (МО)

Модель предметной области (МО) дает описание предметной области (ПО) на концептуальном уровне и представляет собой совокупность концептов и межконцептных отношений. *Концепты* — это абстрактные объекты, используемые для представления элементов информации ПО. Существуют концепты нижнего уровня или *атомные концепты*, которые соответствуют одному фрагменту информации, и концепты высшего уровня или *составные концепты*, которые состоят из множества других концептов (высшего или нижнего уровней) [11,12,15–17].

Межконцептные отношения представляют различные отношения между концептами. Следующие типы отношений между концептами являются наиболее используемыми:

- тип *часть (part-of)* представляет композиционное отношение: показывает, какую долю первый концепт составляет от второго;
- тип *связь (link)* означает, что существует связь первого концепта со вторым (например, в виде гиперссылки);
- тип *предпосылка (prerequisite)* означает, что первый концепт должен быть предварительно изучен до изучения второго концепта;
- тип *ингибитор (inhibitor)* означает, что первый концепт не должен изучаться до тех пор, пока не будет изучен второй.

Можно выделить три типа моделей предметной области, различающихся по уровню сложности структуры [7]:

- самую простую структуру имеет МО *первого уровня*, являющаяся независимым множеством концептов (отсутствуют межконцептные отношения);
- МО *второго уровня* (сетевая МО) предполагает наличие связей между концептами и представляет собой семантическую сеть, состоящую из концептов и межконцептных отношений. Может существовать несколько типов концептов и межконцептных отношений;
- МО *третьего уровня* (фреймовая МО) предполагает наличие у концептов внутренней структуры в виде множества атрибутов, при этом концепты различных типов могут иметь различные множества атрибутов.

Одной из самых важных функций МО является обеспечение структуры для представления знаний пользователя ПО.

АГС можно подразделить на три основные группы в зависимости от метода организации связи между МО (концептами) и гиперпространством системы (гипермедиа-страницами) [7]:

- самый простой метод — это *индексация страниц* концептами, относящимися к содержанию этих страниц; он может применяться даже для МО первого уровня;
- второй метод, похожий на предыдущий, — это *индексация фрагментов*: содержание страницы разбивается на множество фрагментов, каждый из которых отдельно индексируется множеством концептов, относящихся к содержанию данного фрагмента. При разбиении на относительно небольшие фрагменты каждый фрагмент индексируется ровно одним концептом. Этот метод также можно использовать при МО первого уровня;
- третий метод (*прямая связь*) отличается от предыдущих методов тем, что для страниц не поддерживаются индексы; гиперпространство АГС строится непосредственно исходя из структуры МО. Таким образом, каждый концепт МО представлен гипермедиа-страницей или гипердокументом, а отношения между концептами соответствуют гиперссылкам между страницами. Страница или документ, представляющий концепт, могут быть как статическими, так и динамическими: т.е. генерироваться налету, исходя из внутренней структуры концепта. Последний метод является самым мощным из всех вышеперечисленных, однако он требует использования МО второго, а лучше третьего уровня.

2.2. Модель пользователя (МП)

2.2.1. Моделирование пользователя

Моделирование пользователя можно определить как адаптацию поведения системы в процессе взаимодействия с ним в соответствии с теми предположениями, которые делает система, основываясь на собранной о пользователе информации. Для обеспечения адаптации АГС необходима модель пользователя, с помощью которой система может определить что, когда и как должно быть адаптировано. По существу МП — это набор фактов о пользователе, сопутствуемый набором процессов, которые ими управляют: добавление в МП новой информации по мере ее поступления, разрешение противоречий в сделанных предположениях и выведение по возможности новой информации. Исследования в области моделирования пользователя затрагивают проблемы представления МП в системе, формирования предположений о пользователе из входных данных или других источников, а также использования системой этих предположений для адаптации взаимодействия к пользователю.

2.2.2. Модель пользователя

Модель пользователя (МП) в АГС предполагает *явное* представление знаний, предпочтений, целей, интересов, истории навигации и других характеристик пользователя и служит для адаптации к пользователю различных аспектов АГС. МП состоит из именованных элементов, для которых хранится набор пар вида атрибут-значение (компонентов МП). На концептуальном уровне можно представлять МП в виде табличной структуры, в которой для каждого элемента хранятся значения атрибутов. Большинство элементов в МП представляют концепты МО. Некоторые другие элементы могут представлять различные аспекты пользователя, такие как цели, предпочтения, интересы или стереотипная классификация (типа новичок, эксперт) и т.д. [11, 12, 15–17].

Можно классифицировать МП согласно некоторым основным аспектам:

- *способ получения (явный и неявный)*: при явном способе система в явном виде запрашивает у пользователя информацию, необходимую для создания и обновления модели. Альтернативный подход, при котором система незаметно и ненавязчиво для пользователя конструирует МП, отслеживая и анализируя взаимодействия с пользователем (например, историю навигации), с последующим выведением различных предположений о пользователе;

- *степень специализации (общие и индивидуальные модели)*: общая модель предполагает наличие одной общей модели для всех пользователей системы, она используется в случае, когда система предназначена для эксплуатации однородной группой пользователей. Индивидуальная модель принимает во внимание индивидуальные особенности пользователей и поддерживается для каждого пользователя своя. Стереотипная модель является способом комбинирования двух вышеперечисленных моделей: стереотип — набор характеристик, связанных друг с другом, определяющий некоторый класс пользователей;
- *модифицируемость (статические и динамические модели)*: в то время как статическая модель не изменяется, динамическая модель постоянно обновляется по мере получения новой информации в течение сеанса взаимодействия с пользователем;
- *временная протяженность (краткосрочные и долгосрочные модели)*: в отличие от краткосрочных моделей, долгосрочные модели сохраняются от одного сеанса взаимодействия с пользователем до другого;
- *метод использования (дескриптивные и прескриптивные модели)*: более традиционным является дескриптивное использование МП, в этом представлении модель пользователя — простая база данных, содержащая информацию о пользователе, у которой система может запрашивать текущие данные о пользователе. Прескриптивное использование модели предполагает, что система моделирует (имитирует) пользователя для проверки интерпретации его ответа.

2.2.3. Методы моделирования

Можно выделить два основных метода моделирования пользователя: оверлейное и стереотипное моделирование [13, 14].

Оверлейное моделирование (overlay modeling) чаще всего используется в интеллектуальных системах обучения для моделирования знаний, при этом знания пользователя описываются как подмножество знаний ПО эксперта (т.н. «оверлей»). Для каждого концепта МО в модели пользователя вычисляется и сохраняется некоторое значение (или несколько значений), оценивающее уровень знания пользователем этого концепта (оверлейная модель). Оверлейная модель знаний (overlay model) может быть представлена множеством пар «концепт — значения атрибутов».

Следующие атрибуты обычно используются для обозначения уровня знаний концепта пользователем [15–17].

- «*Значение знания*» (или просто *значение*) показывает уровень знания концепта пользователем. Все пары концепт–значение вместе формируют оверлейную модель, которая представляет «знания» пользователя. Некоторые АГС используют булеву модель пользователя, в которой для каждого концепта определено, знает или не знает пользователь концепт. Другие АГС используют небольшой набор значений, например, «неизвестен», «изучен», «хорошо изучен» и «известен» или большой набор (например, проценты).
- Атрибут «*чтения*» показывает, просматривал (читал, изучал) ли пользователь какую-нибудь информацию (фрагмент, страницу или набор страниц) о концепте. В веб-системах атрибут чтения используется, чтобы генерировать различное представление для гиперссылок к посещенным и непосещенным страницам (по умолчанию это фиолетовый и синий цвета). Атрибут чтения может иметь булевы значения в некоторых АГС, в то время как в других АГС это может быть список времен доступа.
- Менее распространенный атрибут — это «*готовность для чтения*», который показывает, готов ли пользователь для просмотра и изучения информации об этом концепте (это означает, например, что пользователь уже приобрел достаточное количество предварительно необходимых знаний.)

Использование оверлейной модели особенно целесообразно, когда материал обучения может быть представлен в виде иерархии предварительных условий. Как уже было сказано выше, в рамках модели предполагается, что знание пользователя составляет некоторое подмножество знания эксперта, и цель обучения состоит в расширении этого подмножества. Модель также предполагает, что пользователь не будет изучать того, чего не знает эксперт. В частности, не принимаются во внимание неправильные представления и заблуждения, изначально имеющиеся у пользователя, или приобретенные им в процессе обучения. Вторым недостатком оверлейной модели заключается в том, что нет механизма для разграничения знаний, которые пользователь еще не приобрел, и знаний, которые еще не были ему представлены, что имеет смысл для стратегии обучения.

Дифференциальная модель (differential model) обращается к этим вопросам. Дифференциальная модель является расширением оверлейной модели. Знание разделено на то, которое было продемонстрировано пользователю, и то, которым пользователь не обладает. Модель подразделяет все знание ПО

на представленное и не представленное пользователю. Оверлейная модель применяется к знанию, уже представленному пользователю. В отличие от оверлейной, дифференциальная модель принимает во внимание неправильные представления и ошибки пользователя.

Пертурбационная модель (perturbation model) принимает во внимание знания, которыми обладает пользователь, но которые не представлены в модели знания эксперта ПО. Пертурбационная модель расширяет модель эксперта добавлением библиотеки ошибок (bug library). Процесс ее создания может быть перечисляющим или порождающим. Перечисляющий процесс составляет список всех возможных неправильных представлений с помощью анализа ПО и ошибок, которые допускает пользователь. Порождающий метод пытается генерировать ошибки исходя из лежащей в основе познавательной теории. Оверлейная модель может быть применена поверх комбинированной модели эксперта и библиотеки ошибок. Как и для простой оверлейной модели, цель обучения — увеличить подмножество знания эксперта при исключении неправильных представлений.



Рис. 3. Оверлейная, дифференциальная и пертурбационная модели знаний

Стереотипное моделирование (stereotype modeling) — один из первых методов в области моделирования пользователя, предложенный в 1979 году Элэйн Рич (Elaine Rich). Этот метод целесообразно использовать в случаях, когда требуется быстрая, но не обязательно правильная оценка пользователя.

Можно определить стереотип как набор некоторых взаимосвязанных характеристик, присущих всем членам определенной подгруппы пользователей. Стереотипная модель различает несколько типичных или «стереотипных» пользователей (например, «новичок», «средний», «эксперт» и т.п.). В процессе моделирования все пользователи группируются в классы (каждый пользователь ассоциируется с одним или более стереотипом), при этом

полагается, что пользователи, принадлежащие к одному классу, имеют одинаковые характеристики. Стереотипная классификация может быть сделана для каждого аспекта адаптации. Стереотипная модель может быть представлена как множество пар «стереотип–значение», где значением может быть «истина» или «ложь» (означающее, принадлежит ли пользователь данному стереотипу).

Кроме характеристик подгрупп пользователей, стереотипы могут содержать так называемые триггеры (инициирующие условия), которые представляют ключевые характеристики, позволяющие идентифицировать пользователя как принадлежащего к соответствующей подгруппе пользователей. Триггеры могут относиться к текущим пользовательским характеристикам, эксплуатационным характеристикам (например, истории навигации), данным окружения (например, данным о конфигурации, оборудовании). Стереотипы применяются к пользователю, если они назначены «вручную» или если их триггеры совпадают с доступной информацией о пользователе (автоматическая классификация). В результате все характеристики соответствующего стереотипа приписываются пользователю.

Суммируя вышесказанное, можно представить, что стереотип состоит из следующих частей:

- множества инициирующих условий (триггеров), являющиеся логическими выражениями, активирующими стереотип;
- множества условий отвода (ретракций), которые ответственны за деактивацию активного стереотипа;
- множества предположений (выводов) стереотипа, которые служат предположениями по умолчанию при ассоциировании пользователя со стереотипом.

Можно сформулировать три этапа, которые нужно выполнить при разработке стереотипов:

- определение подгрупп пользователей: необходимо выделить внутри группы пользователей подгруппы, члены которых имеют некоторые однородные характеристики;
- идентификация ключевых характеристик: требуется определить небольшое число ключевых характеристик, позволяющих идентифицировать членов подгруппы пользователей (присутствие или отсутствие этих характеристик должно быть распознаваемо системой);
- представление в виде (иерархически упорядоченных) стереотипов: требуемые характеристики определенных групп пользователей должны быть формализованы в подходящей системе представления. Совокупность представленных характеристик подгруппы поль-

зователей называется «стереотипом» этой подгруппы. Если содержимое одного стереотипа образует подмножество содержимого другого, может быть построена иерархия стереотипов, в которой содержимое стереотипа более высокого уровня наследуется стереотипом более низкого уровня и таким образом представлено один раз.

При использовании стереотипного моделирования может возникнуть следующая проблема: стереотипы могут быть так специализированы, что будут состоять только из одного пользователя, или пользователь вообще не сможет быть классифицирован.

Можно определить четыре модели согласно типу организации связи между стереотипами: луковая (многоуровневая) модель, латуковая модель, многоядерная латуковая модель, ориентированный ациклический граф.

- «*Многоуровневая*» или «*луковая модель*» (*onion model*) — иерархическая модель, в которой содержимое стереотипов упорядочено в отношении линейных подмножеств. Пользователи, принадлежащие к стереотипу S , наследуют знание, ассоциированное с более общими стереотипами, чем S (наследование без исключений). В качестве примера можно привести классификацию пользователей, в которой «новички» знают подмножество концептов A ; у «продвинутых» пользователей есть знания новичков, но они также знают концепты из подмножества B ; наконец, «эксперты» знают концепты из A , B и C .
- «*Латуковая модель*» (*lettuce model*): для этой модели характерно наличие стереотипа-ядра, являющегося подмножеством всех других стереотипов, которые могут быть независимы друг от друга. Эта модель часто используется для представления знаний пользователей в ПО подобных командам UNIX: пользователи владеют только относительно небольшим множеством основных команд и рядом функционально связанных команд (например, для управления файловой системой, электронной почтой и т.д.).
- «*Многоядерная латуковая модель*» (*multikernel lettuce model*): это обобщение латуковой модели, в которой существует несколько ядер, являющихся пересечениями некоторых стереотипов.
- «*Ориентированный ациклический граф*» (*DAG*): многоядерная модель расширяется тем, что общности из двух или более ядер также сжимаются в ядро. В результате получается иерархия стереотипов, в которой каждый узел может иметь больше одного узла высшего уровня.

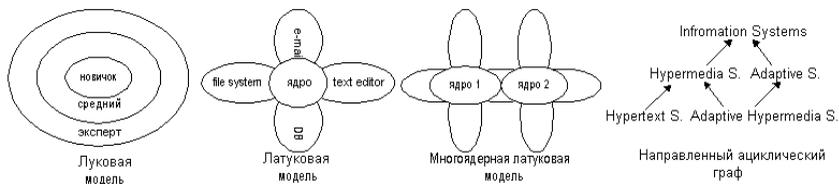


Рис. 4. Модели стереотипов: луковая, латуковая, многоядерная латуковая модель и ориентированный ациклический граф

2.3. Модель адаптации (МА)

Модель адаптации (МА) описывает, как должна происходить адаптация в зависимости от МО и МП. Она состоит из правил адаптации, которые формируют связь между МО и МП и определяют представление генерируемой информации и обновление модели пользователя [11, 12, 15–17].

Обычно правило имеет вид: *если <условие> то <действие>*, где условие может определять происхождение внешнего события, например, обращение к странице, плюс логическое выражение, относящееся к значениям атрибутов из МО или МП. Действием может быть модификация значений атрибутов в МП или присваивание объекту спецификации представления. Также в правиле может указываться «фаза» выполнения, указывающая на момент времени, в который должно применяться правило: до или в течение генерации (фаза “pre”) и после генерации страницы (фаза “post”). Основанием для наличия двух стадий выполнения является то, что сначала может потребоваться осуществить некоторую адаптацию, основываясь на «текущем» состоянии модели пользователя (фаза “pre”), а затем модифицировать модель пользователя после генерации представления страницы (фаза “post”). Также в правиле может быть определено, может ли оно инициировать запуск других правил или нет.

Все правила адаптации подразделяются на «общие» и «специальные» правила. В отличие от общих правил, которые выполняются для всех концептов, в специальных правилах указывается множество концептов, к которым эти правила должны применяться. Специальные правила имеют приоритет над общими правилами, и, таким образом, они используются для определения исключений в общих правилах. В качестве примера можно привести два простых правила, написанных с использованием произвольно выбранного синтаксиса:

Пример 1. Следующее правило определяет, что при обращении к странице для соответствующего концепта в МП устанавливается атрибут «чтения» равным истине в фазе “post”:

$$\langle access(C) \Rightarrow C.read := true; post; true \rangle$$

Правило также утверждает, что оно запустит другие правила, которые имеют атрибут «чтения» ($C.read := true$) в своей левой части.

Пример 2. Следующее правило выражает, что когда пользователь обращается к странице, определяющей концепт, «готовый для чтения», то значение знания для этого концепта становится «изученным» в фазе “pre”:

$$\langle (access(C) \ \& \ C.ready-to-read = true) \Rightarrow C.knowledge-value := learned; pre; true \rangle$$

3. ВИРТУАЛЬНЫЙ МУЗЕЙ ИСТОРИИ ИНФОРМАТИКИ В СИБИРИ (СВМ)

В разрабатываемом виртуальном музее истории информатики в Сибири применяются вышеперечисленные методы и технологии АГ. Виртуальный музей создается в виде информационно-поисковой, справочной и обучающей адаптивной гипермедиа-системы, доступной в интернете.

База данных музея поддерживает хранение и обработку информации о следующих объектах: публикациях, документах архива, проектах, ученых-информатиках, коллективах, событиях, конференциях и вычислительной технике. Все вышеперечисленные объекты являются «экспонатами» виртуального музея. Экспонаты, объединенные по тематическому, хронологическому или типологическому критерию, составляют «экспозицию» или «экскурсию». Экспозиции составляют «зал экспозиций», а экскурсии — «зал экскурсий». Оба этих зала являются «открытыми», т.е. доступными для просмотра всеми пользователями музея.

В музее также имеются запасники — залы, доступные только для зарегистрированных пользователей: библиотека, архив, хроника событий, зал ученых-информатиков, зал коллективов, зал проектов, зал вычислительной техники, зал конференций, зал новых поступлений и зал подготовки экспозиций и экскурсий. В библиотеке собраны книги, монографии, сборники статей, учебные и методические пособия, статьи из научных журналов, тезисы конференций и т.д. Архив представляет собой совокупность текстовых, графических, звуковых и видео-материалов. Хроника событий включает описания наиболее выдающихся событий из истории развития информа-

тики в Сибири. Зал информатиков содержит информацию о наиболее выдающихся ученых-информатиках, включая биографии, основные печатные труды и достижения, фото и пр. В зале коллективов содержатся данные о коллективах: группах, лабораториях и институтах. В зале проектов размещены данные о проектах, создаваемых в рамках работ по информатике (темы, системы). В зале вычислительной техники расположены экспонаты, имеющие отношение к вычислительной технике, которая использовалась и разрабатывалась с начала создания Сибирского отделения Академии наук. Зал конференций содержит информацию о различных научных мероприятиях. Новые экспонаты, добавляемые пользователями музея, помещаются в зал новых поступлений. В зале подготовки экспозиций и экскурсий размещаются экспозиции и экскурсии, создаваемые пользователями музея.

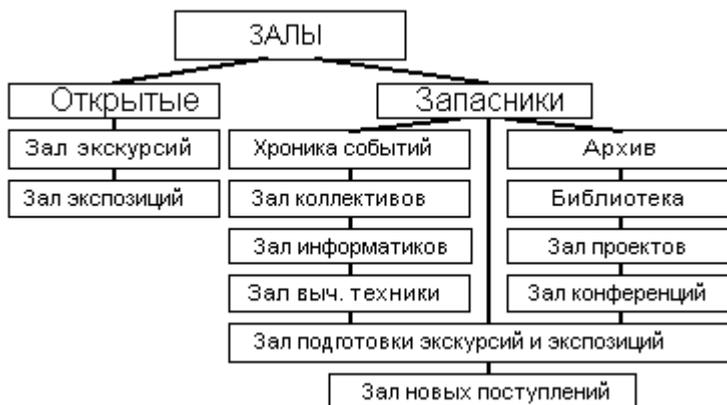


Рис. 5. Структура виртуального музея

Интерфейс музея разрабатывается с учетом его использования различными категориями пользователей. Все пользователи музея подразделяются на две основные категории: незарегистрированные пользователи («посетители») и зарегистрированные пользователи («специалисты»), различающиеся по уровню доступа к информационным ресурсам. «Посетители» могут просматривать только «открытые» залы (залы экскурсий и экспозиций), а «специалисты» — все залы, включая «запасники». «Специалисты» делятся на две группы: группу «простых специалистов» («волонтеры», «экскурсоводы» и «экспозиторы»), работающих только в зале новых поступлений, и группу «музейных работников» («библиотекари», «архивариусы», «хронологи» и т.д.), имеющих права на ввод и редактирование соответствующих

ресурсов музея, во главе этой группы находится главный администратор музея.

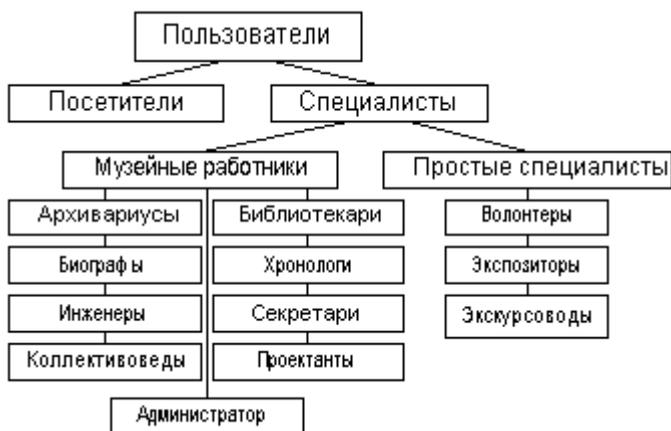


Рис. 6. Категории пользователей музея

К настоящему времени спроектирован и реализован гипермедиа-интерфейс для информационного наполнения музея: просмотра, поиска, ввода и редактирования данных обо всех вышеперечисленных объектах, а также механизм для их связывания. Реализован интерфейс для регистрации и аутентификации пользователей музея и ведения электронной конференции пользователей.

Для адаптивного представления информации в виртуальном музее предполагается использовать такие методы адаптивного представления информации, как дополнительные, предварительные объяснения и сортировка, а также такие методы адаптивной навигационной поддержки, как полное руководство, сортировка ссылок, сокрытие ссылок, аннотирование ссылок и генерирование ссылок.

3.1. Представление знаний предметной области (ПО)

3.1.1. Онтология и модель ПО

Для описания и концептуального структурирования ПО музея был проведен онтологический анализ с целью создания онтологии ПО. Онтологии были первоначально разработаны в области искусственного интеллекта, чтобы облегчить коллективное и многократное использование знаний. Поз-

же, онтологии использовались для интеллектуального поиска знаний в Web как инструмент моделирования семантической информации (метаданных), используемой для аннотирования веб-документов. В данном случае, онтологии используются, чтобы обеспечить максимальную гибкость при представлении знаний ПО. Также они являются основным элементом для достижения разделения представления и содержания, что необходимо для обеспечения адаптивного представления информации.

Основной характерной чертой онтологического анализа является разделение совокупности знаний ПО на классы объектов и определение их онтологий, т.е. совокупности их фундаментальных свойств и отношений. В ходе проведенного онтологического анализа были исследованы объекты, составляющие ПО музея, их характеристики и взаимосвязи. В результате была разработана онтология ПО, или же совокупность словаря терминов, их определений и взаимосвязей между ними. В терминах онтологии в любой системе существуют две основные категории предметов, такие как сами объекты, составляющие систему, и взаимосвязи между этими объектами, характеризующие состояние системы. Понятие взаимосвязи однозначно описывает, или, другими словами, является точным дескриптором зависимости между объектами системы, а термины являются, соответственно, точными дескрипторами самих объектов.

Разработанная онтология состоит из множества классов, представляющих совокупности однородных объектов ПО. Онтологические классы определены для каждого из типов объектов (экспонатов), хранящихся в виртуальном музее. Это, соответственно: классы публикаций, документов архива, проектов, ученых-информатиков, коллективов, событий, конференций, вычислительной техники, экскурсий и экспозиций. Экземпляры вышеперечисленных классов представляют знания об экспонатах музея соответствующих типов.

Для каждого из вышеперечисленных классов определен набор атрибутов (характеристик) и отношений (взаимосвязей), характерный для данного класса объектов. Структуру экземпляров каждого класса представляет соответствующий протофрейм, имеющий следующий вид:

(Имя фрейма:

Имя слота 1 (значение слота 1)

Имя слота 2 (значение слота 2)

.....

Имя слота K (значение слота K)).

Слоты соответствуют атрибутам и отношениям, определенным для данного класса. При конкретизации фрейма ему и слотам присваиваются кон-

кретные имена, и происходит заполнение слотов. Таким образом, из протофреймов получают фреймы-экземпляры (экземпляры класса).

Каждый класс (экземпляр класса) имеет следующие основные атрибуты: уникальный универсальный идентификатор (*uuid*), название (имя), дату, краткое и полное описание, а также служебные данные (пользователь, добавивший объект, дата добавления, права модификации и т.д.). Кроме основных атрибутов, классы могут иметь дополнительные атрибуты и отношения, характерные для объектов данного класса. Так, например, класс «ученых-информатиков» имеет такие дополнительные атрибуты, как биография, фотография, и такие отношения, как список публикаций, список проектов и т.д.

Например, структура экземпляров класса «ученых-информатиков», записанная в форме протофрейма, будет иметь следующий вид:

(Ученые-информатики:

Идентификатор (значение слота 1)

ФИО (значение слота 2)

Дата рождения (значение слота 3)

Образование (значение слота 4)

Ученые степени (значение слота 5)

Ученые звания (значение слота 6)

Кандидатская диссертация (значение слота 7)

Докторская диссертация (значение слота 8)

Научные интересы (значение слота 9)

Текст биографии (значение слота 10)

Фотография (значение слота 11)

Адрес (значение слота 12)

E-mail адрес (значение слота 13)).

Если в качестве значений слотов использовать конкретные данные, то получится фрейм-экземпляр.

Для каждого онтологического класса определена одноименная таблица в базе данных (БД) музея, структура которой соответствует определенному протофрейму (поля таблицы — слоты протофрейма). Информация об экземплярах класса содержится в соответствующей таблице БД, в которой каждому экземпляру соответствует запись в таблице.

После определения онтологии модель ПО строится посредством создания фреймовой сети концептов и межконцептных отношений с использованием классов и отношений, определенных в онтологии. Концепты и межконцептные отношения ПО формируются путем создания фреймов-экземпляров, т.е. заданием значений слотов в протофреймах, и связывания

их друг с другом, используя концептуальный словарь, определенный онтологией. Таким образом, модель ПО является моделью третьего уровня (фреймовой моделью), которая состоит из взаимосвязанных концептов, имеющих внутреннюю структуру (представленную в виде фрейма).

Концепты ПО используются также для поддержания оверлейной модели знаний пользователя. Эта модель динамически обновляется, отражая прогресс в изменяющемся знании пользователя.

3.1.2. Модель представления

Разделение представления и содержания достигается путем задания модели представления: для каждого класса онтологии определены т.н. *шаблоны представления*. Шаблоны определяют, какие части (атрибуты и отношения) экземпляра конкретного класса должны быть включены в представление и в каком порядке, а также их визуальное представление и расположение. Шаблоны дополняются так называемыми *правилами представления*, которые отвечают за генерирование адаптивных конструкций представления.

Шаблоны представляют собой html-код, расширенный вставками РНР-инструкций, позволяющими вставлять в html-код различные управляющие операторы и обращаться к концептам МО (атрибутам и отношениям), которые требуется представить. Веб-представление для определенного концепта (экземпляра класса) динамически генерируется на основе шаблона представления для данного класса и информации о данном концепте, представленной в модели ПО. С помощью вставок РНР-инструкций в html-код во время генерации представления значения требуемых атрибутов и отношений автоматически извлекаются из БД и динамически подставляются в соответствующие места шаблона представления.

Для каждого класса объектов определены два типа шаблонов представления: шаблон для ввода и редактирования и шаблон для просмотра информации об объекте. Шаблон для ввода и редактирования представляет собой набор форм с полями для ввода в БД значений атрибутов и отношений. Шаблон для просмотра является непосредственно шаблоном представления, на его основе генерируется веб-представление экземпляров соответствующего класса. Оба типа шаблонов представляют информацию, разделенную на три части: общая, дополнительная и служебная. Общую информацию составляют такие атрибуты, как название (имя) объекта, дата, краткое описание и т.д. Дополнительная информация включает в себя, например, полное описание. Служебная информация содержит данные о пользователе, добавившем объект в БД, дату добавления, права изменения и т.д.

Что касается «полного описания» объекта, то оно тоже является своего рода шаблоном, но не predetermined в системе, а создаваемым самим пользователем для данного объекта. Шаблон создается пользователем в режиме on-line с помощью разработанного специального html-редактора. Редактор поддерживает стандартные функции форматирования текста, включения графической, аудио- и видео-информации. Имеется возможность предварительного просмотра текста, генерируемого по создаваемому шаблону, в процессе ввода и редактирования. Реализована функция автоматического и динамического включения информации (атрибутов и отношений) об объектах МО в представление. Реализована возможность выбора для каждого типа объектов соответствующих атрибутов, а также возможность указания способа подстановки (простая подстановка или гиперссылка).

Таким же образом создаются экспозиции, с тем лишь отличием, что их веб-представление имеет структуру, состоящую из набора связанных гипермедиа-страниц. Шаблоны для этих гипермедиа-страниц конструируются пользователем аналогичным образом с помощью того же html-редактора.

Таким образом, в системе используется третий метод (прямая связь) для организации связи между концептами модели ПО и гиперпространством системы (гипермедиа-страницами), т.е. гиперпространство строится непосредственно исходя из структуры ПО. Каждый концепт модели ПО представлен динамической гипермедиа-страницей или гипердокументом, которые генерируются налету, исходя из внутренней структуры концепта.

3.2. Моделирование пользователя

Для зарегистрированных пользователей поддерживается МП, состоящая из модели индивидуальных сведений, модели категорий, модели предпочтений и модели знаний.

Статическая *модель индивидуальных сведений* включает информацию об идентификационных данных пользователя (идентификатор и пароль для входа в систему), ФИО, почтовом и e-mail адресах. Эта информация хранится в соответствующей таблице БД и извлекается во время авторизации пользователя.

Модель категорий моделирует права доступа пользователя к информационным ресурсам музея. Она реализуется с помощью статической стереотипной модели: для каждой категории пользователей музея определен одноименный стереотип, характеризующийся определенными значениями атрибутов. В качестве атрибутов используются имена типов ресурсов БД

(публикации, проекты, события и т. д.), значениями атрибутов являются права для доступа к соответствующему типу ресурсов (просмотр, добавление, изменение, удаление и их комбинации). В таблице приведен набор стереотипов пользователей с указанием прав доступа к различным типам ресурсов, где просмотр обозначен как «П», добавление — «Д», изменение — «И» и удаление — «У».

	Экс-курсии	Экспо-зиции	Публи-кации	Док-ты архива	Проек-ты	Ин-форма-тики	Кол-лекти-вы	Собы-тия	Конфе-ренции	Выч. техни-ка
Посетители	П	П								
Специалисты	П	П	П	П	П	П	П	П	П	П
Волонтеры	П	П	ПД	ПД	ПД	ПД	ПД	ПД	ПД	ПД
Экскурсоводы	ПДИ	П	П	П	П	П	П	П	П	П
Экспозиторы	П	ПДИ	П	П	П	П	П	П	П	П
Библиотекари	П	П	ПДИ	П	П	П	П	П	П	П
Архивариусы	П	П	П	ПДИ	П	П	П	П	П	П
Проектанты	П	П	П	П	ПДИ	П	П	П	П	П
Биографы	П	П	П	П	П	ПДИ	П	П	П	П
Коллективове-ды	П	П	П	П	П	П	ПДИ	П	П	П
Хронологи	П	П	П	П	П	П	П	ПДИ	П	П
Секретари	П	П	П	П	П	П	П	П	ПДИ	П
Инженеры	П	П	П	П	П	П	П	П	П	ПДИ
Администра-торы	ПДИУ	ПДИУ	ПДИУ	ПДИУ	ПДИУ	ПДИУ	ПДИУ	ПДИУ	ПДИУ	ПДИУ

При входе в систему каждый пользователь идентифицируется как принадлежащий к определенному стереотипу (категории), исходя из структуры его идентификатора, и при помощи модели категорий получает соответствующие права доступа к информационным ресурсам.

Модель предпочтений моделирует различные предпочтения пользователей, в частности, способ представления информации (использование только текста, графики, звука, видео и т.д.). Она реализуется с помощью статической стереотипной модели, атрибутами которой являются вышеперечисленные способы представления информации, а значениями — истина или ложь.

Модель знаний используется для моделирования знаний ПО пользователя и реализуется посредством оверлейной модели. Система получает данные для модели, отслеживая историю навигации пользователя. Знание пользователя представляется как подмножество знания эксперта ПО: для каждого концепта МО определен набор пар атрибут–значение, где в качестве ат-

рибутов используются атрибут "чтение" (прочитан, не прочитан) и атрибут "знание" концепта (изучен, не изучен). Модель знаний для каждого пользователя представляется в виде следующего вектора, содержащего информацию о множестве прочитанных и изученных концептов ПО:

(users_uid, concept1, read-value1, known-value1, ... conceptN, read-valueN, known-valueN),

где users_uid — идентификатор пользователя, concept — идентификатор концепта, read-value — значение чтения, known-value — значение знания. По мере просмотра пользователем информации модель автоматически обновляется.

ЗАКЛЮЧЕНИЕ

В статье представлен проект создания виртуального музея истории информатики в Сибири, разрабатываемого в виде информационно-поисковой, справочной адаптивной гипермедиа-системы, доступной в интернете. Работа над данным проектом ведется коллективом сотрудников ИСИ СО РАН, ИМ СО РАН и НГУ при финансовой поддержке РФНФ (02-05-12010).

В статье представлены методы и технологии АГ, описаны архитектура АГС и ее основные компоненты (модель предметной области, модель пользователя и модель адаптации), рассмотрены вопросы моделирования предметной области и моделирования пользователя. Описанные методы и технологии адаптивной гипермедиа, моделирования предметной области и моделирования пользователя были использованы при создании данного виртуального музея. В статье обсуждаются вопросы представления знаний ПО, построения онтологии и модели ПО, а также разработки модели пользователя.

СПИСОК ЛИТЕРАТУРЫ

1. **Волянская Т.А.** Виртуальный музей истории информатики в Сибири // Материалы Междунар. конф. молодых ученых по математическому моделированию и информационным технологиям. — Новосибирск, 2002. — С. 49.
2. **Волянская Т.А.** Методы адаптации гипермедиа и их применение при создании виртуального музея истории информатики в Сибири // Материалы XL Междунар. науч. студенческой конф. «Студент и научно-технический прогресс». — Новосибирск, 2002. — С. 173–174.

3. **Волянская Т.А.** Методы и технологии адаптивной гипермедиа // Современные проблемы конструирования программ. — Новосибирск, 2002. — С. 38–68 .
4. **Касьянов В.Н., Несговорова Г.П., Волянская Т.А.** Виртуальный музей истории информатики в Сибири // Электронные изображения и виртуальные искусства. — Киев, 2002. — С. 242–250.
5. **Касьянов В.Н., Несговорова Г.П., Волянская Т.А.** Виртуальный музей истории информатики в Сибири // Современные проблемы конструирования программ. — Новосибирск, 2002. — С. 169–181.
6. **Brusilovsky P.** Adaptive Hypermedia // User Modeling and User-Adapted Interaction. — 2001. — Vol 11. —P. 87–110.
7. **Brusilovsky P.** Adaptive hypermedia, an attempt to analyze and generalize // Lect. Notes. Comput. Sci. — 1996. — Vol. 1077. — P. 288–304.
8. **Brusilovsky P.** Methods and techniques of adaptive hypermedia // User Modeling and User-Adapted Interaction. — 1996. — Vol 6. — P. 87–129.
9. **Brusilovsky P., Cooper D. W.** Domain, Task, and User Models for an Adaptive Hypermedia Performance Support System // Proc. of 2002 Internat. Conf. on Intelligent User Interfaces. — San Francisco, CA, 2002. — P. 23–30.
10. **De Bra P.** Adaptive Hypermedia on the Web: Methods, techniques and applications // Proc. of the AACE WebNet'98 Conf. — Orlando, Fl., 1998. — P. 220–225.
11. **De Bra, P., Brusilovsky P., Houben G.-J.** Adaptive Hypermedia: From Systems to Framework // ACM Computing Surveys. — 1999. — Vol. 31, N 4. — Article N 12.
12. **De Bra P., Houben G.J., Wu H.** AHAM: A Dexter-based Reference Model for Adaptive Hypermedia // Proc. of the ACM Conf. on Hypertext and Hypermedia. — Darmstadt, Germany, 1999. — P.147–156.
13. **Kobsa A.** User Modeling: Recent Work, Prospects and Hazards // Adaptive User Interfaces: Principles and Practise. — Amsterdam, North Holland Elsevier. — 1993.
14. **Kobsa A.** User Modeling in Dialog Systems: Potentials and Hazards // AI & Society. — 1990. — Vol. 4. — P. 214–240.
15. **Wu H., De Bra P., Aerts A., Houben G.J.** Adaptation Control in Adaptive Hypermedia Systems // Lect. Notes. Comput. Sci. — 2000. — Vol. 1892. — P. 250–259.
16. **Wu H., De Kort E., De Bra P.** Design Issues for General-Purpose Adaptive Hypermedia Systems // Proc. of the ACM Conf. on Hypertext and Hypermedia. — Aarhus, Denmark, 2001. — P. 141–150.
17. **Wu H., Houben G.J., De Bra P.** Supporting User Adaptation in Adaptive Hypermedia Applications // On-line Conf. and Informatiewetenschap 2000. — De Doelen, Rotterdam, 2000.