УДК 519.17:004.434:004.94

ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ СЛОЖНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНТЕРНЕТ УЗЛА

А.Ю. Меренков

Запорожская государственная инженерная академия
69006, Запорожье, проспект Ленина, 226,
тел. (8-067) 614 4983, merenkov@inbox.ru

Представлена модель иерархической структуры программного обеспечения Интернет узла. Проведен анализ и оценка возможности применения визуального моделирования для задач этого класса. Приведены примеры визуального моделирования структуры системы. Применение иерархической структуры данных позволит автоматизировать процесс создания пользовательского интерфейса сложных программных систем.

This report presents a model of hierarchical database structure, analysis and evaluation of possibility to utilize visual modeling language to solve this class of problems, modeling examples of system's structure. Usage of hierarchical database structure allows to automate the construction of user interface in complex programming systems.

Введение

Объектно-ориентированное визуальное моделирование – молодая и бурно развивающаяся область компьютерной инженерии. В начале 90-х годов по этой теме появилось много фундаментальных работ. Наибольшее влияние на формирование этой области оказали исследования Г. Буча, И. Джакобсона, Д. Рэмбо, П. Коуда, Д. Харела, Б. Селика и др., усилиями которых был создан стандарт в этой отрасли – язык UML (Unified Modeling Language) [1].

Актуальность этого направления обнаруживает проблемы практического применения визуального моделирования.

Главная проблема заключается в принципиальной трудности адекватной формализации процесса создания программного обеспечения (ПО). Программирование является в большой степени творчеством [2]. Буч в своей знаменитой монографии [3] часто повторяет, что его метод – не поваренная книга, имея в виду уникальность каждого проекта. Объектно-ориентированное визуальное моделирование призвано понизить сложность создания ПО, повысить удельный вес и качество анализа и проектирования. Однако, столкнувшись с проблемой формализацию процесса разработки ПО, методологи фактически переадресовали ее создателям CASE-пакетов.

Таким образом, остается неясным принципиальное положение диаграмм при создании ПО (относительно проектной документации, программного кода и т.д.), и вследствие этого отсутствуют общие концепции связи между этими представлениями ПО (прежде всего, с программным кодом, в связи с чем возникает много необоснованных обещаний генерировать код полностью по диаграммам).

Прямым следствием неясного положения визуального моделирования в процессе разработки ПО является проблема поддержания итеративности процесса разработки системы. Сюда относится и проблема семантических разрывов между моделями, и контроль целостности информации в разных моделях, и односторонняя связь с программным кодом и т.д.

Теоретические разработки в области визуального моделирования делают упор на формальные модели и общие проблемы компьютерной инженерии. Конкретные решения по использованию визуального моделирования предлагаются в рамках CASE-пакетов, но ознакомиться с ними можно, только купив данный программный продукт. Таким образом, налицо разрыв между наукой и практикой. В докладе рассматривается важный частный случай ПО, при разработке которого применимы, помимо компонентного подхода, другие средства визуального моделирования – построение иерархий объектов и классов, теория графов. Это программное обеспечение Интернет узлов, в разряд которых попадают, например, системы электронной коммерции, мобильные сервисы на основе протокола WAP. Создание высокоуровневых средств спецификации таких систем с возможностью автоматической кодогенерации является актуальной задачей, несмотря на имеющиеся в этой области разработки[4-12].

Сложные программные системы

Сложные системы часто являются иерархическими и состоят из взаимозависимых подсистем, которые в свою очередь также могут быть разделены на подсистемы. Саймон отмечает: "тот факт, что многие сложные системы имеют почти разложимую иерархическую структуру, является главным фактором, позволяющим нам понять, описать и даже "увидеть" такие системы и их части" [5]. В самом деле, скорее всего, можем понять лишь те системы, которые имеют иерархическую структуру. Важно осознать, что архитектура сложных систем складывается и из компонентов, из иерархических отношений этих компонентов. Речтин отмечает: "Все

системы имеют подсистемы, и все системы являются частями более крупных систем... Особенности системы обусловлены отношениями между ее частями, а не частями как таковыми" [1].

Выбор, какие компоненты в данной системе считаются элементарными, относительно произволен и в большой степени оставляется на усмотрение исследователя. Низший уровень для одного наблюдателя может оказаться достаточно высоким для другого. Саймон называет иерархические системы разложимыми, если они могут быть разделены на четко идентифицируемые части, и почти разложимыми, если их составляющие не являются абсолютно независимыми.

Особенности программного обеспечения Интернет узлов

Различие в подходах к составлению спецификации программной системы, моделированию и реализации полученных моделей в программном коде приводит к проблемам, характерным для сложных программных систем. Для программного обеспечения Интернет узлов особенно остро стоит проблема совместимости исполняемой среды. Это обусловлено самой структурой системы, основанной на взаимосвязи:

База данных – Сервер – Клиент.

База данных – это одна из реализаций реляционной СУБД. Сервер – это специализированный комплекс аппаратных и программных средств предоставляющий сервис обмена данными по протоколу HTTP (Hyper Text Transfer Protocol). Сервер также является средой исполнения программ. Клиент - это программный и аппаратный комплекс для запросов на Сервер и представления информации пользователю в общепринятом виде.

В процессе развития языков программирования, наиболее жизнеспособным оказался аппаратно-независимый подход к построению интерпретаторов. Таким образом, можем допустить, что программа, написанная на одном из скриптовых языков, будет выполняться одинаково на большинстве Интернет серверов. Такое единообразие обеспечивается стандартной средой исполнения и наличием совместимых интерпретаторов для разных платформ.

Слабым местом здесь является архитектура Клиента. Несмотря на наличие стандартов для гипертекста, существует множество реализаций обозревателя Интернета. Каждый из них имеет расширения стандартов, пытаясь завоевать рыночную нишу, опережая конкурентов. Особо следует отметить мобильные устройства. Для них созданы специальные стандарты передачи данных, не совместимые с обычным гипертекстом [12, 13]. Ограниченность ввода/вывода для этих устройств требует выработки специального подхода для построения интерфейса.

Пример визуального моделирования программного обеспечения Интернет узла

Рассмотрим пример моделирования новостного сайта. Для каждого раздела составим визуальные модели для прецедентов использования, структуры данных и программного обеспечения.

Персональная страничка пользователя. Раздел предназначен для выборки персональной информации о зарегистрированных пользователях.

Прецеденты использования. Возможные действия пользователя показаны на диаграмме прецедентов (рис. 1).

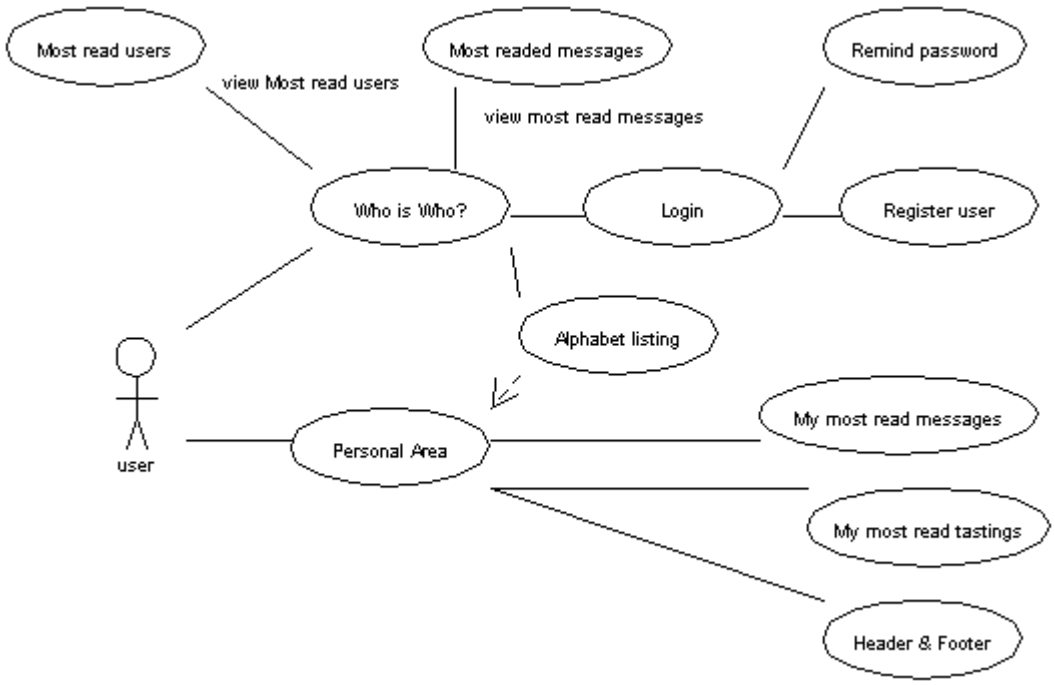

Рис. 1. Прецеденты использования

В данном разделе пользователь может:

- просмотреть самых читаемых пользователей и самые читаемые сообщения; войти в систему и перейти к своему *персональному разделу*.

Список самых читаемых пользователей выглядит, как показано на рис. 2:

Who's who?

Most readed users
Dr.Puro
Sender
Westwoodian
Invited
RaiUnderSiege
Cigarbits
Linked
Romeo
Cigarfest
candygar
si-gar
EBH NewsRoom
R&J
Alo'gar
Allende

Рис. 2. Список пользователей

Формируется такой список с помощью следующего кода:

```
//выбираем таблицу в зависимости от языка
if ($lang=="english") {$table="postings_eng";}
if ($lang=="spanish") {$table="postings";}
if ($lang=="german") {$table="postings_ger";}
$query="select * from $table  where (((forumid=1) or  (forumid=2) or
(forumid=3) or  (forumid=4) or  (forumid=5) or (forumid=6) or  (forumid=7) or  (forumid=8))
and  (area=\"new\"))  group by author  order by hits desc".
```

Выбираются новые сообщения из нужных форумов и группируются по автору, затем сортируются по убыванию по полю *hits*.

Аналогично отображается список самых читаемых сообщений (рис. 3).

Most readed messages

- War´s end cigar by Dr.Puro
- Davidoff: worths what it costs? by Dr.Puro
- Cigars an cancer by Sender
- Cigars an cancer by Westwoodian
- No water in my humidor by Invited
- Ageing cigs by RaiUnderSiege
- Ageing cigs by Cigarbits
- No water in my humidor by Linked
- Live is good by Linked
- War´s end cigar by Romeo
- Ageing cigs by Invited
- Ageing cigs by Cigarfest
- Cigars an cancer by candygar
- Cigars an cancer by Invited
- Traveling to Spain by Linked
- Live is good by si-gar
- Puros Indios by Cigarfest
- What if embargo goes down? by si-gar
- What if embargo goes down? by Linked
- Traveling to Spain by Invited 28-03-02 13:30

Рис. 3. Список сообщений

Формируется такой список с помощью следующего кода:

```
//выбираем таблицу в зависимости от языка
```

```
if ($lang=="english") {$table="postings_eng";}
  if ($lang=="spanish") {$table="postings";}
  if ($lang=="german") {$table="postings_ger";}
$query="select * from $table   where (((forumid=2) or  (forumid=3) or                    (forumid=4) or
(forumid=5) or  (forumid=6) or  (forumid=7) or (forumid=8)) and  (area=\"new\"))  order by hits desc";
```

Так же можно просмотреть информацию о пользователях по алфавиту (рис. 4).

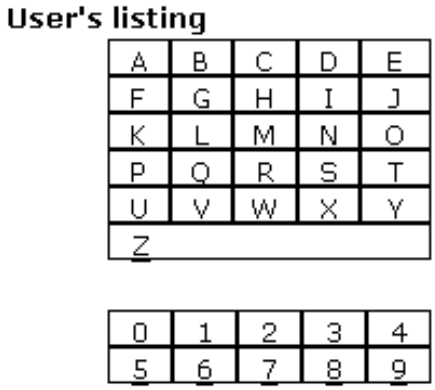

Рис. 4. Алфавитный список

Для того, чтобы попасть в свой персональный раздел, пользователь должен выполнить вход в систему (рис. 5).

Рис. 5. Форма авторизации

Возможна конфигурация заголовков и подписей для своих сообщений в форумах (рис. 6).

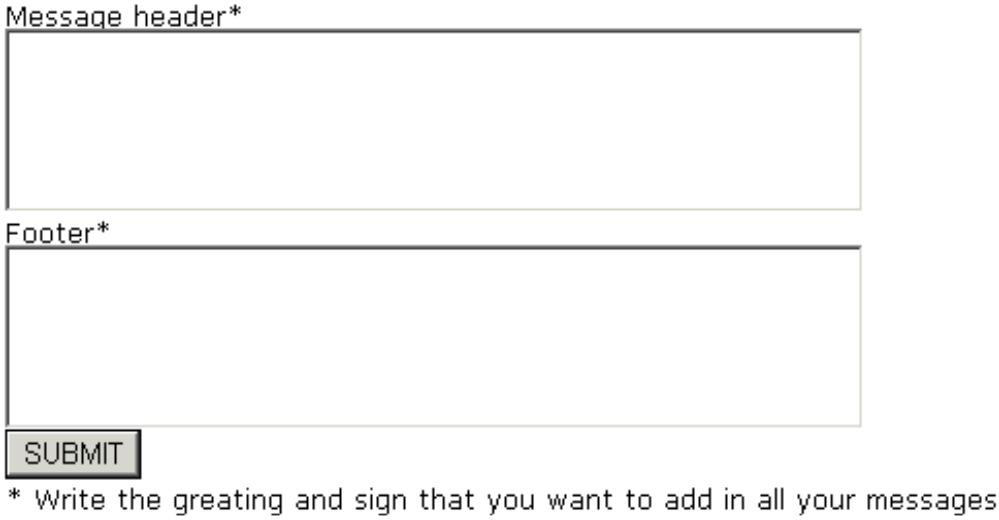

Рис. 6. Настройка сообщений

Форум дебатов. Данный раздел предназначен для обсуждения публикаций и товаров.
Прецеденты использования (рис. 7).

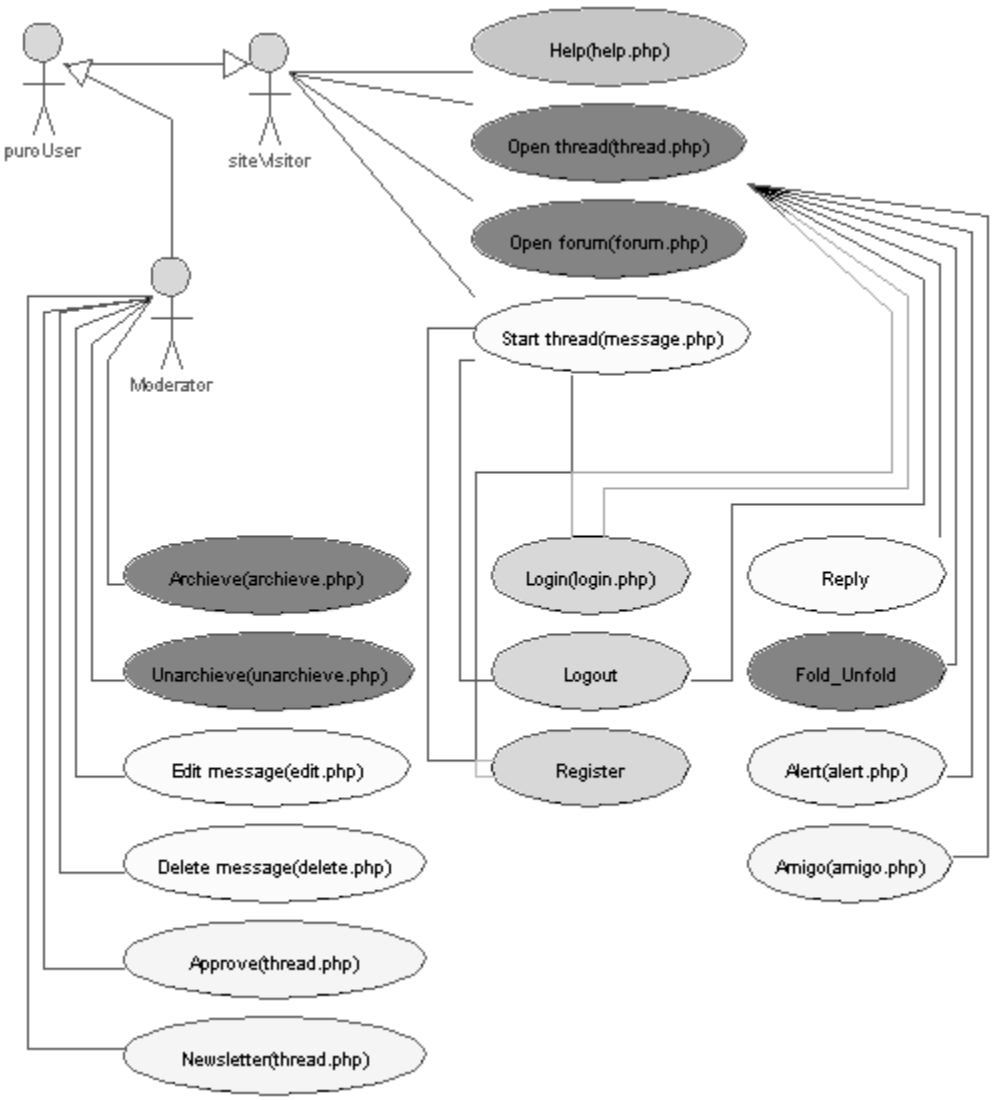

Рис. 7

Программное обеспечение (рис. 8).

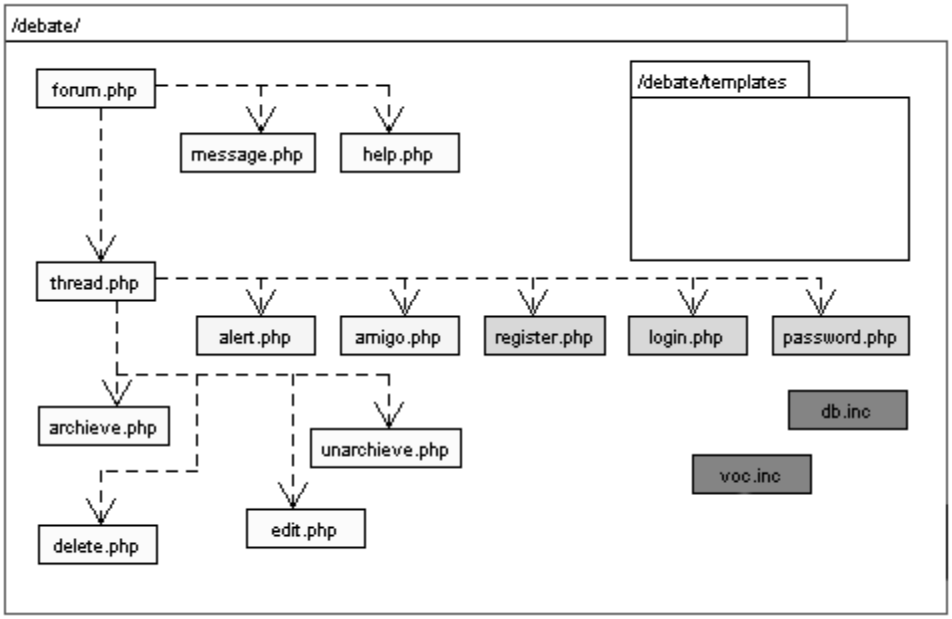

Рис. 8

Схема данных (рис. 9).

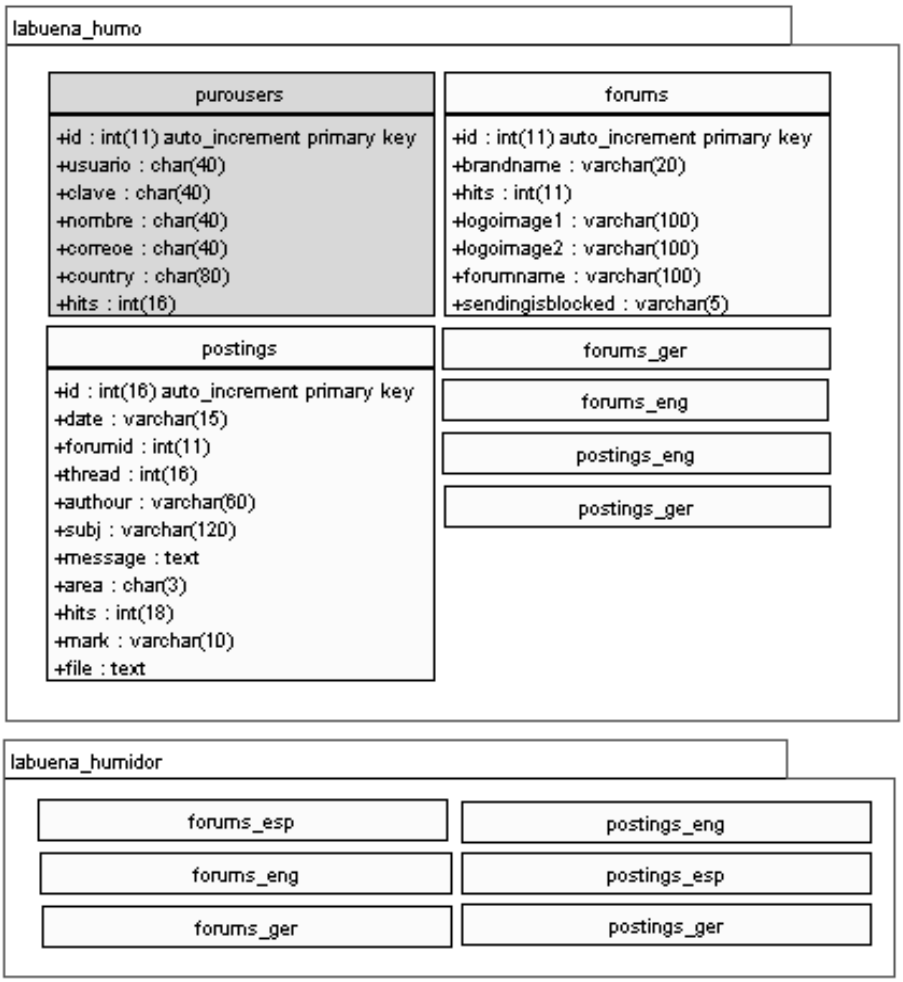

Рис. 9. Прецеденты использования

Статьи. Раздел предназначен для отображения статьи, выбранной в разделе News, а так же информации связанной со статьёй.

Возможности пользователя при работе с данным разделом представлены на диаграмме прецедентов на рис. 10.

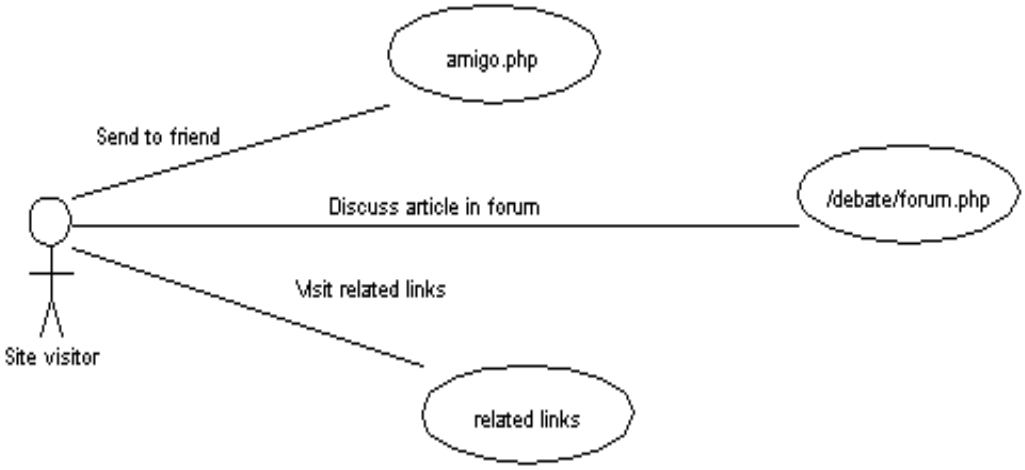

Рис. 10. Прецеденты использования

В данном разделе мы можем прочитать статью. На рис. 11 изображено начало статьи:

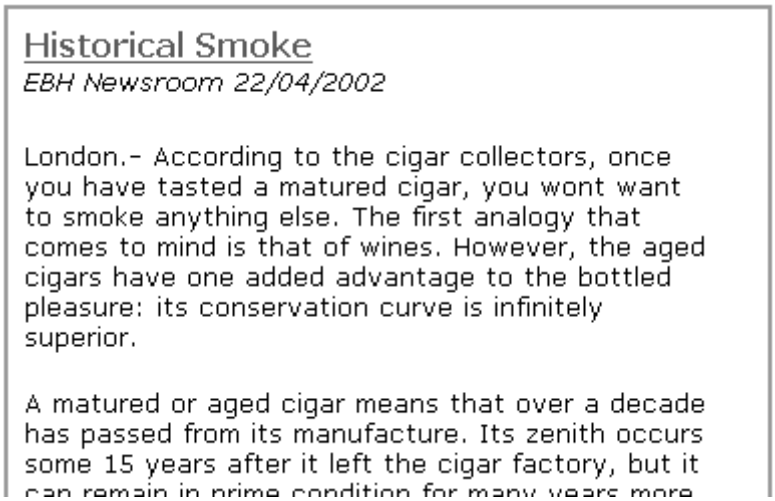

Рис. 11

Выбока статьи и связанной с ней информации (related links) происходит таким образом:

```
$table="HabanoArticles";
$query="select    languageid, title, fontsize, sectionid, area,
                  author,releasedate,text,  link1name,link2name,link3name,
                  link4name,link5name,link6name,  link1value,link2value,link3value,
                  link4value,link5value,link6value,  pollid,threadid,keywords,usethread,
                  usepoll,hits
          from $table
         where id=$id";
```

Как видно из запроса, выбираются ссылки по статье – (link%[name | value]), т.е. название и адрес ссылки. Пустые ссылки не отображаются.

В конце каждой статьи есть возможность отправить эту статью другу:

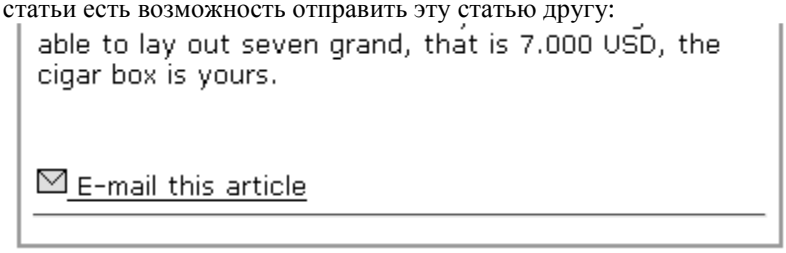

Рис. 12

При переходе по ссылке отрывается окно для отправки статьи (рис. 13).

Рис. 13

В поле TO указывается адрес получателя.
В поле FROM адрес отправителя.
В поле Comment можно указать свой комментарий к статье.
На рис. 14 показаны типы отравки сообщения: только ссылка; заголовок и ссылка; статья целиком.

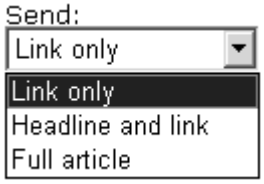

Рис. 14. Варианты отправки

Статью можно обсудить в форуме, посвященном данной статье:
На странице со статьёй показываются последние сообщения в связанном с ней форуме (рис. 15).

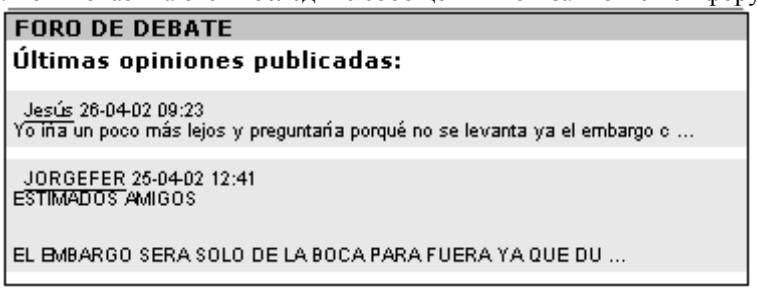

Рис. 15.

Так же можно принять участие в голосовании связанном с данной статьей:

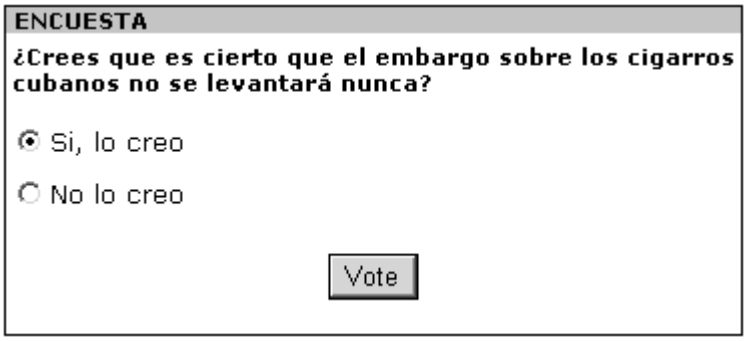

Рис. 16

Выводы

Структуре программного обеспечения Интернет узлов присуща сложность, которая нередко превосходит возможности человеческого разума. Сложные структуры часто принимают форму иерархий; полезны обе иерархии: и классов, и объектов. Сложные системы обычно создаются на основе устойчивых промежуточных форм. Задача разработчиков программных систем - создать у пользователя разрабатываемой системы иллюзию простоты. В то же время должна достигаться четкая взаимосвязь между этапами проектирования, моделирования и использования системы. Применение визуального моделирования на всем жизненном цикле программного обеспечения позволит преодолеть сложность систем и обеспечить управляемость процесса разработки программ.

1. *OMG* Unified modeling language specification (draft). Version 1.3R. http://www.rational.com/uml. 1999.
2. *Ф.П.Брукс мл.* Как проектируются и создаются программные комплексы. Мифический человеко-месяц. - М. – 1979. - 150 с.
3. *Booch. G.* Object-Oriented Analysis And Design With Application, second edition. The Benjamin Publishing Company, Inc. - 1994. - 589 с.
4. *Чеппел Д.* Технологии ActiveX и OLE. М.: Издательский отдел «Русская редакция» ТОО «Channel Trading Ltd.», 1997.- 320 с.
5. *Rumbaugh J., Jacobson I., Booch G.*. The Unified Modeling Language Reference Manual. Addison-Wesley, 1999.- 550 с.
6. *Selic B., Gullekson G. Ward P.T.* Real-Time Object-Oriented Modeling. John Wiley & Sons. Inc. - 1994. - 525 с.
7. *ITU Recommendation Z.100:* Specification and Description Language. 1993. 204 с.
8. *Harel D., Politi M.* Modeling Reactive Systems with Statecharts: state machine approach. McGraw-Hill, 1998. - 258 с.
9. *Гома Х.*, Uml. Проектирование систем реального времени, распределенных и параллельных приложений, ДМК, 2002.
10. *Липаев В.В.*, Надежность программных средств, СИНТЕГ, 1998.
11. *M. Ohrt*, A. Zmievsky. Smarty manual. http://smarty.php.net.
12. *WAP Forum*, Wireless Application Environment Specification, April 30, 1998.
13. *WAP Forum*, Wireless Application Protocol Architecture Specification, April 30, 1998.