

Руслан Богатырев

## Технология Curl и концепция X Internet

Источник: Мир ПК, #09/2001.

**В начале ноября 2001 г. в Бостоне (США) пройдет международный форум, посвященный новой концепции X Internet. На нем запланированы три ключевых выступления: президента Microsoft, директора Sun Labs и президента компании Curl (бывшего президента IBM Instruments). В центре обсуждения будет новый сценарный язык Curl, которому специалисты прочат большое будущее.**

В основе технологии Curl лежит вполне прозрачная идея – резко снизить нагрузку на Internet за счет перехода от передачи данных к передаче команд. Это чем-то напоминает идею Java с той принципиальной разницей, что теперь для обмена информацией используется сценарный язык, а не специальные протоколы, реализованные в универсальном языке.

Технология Curl весьма созвучна новой концепции X Internet, которую предложил в конце 2000 г. председатель совета директоров аналитической компании Forrester Research Джордж Колони [1]. X Internet – это исполняемый (eXecutable) Internet, предполагающий замену пассивной навигации по Web-страницам активным общением. Это ведет к интенсификации сетевых сервисов. Данные все больше будут вытесняться программами, и соответственно непременно изменится соотношение роли Web-серверов и браузеров. Все большее развитие получит схема peer-to-peer (равноправного обмена). В некотором смысле это вступает в противоречие с другой концепцией – Internet нового поколения (Next Generation Internet), выдвинутой корпорацией Microsoft в 2000 г. Суть ее в том, чтобы перейти от разрозненных сайтов и устройств, подключенных к Internet, к комплексу компьютеров, устройств и сервисов, которые работают совместно для достижения качественно более лучших, интегрированных решений. Платформа Microsoft .NET как раз-таки и реализует эту концепцию. Вряд ли стоит говорить о прямом противостоянии этих двух точек зрения, скорее концепции Forrester и Microsoft начнут взаимно трансформироваться с учетом изменения ситуации.

Компания Curl позиционирует новую технологию на четыре рынка: финансовые дома, электронный бизнес, управление информационным продуктом, индустрия развлечений. В настоящее время у компании Curl два крупных клиента: Siemens и British Telecom. Но первая волна публикаций [2, 3] вызвала лавинообразный интерес к технологии, так что число клиентов неизменно будет увеличиваться.

Технология Curl нашла свое воплощение в трех составляющих: одноименном сценарном языке, подключаемом модуле Surge (для расширения функциональности браузера) и инструментальной среде Surge Lab. Модуль Surge имеется пока только для Windows (Windows XP, Windows 9x/Me, Windows NT/2000). В стадии подготовки версии для Mac OS X и Linux.

В контексте технологии Curl на клиентской стороне должен вестись разбор компактного кода, передаваемого серверной стороной по каналам связи. С этой целью придется добавлять «интеллекта» браузерам. Технология Java с подобной задачей не справилась. Почему должна справиться технология Curl? Авторы языка считают, что сценарный язык лучше подойдет для этой цели, нежели язык традиционный.

Сценарный язык Curl разработан сравнительно недавно (проект стартовал в октябре 1995 г. и спонсировался агентством DARPA). Колыбелью грядущей революции в Web-программировании стала Лаборатория компьютерных наук Массачусетского технологического института (MIT). В феврале 1998 г. руководители проекта Стефен Уард и Майкл Дертусос вместе с отцом-основателем современного Internet Тимом Бернерс-Ли создали компанию Curl (<http://www.curl.com>). Первая коммерческая версия языка увидела свет весной 2001 г. – спустя ровно шесть лет после появления Java.

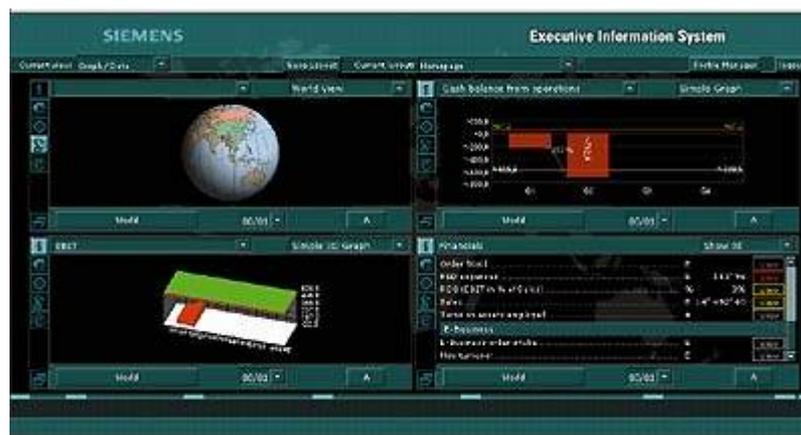


Рис. 1. Электронная панель информационной системы компании Siemens.

Curl в переводе с английского означает «локон, завиток». Такое название было выбрано по той причине, что конструкции языка заключаются в фигурные (по-английски «кудрявые») скобки. Curl позиционируется авторами как язык систем плавного перехода (gentle slope system) и язык программирования информационного продукта (content language). За счет вкрапления в обычный текст различных конструкций (по аналогии с тегами языков разметки и командами языка TeX), начиная от элементарных операций форматирования и заканчивая сложными командами в рамках ООП, Curl облегчает непрограммистам управление мощными средствами программного окружения.

Язык Curl ориентирован одновременно на программирование и на формирование информационного продукта (content). В Curl любой символ, не заключенный в фигурные скобки, трактуется как обычный текст и непосредственно отображается на экране. Все, что заключено в фигурные скобки, трактуется как выражение на языке Curl. Такие скобки могут быть вложенными. Язык оперирует понятиями значений (скалярные типы, в частности целые и вещественные числа, байты и т.п.) и объектов (ссылки на значения).

С точки зрения объектной модели Curl имеет много общего с языком Java. Принципиальная разница между Curl и Java состоит в том, что Java является языком объектно-ориентированного программирования (ООП), тогда как Curl сочетает в себе ООП с форматированием текстов и созданием сценариев. В отличие от Java язык Curl поддерживает множественное наследование, а также использует параметризованные типы, которые работают аналогично шаблонам языка Си++. Curl поддерживает развитые средства макрообработки (часть из них пока только на бумаге), включая выражение include для вставки фрагментов исходного текста.

Для языка характерны расширяемый синтаксис и поддержка строгой типизации одновременно с сохранением бездекларированного использования переменных. Как отмечают авторы языка [4], значительное влияние на проектирование Curl оказали Лисп, Си++, Tcl/Tk, TeX и HTML. Генерирование исполняемого кода осуществляется «на лету» с помощью встроенного в специальный подключаемый модуль Surge динамического компилятора (JIT). Инструментальная система Curl и компилятор написаны на самом Curl (поставляются в исходных текстах). Система безопасности строится вокруг разновидности модели песочницы, принятой в Java. Важной особенностью реализации языка является то, что при компиляции учитывается (наследуется) контекст среды (в зависимости от полномочий и установок формируется и соответствующий код).

Для хранения информации на компьютере пользователя в Curl предусмотрен механизм хранения долговременных данных (persistent data), гораздо более гибкий, чем известные средства cookie-файлов. А вот обработка исключений, схожая с Java, реализована в несколько ограниченной форме.

**Листинг.** Пример создания и вызова собственной процедуры BlinkingText.

```
{define-text-proc {BlinkingText ...}:any
  let color-str:String = "green"
  let flash:TextFlowBox =
    {TextFlowBox
      font-weight = "bold",
      color = color-str,
      {value ...}
    }
  {Timer
    interval = 1s,
```

```
{on TimerEvent do
  {if color-str == "red" then
    set color-str = "green"
  else
    set color-str = "red"
  }
  set flash.color = color-str
}
}
{return {paragraph {value flash}}}
}
{BlinkingText Curl приветствует Вас!}
```

Сценарии Curl скорее похожи на апплеты на Java, тиклеты на Tcl (среда SafeTcl) и апплеты на Обероне (среда Juice), нежели на Flash-модули. Curl может применяться при разработке всех видов ПО: клиентского, связующего и серверного, хотя именно поддержка клиентской части сейчас является его наиболее сильной стороной.

Несмотря на атмосферу эйфории, царящую в компьютерном сообществе в связи с потенциальными возможностями языка Curl, сейчас проблем пока гораздо больше, чем решений. Основное внимание уделено графическому интерфейсу (с активным использованием наработок Tcl/Tk), сетевым средствам и языку XML. Так, средства двух- и трехмерной графики со встроенным механизмом рендеринга доминируют, в то же время поддержка организации доступа к БД и компонентного программирования хромают.

Язык Curl, появление которого было вызвано очередным витком эволюции Internet, предпринимает попытку объединить на основе сценарного программирования работу тех, кто непосредственно создает информационный продукт (авторов), и тех, кто обеспечивает его распространение и сервисное обслуживание (программистов). В связи с этим стоит вспомнить слова, сказанные Марвином Минским (1969), автором первого человекоподобного робота и основателем Лаборатории искусственного интеллекта все в том же Массачусетском технологическом институте: «Будущие языки программирования станут в большей степени сосредоточиваться на целях и в меньшей степени на процедурах, специфицированных программистом».

## Литература

0. Colony G. My View: X Internet // Forrester Research, October 2000.
0. Korzeniowski P. A New Web Language Is Born // eAI Journal, August 2001.
0. Muffke F. The Curl Programming Environment // Dr.Dobb's Journal, September 2001.
1. Hostetter M. et al. Curl: A Gentle Slope Language for the Web // MIT Laboratory for Computer Science, <http://cag-www.lcs.mit.edu/curl/wwwpaper.html>.