

Руслан Богатырев

## Тайны Microsoft. Смарт-теги и гипертекст

Источник: Мир ПК, #06/2002

Я думал, произойдет настоящая революция,  
а вижу полное предательство.  
Теодор Нельсон,  
автор термина «гипертекст»

*Мы настолько привыкли работать с гиперссылками в Интернете, что, наверное, даже не задумываемся над тем, насколько со временем исказилась первоначальная идея провозвестников этой революционной технологии. Наши дни показывают, что когда мы забываем историю и истоки технологий, то не только лишаем себя возможности предугадывать будущее, но и совершаем такие ошибки, которых, несомненно, сумели бы избежать.*

### Что скрывается за термином «смарт-тег»

Корпорация Microsoft впервые стала публично применять термин «смарт-тег» (smart tag, в пер. с англ. — «интеллектуальная метка») в 2001 г. для обозначения специального механизма контекстно-зависимого исполнения действий на основе ассоциативной связи с текстовыми фрагментами. Причем изначально данную концепцию, являющуюся своеобразным гибридом гипертекста и сопоставляемого с ним механизма меню, планировалось использовать в новом поколении двух ключевых продуктов Microsoft: в семействе операционных систем Microsoft Windows XP (преимущественно в рамках браузера Internet Explorer 6.0) и в пакете офисных приложений Microsoft Office XP (в Word 2002, Outlook 2002, Excel 2002 и PowerPoint 2002). Однако эти планы не были реализованы (из браузера данный механизм пришлось изъять), и в итоге смарт-теги появились на рынке в весьма урезанном виде, да и то только в Office XP.

К сожалению, в отечественной и зарубежной компьютерной прессе смарт-теги окутаны такой завесой таинственности, что разобраться в их сути и уж тем более в перспективах применения далеко не просто, хотя все поясняется простой формулой: «смарт-тег = гиперссылка + меню + компонент» (табл.1).

Не будем пока вдаваться в подробности причин весьма сдержанного маркетингового продвижения смарт-тегов корпорацией Microsoft (об этом — чуть позднее). Пока же попробуем разобраться, что же это такое, на примере набора русскоязычных смарт-тегов компании Exteria (см. <http://www.exteria.ru/pr/20020118>).

Посмотрите на рисунок. Перед вами изображение Рабочего стола в Windows 2000, где слева открыто окно Internet Explorer, справа — Microsoft Outlook. При взгляде на Web-страницы можно заметить, что некоторые фразы (слова и даже их фрагменты) подчеркнуты специальной пунктирной чертой. Если к ним (в данном случае — «Всеволод Предтеченский») подвести курсор мыши (без нажатия кнопок), то тут же появляется кнопка с изображением латинской буквы i в кружочке (как известно, это международный символ, обозначающий справочную службу). Теперь на изображении i нажмите и удерживайте левую кнопку мыши: тут же раскроется соответствующее меню, где вы уже обычным образом выбираете требуемый пункт. На нем отпустите левую кнопку мыши, и тогда сразу выполнится ассоциированное с этим пунктом действие. Таким образом, все работает одним нажатием мыши, хотя и растянутым во времени.

Таблица 1. Термины для смарт-тегов.

Термин	Определение
Смарт-ссылка (smart link)	Помеченный (подчеркнутый особым образом) текст, распознаваемый обработчиками смарт-тегов (другими словами, это словокомплекс, входящий во множество значений типа соответствующего смарт-тега)
Значок смарт-тега (smart tag symbol)	Кнопка вызова меню (i в кружочке), соотнесенная с помеченным текстом
Смарт-тег (smart tag)	Смарт-ссылка вместе с ассоциированным с ней набором действий (представлен в меню)
Встроенный смарт-тег (embedded smart tag)	Штатный смарт-тег, встроенный в приложение (Word, Excel, PowerPoint)
Тип смарт-тега (smart tag type)	Набор допустимых значений-словокомплексов (указаны перечислением и/или правилом)
Обработчик смарт-тега (smart tag handler)	Программный код оперирования смарт-тегом, состоящий из двух частей: распознавателя (recognizer) и исполнителя (action provider); обработчики могут быть простыми (распознаватель работает с одним типом смарт-тега) и составными (распознаватель оперирует несколькими типами смарт-тегов)
Контейнер смарт-тегов (smart tag container)	XML- или COM-компонент, определяющий типы смарт-тегов и обработчики смарт-тегов

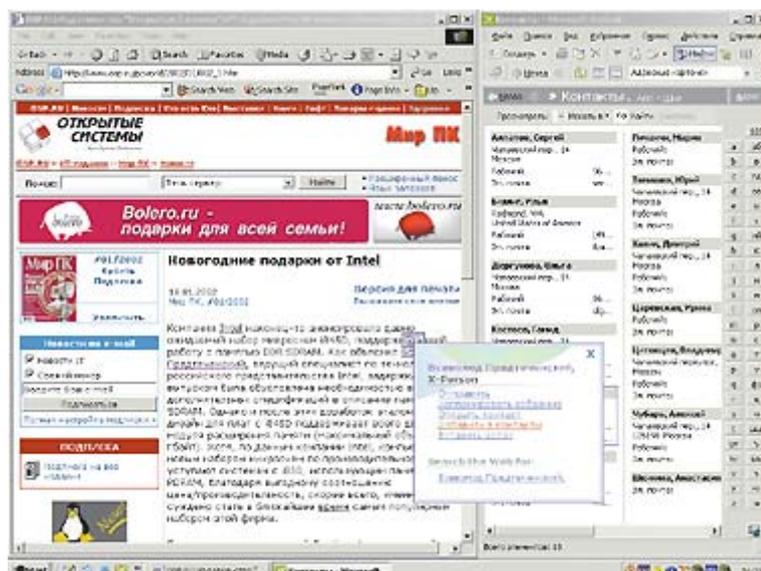


Рис. 1. Пример работы смарт-тегов (русские имена, X-Person)

В нашем случае, когда мы нашли в тексте интересующее нас имя персоны (Всеволод Предтеченский) и подвели к нему курсор мыши, то инициировали раскрытие меню (X-Person), чтобы выбрать там действие «Добавить в контакты». Если бы эта операция была выполнена, то в Outlook открылась бы соответствующая форма регистрации новой персоны, куда в нужные поля автоматически заносились бы в требуемом порядке параметры персоны, извлеченные из текста (в нашем случае — имя и фамилия). Кстати, абсолютно так же работает механизм смарт-тегов и с PDF-документами, просматриваемыми через браузер IE 6.0 (за счет интеграции с Adobe Acrobat).

Обратите внимание, что смарт-теги действуют по вполне прозрачной схеме:

- распознавание фраз-лексем (далее - словокомплексы);
- отождествление словокомплексов с известными типами смарт-тегов (имена, компании, денежные единицы, термины, номера авиарейсов, договоров, накладных, артикулы товаров и т.п.), причем один и тот же словокомплекс может относиться сразу к нескольким типам смарт-тегов;
- отображение меню, ассоциированного с данным типом смарт-тега;
- исполнение действия, связанного с выбранным пунктом меню (причем действие параметризуется текстовым окружением распознанного словокомплекса вплоть до передачи всего абзаца).

Пытливый читатель наверняка сообразит, что подобный механизм дает новую жизнь технологии гипертекста. Если в роли пунктов меню будут выступать альтернативные ссылки на разные Web-страницы (источники информации), то получатся вариативные гиперссылки. Так, при появлении в тексте слова «погода» ему автоматически будут сопоставлены на выбор различные варианты (на основе информации местного бюро погоды, Росгидромета, Национальной службы погоды США, CNN и т.п.). Обратите внимание на весьма существенную деталь: гиперссылки в случае смарт-тегов выделяются динамически. Переставите ли вы слова, измените ли падеж/число/спряжение, все равно ссылка будет выделяться автоматически и корректно (в случае, разумеется, соответствующей реализации распознавателя смарт-тегов).

Еще одно назначение смарт-тегов — исполнение служебных инструкций в форме письма/документа поступающих к пользователю в почтовом клиенте Microsoft Outlook и текстовом редакторе Microsoft Word. Если выбрать соответствующий пункт в тексте, то здесь можно инициировать бизнес-процессы, специфичные для конкретного предприятия в контексте полномочий и роли данного исполнителя.

Казалось бы, наконец-то появилось новое мощное средство интеграции приложений, способное совершить переворот не менее грандиозный, чем это сделала в свое время концепция гипертекста в рамках ставшей уже привычной Web-среды. Но не тут-то было...

## Патентные войны. Дела British Telecom и Microsoft

Компьютерные и информационные технологии очень часто вступают в противоречие с интересами бизнеса, и потому вряд ли стоит удивляться их изъятию и искажению, а также тому, что в результате до потребителя (т.е. до нас с вами) доходят лишь такие реализации, которые удовлетворяют интересам конкурирующих компаний и сложившейся маркетинговой ситуации.

Вот и со смарт-тегами наблюдается довольно запутанная картина. Еще накануне появления Microsoft Office XP (напомню, что официальная дата его представления в США — 31 мая 2001 г., в России — 19 сентября 2001 г.) в прессе разгорелись страсти вокруг технологии смарт-тегов. На целом ряде сайтов публиковались сообщения, предупреждающие о том, что смарт-теги таят в себе угрозу. Разумеется, кампания развертывалась вокруг возможного нарушения интеллектуальных прав при применении этой технологии.

Что же так испугало противников смарт-тегов? Прежде всего, просмотр Web-страниц сквозь призму смарт-тегов (мнимые «розовые очки»), когда возникала возможность динамической простановки ссылок там, где автор текста их не размещал, с направлением на те источники информации, которые важны с точки зрения разработчика и пользователя смарт-тега, а не самого автора. И хотя в Microsoft были заранее предусмотрены возможности отключения данного механизма (в частности, в браузере IE 6.0 блокировалось распознавание HTML-страниц, помеченных метатегом MSSmartTagsPreventParsing), общественное мнение было столь взбудоражено, что во избежание возможных проблем корпорация Microsoft уже в июне 2001 г. официально объявила об изъятии механизма смарт-тегов из IE 6.0, входящего в состав Windows XP. В противном случае могли возникнуть серьезные правовые препятствия для выпуска этой ОС, намеченного на 25 октября 2001 г. Обратите внимание, что рисунок был получен с помощью последней бета-версии IE 6.0. В финальной версии механизм смарт-тегов изъят, и увидеть подобное распознавание уже не удастся (хотя аналогичную работу тех же смарт-тегов можно наблюдать в Word 2002).

О потенциальной возможности нарушения (при использовании смарт-тегов) закона об авторском праве и американских федеральных правил ведения бизнеса заявила, в частности, Робин Гросс, адвокат фонда Electronic Frontier Foundation. А по мнению ряда других юристов, автоматическая простановка динамических ссылок подпадает под принцип цитирования, что никак не нарушает положений закона. Как бы то ни было, данный вопрос требует серьезной юридической проработки, что, видимо, и стало главной причиной изъятия механизма смарт-тегов из Internet Explorer и Windows XP. Скорее всего, это временная мера, и не исключено, что уже в нынешнем году можно будет ожидать полноценного появления смарт-тегов в ведущих продуктах Microsoft.

Если говорить об официальной позиции самой Microsoft, то еще в июне 2001 г. было заявлено, что изъятие механизма смарт-тегов носит временный характер и что обусловлено оно недостаточной готовностью компании к реализации данной технологии до октября 2001 г., т.е. до момента начала поставок Windows XP. Для полноты картины наряду с чисто правовыми вопросами и проблемами, связанными с не совсем корректной реализацией смарт-тегов (что и показывает опыт их разработки и использования в приложениях Office XP), необходимо сказать и о таких не менее серьезных задачах, как развертывание наборов смарт-тегов в компаниях, обеспечение корпоративной безопасности и наличие на рынке практически ценных решений (для всего этого нужно выиграть время).

Несостоявшиеся иски против Microsoft по поводу смарт-тегов наталкивают на параллели с двумя другими нашумевшими делами: ставшим уже достоянием истории иском Apple к Microsoft, а также недавним иском British Telecom к Интернет-провайдеру Prodigy Communications.

Компания Apple подала 22 марта 1988 г. иск против Microsoft и Hewlett-Packard, в котором было выдвинуто обвинение в использовании интерфейса Macintosh в продуктах Microsoft Windows и HP NewWave. Одно из самых громких судебных разбирательств длилось около пяти лет (к его окончанию на рынке уже появилась ОС Windows 3.1). Как известно, и Microsoft и Apple заимствовали идеи из одного и того же источника — из Xerox PARC (проекты Alto и Star). Это было достаточным основанием (так называемый prior art) для того, чтобы в итоге отклонить иск Apple. Стоит заметить, что в 1991 г. Xerox выдвинула встречный иск к Apple, который суд тоже отклонил.

Что же касается иска British Telecom, то о нем стоит поговорить подробнее, тем более что он имеет самое непосредственное отношение к смарт-тегам и затрагивает святая святых — технологию гипертекста, лежащую в основе современного Интернета и всей Web-среды. Когда в июне 2000 г. крупнейшая английская телекоммуникационная компания British Telecom обнаружила в своих архивах патент США No. 4873662 «Information Handling System and Terminal Apparatus Therefor», то она обратилась к Prodigy и еще 16 Интернет-провайдерам (включая и прямого своего конкурента — America Online) с предложением выкупить лицензию на его использование (патент был принят в октябре 1989 г., а срок его действия истекает в 2006 г.). По мнению экспертов British Telecom, данный патент регулирует механизм доступа к территориально удаленной информации через механизм гиперссылок. На самом же деле, если внимательно прочитать текст патента, несложно сделать вывод, что к самим гиперссылкам это имеет весьма отдаленное отношение, хотя и отрицать такую связь было бы неверно (вопрос, как обычно, опирается в терминологию и трактовку изложенных принципов).

В самом патенте речь идет о центральном компьютере и терминалах (что уже не соответствует структуре Интернета), о делении всей хранимой и передаваемой по линиям связи информации на блоки двух типов: отображаемой на экране и управляющей этим отображением. Как указано в документе, «второй тип блоков может включать информацию для обеспечения полного адреса другого блока, который может быть выбран с помощью действия определенной клавиши на клавиатуре». Данный патент, скорее, покрывает важную область таблиц стилей (правами на которые, кстати, обладает Microsoft). Интересно, что заявка на патент была подана еще в июле 1976 г., т.е. за 13 лет до его выдачи.

Как известно, и в США, и в Великобритании, в отличие от континентальной Европы, действует прецедентное право, что еще более усугубляет ситуацию. Если дело против Prodigy выиграет British Telecom, то, несмотря на все заверения руководства этой компании о заведомом отказе от преследования мелких провайдеров, дальше наверняка покатится волна автоматически выигрываемых исков к другим провайдерам. А это, в свою очередь, может существенно подорвать темпы развития Интернета. Кроме того, многие компании призадумаются об использовании Web-решений, поскольку возникнет вероятность возбуждения исков и против них. Мрачные картины можно рисовать и дальше, но допустить такое развитие событий отнюдь не в интересах американского бизнеса. Пока же рассмотрение вопроса отложено до сентября 2002 г.

Разумеется, корпорация Microsoft не могла не знать о таком грандиозном деле, так что если причины частичной реализации смарт-тегов кроются в решении данного вопроса (сейчас они поддерживаются только в Office XP), то в браузере Internet Explorer и, следовательно, в Windows XP ожидать появления механизма смарт-тегов раньше осени 2002 г. не стоит.

## **Дуглас Энгельбарт. Демонстрация всех времен и народов**

Иск British Telecom к Prodigy вновь привлек внимание общественности к истории формирования технологий. Уж казалось бы, такая вдоль и поперек изученная тема, как гипертекст, вряд ли может таить в себе что-то новое.

Действительно, воспринимается эта история стереотипно так: мы пользуемся теми гиперссылками, которые в начале 1990-х гг. ввел Тим Бернерс-Ли, когда экспериментировал в Центре ядерных исследований (CERN) с языком SGML и создал на его основе язык HTML. Но идею придумал, разумеется, не он. Идеологами гипертекста признаны Ванневар Буш, Дуглас Энгельбарт и Теодор Нельсон. К сожалению, со временем концепция гипертекста стала восприниматься многими через кривое зеркало его повсеместно доступной реализации в современных браузерах (HTML с протоколом HTTP). А ведь HTML-исполнение не только имеет массу вполне устранимых изъянов, но и искажает первоначальные замыслы авторов гипертекста (отсутствие двунаправленных ссылок, наглядной визуализации связей документов и т.п.). Достаточно вспомнить хотя бы слова Теодора Нельсона, взятые в качестве эпиграфа для настоящей статьи. Чтобы доказать миру свою правоту, он вот уже добрых два десятка лет трудится (теперь в Японии) над созданием собственной системы XANADU.

Но это тема отдельного обстоятельного разговора. Здесь же я лишь вкратце приведу свою таблицу истории развития гипертекста в контексте обсуждения смарт-тегов (табл. 2).

Таблица 2. История развития гипертекста и смарт-тегов.

Автор	Год	Работы
Ванневар Буш	1945	"As We May Think", воображаемая система Memex
Дуглас Энгельбарт	1962	"Augmenting Human Intellect: A Conceptual Framework", система N-LAM/T
Теодор Нельсон	1965	Термин "гипертекст"
Дуглас Энгельбарт	1968	Система NLS/Augment (oN Line System), публичная демонстрация в Сан-Франциско
Алан Кей	1973	Термин "персональный компьютер", идея компьютера Dynabook
Чарльз Саймони, Батлер Лэмпсон	1974	Система редактирования Bravo, WYSI-WYG-режим, шрифты и стили
Теодор Нельсон	1974	"Dream Machines", гипертекстовая система XANADU
Алан Кей	1974	Система Smalltalk
Дэвид Лиддл	1981	Система автоматизации офиса Xerox Star
Чарльз Саймони	1981	Меню в Multiplan
Дональд Кнут	1986	Термин "Web"
Никлаус Вирт, Юрг Гуткнехт	1988	Система Oberon: гипертекст, Web-службы, команды-прототипы смарт-тегов, компонентное ПО, первые апплеты, прототип Java
Microsoft	1990	Язык Visual Basic
Тим Бернерс-Ли	1991	Язык HTML
Марк Андрессен	1993	Браузер Mosaic
Микаэль Франц	1994	Термин "апплет", JIT-компиляция
Джеймс Гослинг	1995	Язык Java
Microsoft	1996	Язык VBA 5.0 (интеграция всех приложений MS Office)
W3C	1997	XML
Microsoft	2001	Смарт-теги в Office XP

Среди важных вех минувших лет стоит выделить, пожалуй, труды трех авторов: Дугласа Энгельбарта, Чарльза Саймони и Никлауса Вирта, отстоящие друг от друга на десятилетие.

Итак, в декабре 1968 г. в Сан-Франциско открылась Единая осенняя компьютерная конференция (FJCC). Именно на ней произошло событие, впоследствии получившее название «Великая демонстрация всех времен и народов». Дуглас Энгельбарт, сотрудник Стэнфордского исследовательского института (SRI, Stanford Research Institute) в Менло-Парк (что неподалеку от Купертино), вместе со своими коллегами представил прообраз персональных компьютеров и будущей Web-среды.

Он появился на сцене с микрофоном и наушниками и уселся перед демонстрационным экраном за замысловатый рабочий стол, на котором были разложены клавиатура и какие-то непонятные инструменты. Энгельбарт показывал всем, что будет представлять собой повседневная работа в будущем. Он растягивал строчки текста до огромных размеров, а потом сжимал их обратно. Текст вверху экрана можно было «замораживать», а в нижней части — изменять. На разделенном на сектора экране дисплея представлялись текст, графика и видео. Документы имели сноски на другие документы, и по этим сноскам можно было перемещаться!

Всеми действиями Энгельбарт управлял с помощью придуманного им для NASA манипулятора с двумя потенциометрами, получившего название «мышь» (саму же мышь компания Xerox залицензировала в 1971 г.). Световое пятно (оно именовалось «клоп») вслед за перемещениями колесиков мыши двигалось по экрану. Энгельбарт мог щелкнуть на слове, передвинуть его в документе и даже перенести в другой.

Затем наступил черед демонстрации прообраза нынешних видеоконференций. Энгельбарт представил одного из членов своей команды, севшего за такой же рабочий стол, как у него, с микрофоном и наушниками. Перед этим участником тоже находилась телекамера. Энгельбарт и его коллега смогли на расстоянии разговаривать между собой, видеть друг друга и одновременно совершать манипуляции с общим документом.

Такая демонстрация предвосхитила многие открытия, сделанные в последующие годы: Web-документы, экранные окна, мышь, видеоконференции, коллективную работу над документами. Алан Кей, автор языка и системы Smalltalk, компьютера-мечты Dynabook, активный участник проекта Alto, так вспоминал о том времени: «Хотя я и знал заранее все, что они собирались показать, для меня оказались величайшим открытием в жизни объем, широта и глубина проницательности Энгельбарта. Бурные овации свидетельствовали о мгновенном признании аудитории. Все поняли: произошло что-то значительное, и с этого момента мы не сможем оставаться такими, какими были раньше».

Однако в течение многих лет истинный вклад Энгельбарта так и не был оценен по достоинству. Тед Нельсон, познакомившийся с Энгельбартом весной 1967 г., с горечью прокомментировал эту ситуацию: «То, что этот компьютерный пророк не получил никакой благодарности, — истинная правда. Отсутствие признательности Дугу равнозначно ситуации, когда вы стоите рядом с Empire State Building и не знаете, какова его высота. Вам просто известно, что оно выше вас, и многие люди не одобряют этого».

Спустя три десятилетия Дуглас Энгельбарт наконец-то удостоился заслуженных почестей — в 1997 г. он получил премию Лемельсона Массачусетского технологического института в размере 500 тыс. долл. Это самое большое в мире денежное вознаграждение, назначаемое за американские изобретения и открытия.

Спустя пару лет после исторической демонстрации 1968 г. корпорация Xerox создала свой новый исследовательский центр в Пало-Альто (PARC, Palo Alto Research Center), куда перешли многие сотрудники Энгельбарта и его коллеги по Стэнфордскому университету.

(Окончание следует)

Руслан Богатырев

## Тайны Microsoft. Смарт-теги и гипертекст

Источник: Мир ПК, #07/2002

### Чарльз Саймони. Тайный архитектор Microsoft.

*Окончание. Начало в №.6*

Долгие годы имя Чарльза Саймони, одного из выдающихся программистов минувшего столетия, главного архитектора Microsoft, равно как и имя Энгельбарта, было покрыто завесой молчания. Когда я листал страницы архивных материалов, мне удалось найти в публикациях начала 1990-х гг. лишь едва заметный реверанс в сторону Саймони в весьма обстоятельном интервью Билла Гейтса. Он дал его в 1993 г. Дэвиду Эллисону из Национального музея истории США по случаю своего награждения премией Computerworld Smithsonian Award.

«Microsoft Word, — вспоминает Гейтс, — хотя и являлся вторым приложением, с которым мы вышли на рынок после Multiplan, знаменовал собой важный этап... Мы взяли на работу Чарльза Саймони из Xerox PARC. Мы знали, что графический интерфейс там уже был реализован и что лазерные принтеры получат большое распространение. Поэтому мы спроектировали нечто такое, чья структура была бы подготовлена для графического мира. И действительно, мы добились того, что текст можно было показывать курсивом и полужирным начертанием, что с ним можно было работать с помощью мыши, а это позволяло добиться огромной точности в композиции страниц... Пользовательский интерфейс был идентичен интерфейсу в Multiplan. Так появилось семейство продуктов, которое мы назвали Multi-Tool Family».

В реальности же роль Саймони в становлении корпорации, именуемой Microsoft, неизмеримо выше. Обратимся к истории компьютеров и страницам его биографии.

В начале 1960-х гг. Саймони на своей родине в Венгрии писал программы для советского компьютера «Урал-2». Именно тогда, работая в машинных кодах, он и приобрел умение создавать максимально эффективные программы, функционировавшие в условиях крайне ограниченных машинных ресурсов. Привычка делать мнемонические обозначения вылилась у него в создание того стиля формирования идентификаторов, который теперь известен под названием «венгерская нотация».

Будучи аспирантом Стэнфордского университета, Саймони начал работать в Xerox PARC еще в 1970 г., т.е. тогда, когда образовалась эта, пожалуй, самая знаменитая исследовательская лаборатория мира. Основным в те годы был проект Alto. Но компьютер Alto так и не стал промышленным ПК — было создано всего лишь около 2000 машин данного семейства. Их установили в Белом доме, Администрации президента и Национальном бюро стандартов, на них печатались отчеты и документы Конгресса США.

Чарльз Саймони в рамках создания компьютера Alto отвечал за направления работ в области графического интерфейса (GUI), в частности за разработку текстового процессора Bravo. По его словам, это был первый в мире текстовый процессор, обеспечивающий режим WYSIWYG (What You See Is What You Get) — режим точного соответствия представленного на экране документа напечатанному на принтере (композиция страницы, гарнитуры, начертания, кегель, стили), причем работать с таким представлением страницы можно было с помощью мыши. В 1976—1978 гг. он завершил разработку усовершенствованной версии BravoX, положенной в основу одного из самых технологически совершенных и одновременно коммерчески неудачных проектов — Xerox Star. (Спустя десятилетие идеи Xerox Star были взяты на вооружение и развиты Стивом Джобсом в рамках проекта NeXT.) Генеалогическая связь разных проектов показана на рис. 1.

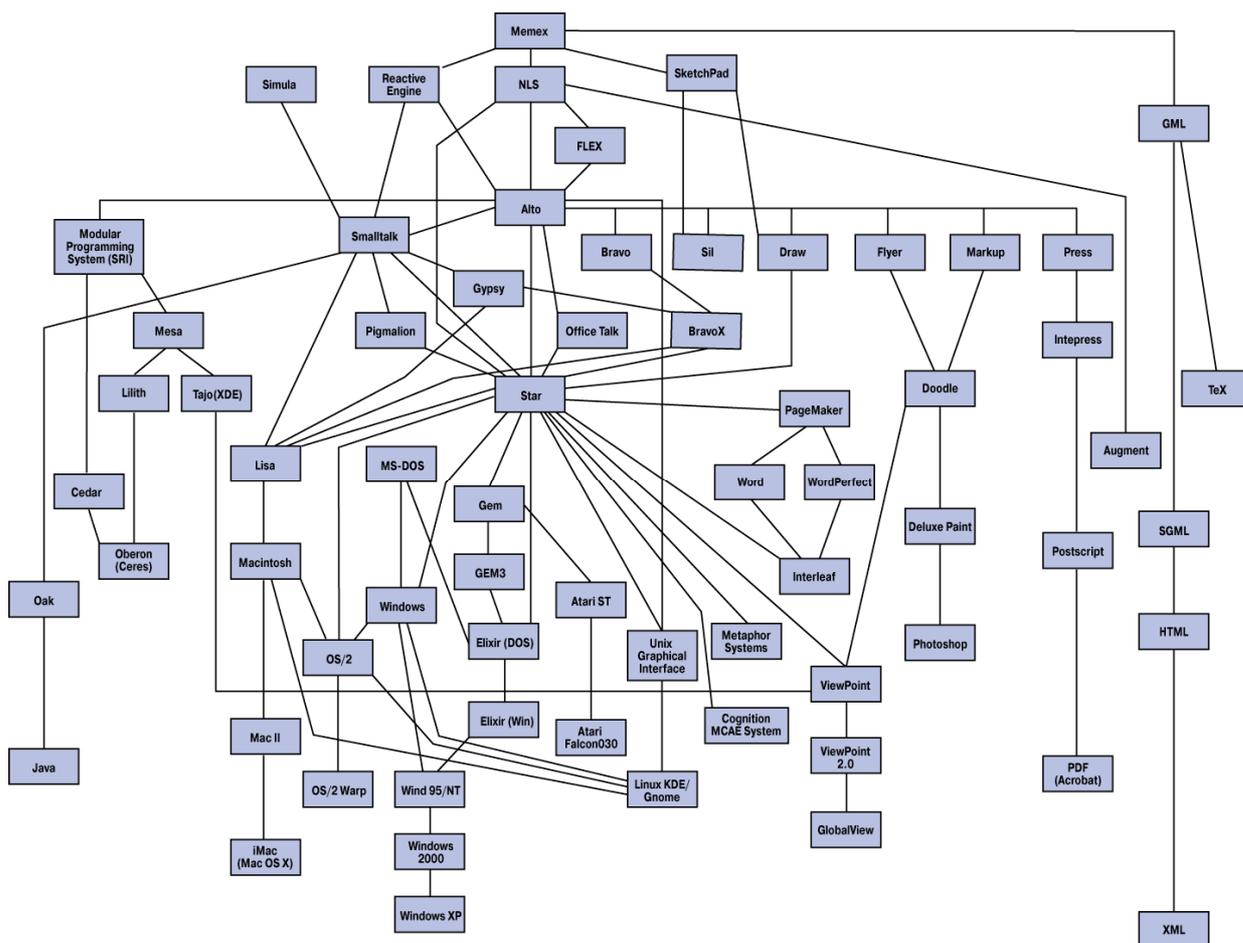


Рис. 1. Генеалогическое древо систем и языков (гипертекст и интерфейс).

Проект Star (начавшийся в 1975 г.) в Xerox PARC вел департамент системных разработок SDD (Software Development Department). В июле 1981 г. работа над ним вылилась в то, что Xerox представила настоящий персональный компьютер Xerox 820, получивший кодовое название Worm («червяк», приготовившийся, видимо, съесть яблоко — Apple). До выхода IBM PC оставался один месяц... Однако время (да и очень ценные кадры) было уже упущено. Билл Гейтс вспоминал: «Ребята из Xerox замахнулись слишком высоко, они пытались сделать невыполнимое, но не смогли найти возможностей, чтобы это осуществить. Когда наконец им это удалось, они чуть-чуть опоздали».

Не дождавшись завершения проекта Star, разочаровавшись и не надеясь увидеть когда-либо повсеместное использование плодов своего труда, из Xerox PARC стали уходить один за другим ведущие сотрудники. Чарльз Саймони перешел в Microsoft, Алан Кей — в Atari. Команду Bravo покинул не только ее руководитель, но и другие участники (однако они направились к прямому конкуренту Microsoft — компании Apple). Так, в Apple попали Том Мэллой, написавший редактор LisaWrite, и Ларри Теслер — один из ведущих разработчиков ПО для будущего компьютера Apple Lisa. Что же касается Стива Джобса, то он не работал в Xerox PARC, но мысль о создании своего ПК зародилась у него в 1979 г. именно там, когда он воочию познакомился с исследованиями в этом центре. Вдохновленный увиденным, Стив загорелся идеей использовать это при конструировании компьютера Lisa, а после того как его несколько отодвинули от проекта, перенацелился на новый ударный проект — Macintosh.

Но вернемся к Саймони. В конце 1980 г. Чарльз Саймони собрался уходить из Xerox PARC, и тут Роберт Меткалф (автор Ethernet) посоветовал ему переговорить с рядом лиц, причем первым в списке значился Билл Гейтс (остальные имена Саймони по прошествии стольких лет вспомнить уже не мог). В ноябре Чарльз встретился с Гейтсом и Стивом Баллмером, после чего согласился на предложенные условия (руководство новым ключевым подразделением — Application Software Group, создание на базе Electronic Paper нового процессора электронных таблиц, подготовка текстового редактора Word).

Саймони поступил на работу в Microsoft 6 февраля 1981 г. и стал сотрудником номер 40. Важно заметить, что в империи Basic (как еще называли Microsoft) именно Саймони настоял на том, чтобы основным языком разработки стал язык Си [2]. На Си был написан Multiplan — прообраз будущей Excel. Несмотря на относительную маркетинговую неудачу с Multiplan, в Microsoft сделали правильные выводы из своих ошибок, и в 1985 г. та же таблица вышла под новым флагом — Excel. С этого момента она начала неуклонно захватывать рынок. Параллельно с Multiplan в 1981 г. были инициированы работы по созданию текстового редактора Multi-Tool Word. В острой конкуренции с WordStar и WordPerfect редактор Word корпорации Microsoft спустя несколько лет после своего официального появления в ноябре 1983 г. завоевал пальму первенства, а с выходом системы Windows стал доминирующим текстовым процессором. Как вспоминает Чарльз Опперман, бывший менеджер проектов в Microsoft, в начале 1982 г. Саймони вместе с Биллом Гейтсом уже вынашивали идеи реализации графической версии MS-DOS, которая привела к появлению операционной системы Windows.

В 1991 г. Саймони был переведен в Microsoft Research, где занимался разработкой концепций интенционального программирования. В настоящее время он в качестве одного из 16 выдающихся инженеров Microsoft (Distinguished Engineers) входит в Business Applications Division.

Большая заслуга Саймони состоит не только в непосредственном участии в создании трех ведущих продуктов Microsoft — Word, Excel и Windows, но и в том, что механизм меню стал столь распространенным и популярным. Вот как в интервью, данном журналу PC World (ноябрь 1983 г.), Чарльз Саймони пояснил идею меню. «Мне нравится явная аналогия с рестораном. Скажем, я захожу во французский ресторан и не умею говорить по-французски. Обстановка необычная, самочувствие неважное. Я боюсь сделать глупость и потому напряжен. Ко мне подходит очень внушительный на вид официант и заговаривает со мной по-французски. Я покрываюсь холодным потом. Наверное, те же самые чувства испытывает бухгалтер, садящийся к компьютеру... Какой же выход? А вот какой: я беру меню и указываю на что-нибудь пальцем. Я на правильном пути. Может быть, я не получу то, что мне нужно, может быть, мне придется есть одних улиток, но я по крайней мере не буду чувствовать неловкость. Но представьте себе, что вы во французском ресторане, в котором нет меню, — это ведь кошмар. То же самое относится и к программе для компьютеров — вам нужно меню. Меню дружелюбны, поскольку люди знают, что означают их пункты, и поскольку они могут выбрать нужный, просто указав на него. Им не приходится разыскивать что-то такое, чего они не смогут найти, и вводить какие-то команды, в которых могут быть ошибки».

Как вы, наверное, знаете, в некоторых московских ресторанах, специализирующихся на экзотических для нас блюдах восточной кухни (японской, китайской, тибетской и т.п.), для удобства посетителей предлагают меню с фотоиллюстрациями соответствующих блюд. Справедливости ради надо заметить, что не Саймони первым придумал механизм меню (он появился в конце 1960-х гг. еще на первых терминалах IBM 3270 с прямым позиционированием курсора). Да и само название «меню» дали, скорее всего, именно в Apple сразу после выхода компьютера Lisa. Дэвид Лиддл, руководитель проекта Xerox Star, вспоминает: «Мы никогда не называли ни один из этих механизмов в Star термином «меню». Apple всегда именовала их меню, мы же не делали этого никогда».

## Команды в системе Oberon как прототип смарт-тегов

Система Oberon (Oberon System, <http://www.oberon.ethz.ch>), созданная в Швейцарском федеральном технологическом институте (ETH, Цюрих) в 1988 г. (рис. 2), играет в наши дни приблизительно ту же роль, что и проекты Alto и Xerox Star в начале 1980-х гг. Для таких корпораций, как Microsoft, IBM и Sun Microsystems, она стала источником плодотворных идей, среди которых можно выделить документо-ориентированный интерфейс, языки разработки переносимого ПО (Java), машинно независимый мобильный код, апплеты, компонентное ПО, динамическую компиляцию (JIT-компиляцию), Web-службы и конечно же смарт-теги.

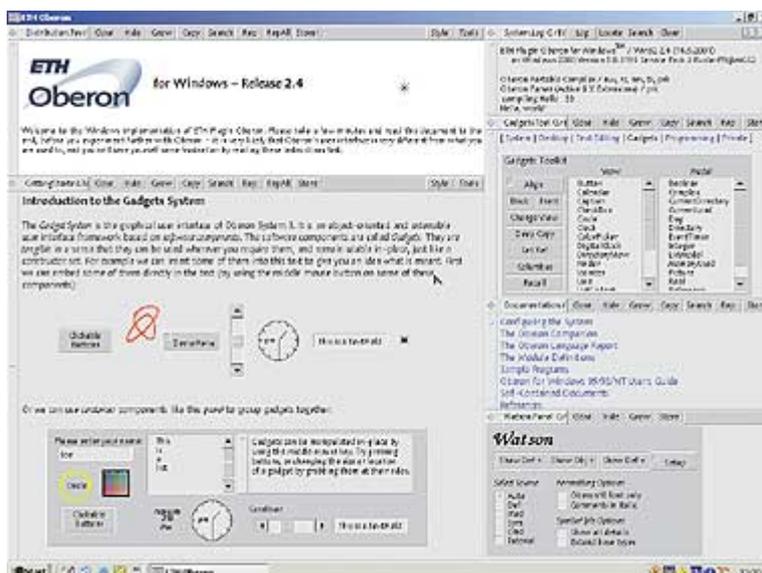


Рис. 2. Панель системы Oberon (поверх системы Windows 2000).

Идеологами этой системы стали Никлаус Вирт и Юрг Гуткнехт, профессора ETH [3]. Система Oberon и ориентированный на нее компьютер Ceres создавались не на пустом месте. За несколько лет до этого (1980—1981 гг.) Вирт вместе с группой своих коллег успешно завершил проект разработки 16-разрядного персонального компьютера Lilith с растровым дисплеем, WYSIWYG-интерфейсом, локальной сетью и лазерным принтером. Весь проект, включая операционную систему Medos, сервисные программы и компилятор системного языка, был реализован на преемнике Паскаля — языке Модуль-2. Идея проекта Lilith возникла у Вирта (он, кстати, в свое время учился у Энгельбарта) в момент посещения им в 1976—1977 гг. все того же Xerox PARC. Однако он обратил внимание не столько на компьютеры семейства Alto, сколько на менее известный проект Mesa (своеобразную системно-ориентированную разновидность Паскаля). Этот проект, на результаты которого опирался Star, вела лаборатория CSL (Computer Science Laboratory). Язык Mesa стал прототипом двух языков — Модуль-2 и Ады. Он использовался внутри корпорации Xerox для реализации не только Star, но и ПО DocuTech. Проект Cedar (преемник Mesa в Xerox PARC), направленный на создание языка для реализации системы DocuPrint, в определенной степени стал катализатором другого проекта Вирта — системы Oberon.

Наш интерес в свете данной статьи будет обращен прежде всего к прототипам смарт-тегов в системе Oberon — к так называемым командам Oberon (commands), играющим роль низкоуровневых обработчиков смарт-тегов. В основе Oberon лежит исповедуемая Виртом концепция расширяемых систем, предусматривающая возможность органичного расширения базовой системы в контексте объектно-ориентированного каркаса. Каждая экспортируемая процедура без параметров (соответствующий метод) в новом загружаемом модуле автоматически становится командой операционной системы. Параметры передаются команде через буфер обмена и области выделения. Чтобы ее выполнить, достаточно в любом месте пользовательского интерфейса (даже в заголовке окна!) ввести (или выделить) текст, а затем просто подвести мышью к тексту и нажать ее кнопку, например:

*Gadgets.Insert Clocks.NewClock ~*

В данном примере команда Insert из модуля Gadgets считывает параметры вплоть до знака завершения (~) и выполняет соответствующий код данной процедуры (в нашем случае в то место, где стоит текстовый курсор, помещается апплет-часы, работающий до момента своего удаления из документа). Программисты в Oberon активно пользуются подобным механизмом, вставляя прямо в исходный текст своего рода инструкцию: команды запуска тестовых примеров.

В отличие от смарт-тегов Microsoft, в системе Oberon по умолчанию нет визуального выделения смарт-ссылок, не появляется меню (его надо предусматривать при разработке процедуры). К тому же распознаватели выполняются по унифицированной и упрощенной схеме: текст смарт-тега должен в точности соответствовать формату представления команды (это не произвольный текст). В то же время такие а-ля смарт-теги в системе Oberon гораздо проще создавать и активизировать, да и распознаются смарт-ссылки практически в любой точке интерфейса, даже на Рабочем столе.

К сожалению, как и в случае проектов NLS и Alto, беда этой системы заключается в том, что она на многие годы опередила свое время. К тому же, в отличие от старых добрых дней, теперь в коммерческих продуктах все реже можно встретить хотя бы упоминание о тех источниках, из которых были заимствованы идеи.

## Смарт-теги против макрокоманд

Казалось бы, зачем нужны смарт-теги, когда есть привычный гипертекст и отдельные программы? Ведь если надо что-то сделать, можно использовать соответствующее приложение. Ответ лежит на поверхности: опыт работы с Web-страницами показывает, что чем глубже по навигации размещена информация (а также действия), тем реже она оказывается востребованной. В случае смарт-тегов множество разрозненных операций переводится в унифицированный интерфейс всплывающих меню, которые в нужный момент всегда оказываются под рукой. Причем с их помощью можно не только быстро выполнять соответствующие тексту поисковые запросы и в общем случае любые сколь угодно сложные SQL-запросы, но и выполнять обработку информации вплоть до решения отнюдь не простых задач.

Приведу пример одного фрагмента, который, будучи превращен в гипертекст (в нашем случае — текст со сноской), лишь незначительно помогает в решении конкретной ситуации. А вот с помощью смарт-тегов такая задача может быть вполне эффективно автоматизирована (с учетом, разумеется, использования несложного лингвистического процессора, обеспечивающего адекватное распознавание запроса на естественном языке).

В конце XIX в. знаменитый английский писатель Льюис Кэрролл в своей книге «История с узелками» [4] приводит следующую историю-загадку: «Вот она, — сказала наконец тетушка. — Последняя запись относится к вчерашнему завтраку. Один стакан лимонада (Почему ты не можешь пить простую воду, как я?), три бутерброда (Горчицы, конечно, в них нет и в помине. Я прямо так и сказала девушке за прилавком, а она в ответ лишь вздернула подбородок. Удивительная дерзость!) и семь бисквитов. Итого 1 шиллинг и 2 пенса... — Постарайся вычислить, если не помнишь, — предложила тетушка. — Что ты заказывала на завтрак вчера, тебе известно. А вот запись о том, что ты заказывала позавчера — в первый день, когда мы отправились завтракать в кондитерскую: один стакан лимонада, четыре бутерброда, десять бисквитов. Итого 1 шиллинг и 5 пенсов.

С этими словами тетушка передала свои таблички Кларе. Сквозь слезы Клара даже не разглядела, что держит таблички вверх ногами»\*.

В данном тексте несложно распознать такие объекты, как числительные (записанные прописью и цифрами), а также связанные с ними продукты (стакан лимонада, бутерброд, бисквит) и денежные единицы (шиллинг, пенс и т.д.). Остается только встроить в обработчик смарт-тегов блок-решатель алгебраических уравнений, конвертер денежных единиц и активизировать вызов обработчика соответствующим словом (например, «итого»). Кстати, попробуйте решить ту же задачу самостоятельно. Чтобы сравнить, взгляните на более простую жизненную ситуацию (рис. 3), где требуется иметь под рукой удобный конвертер валют (запрашивает информацию в реальном времени из ЦБ РФ).

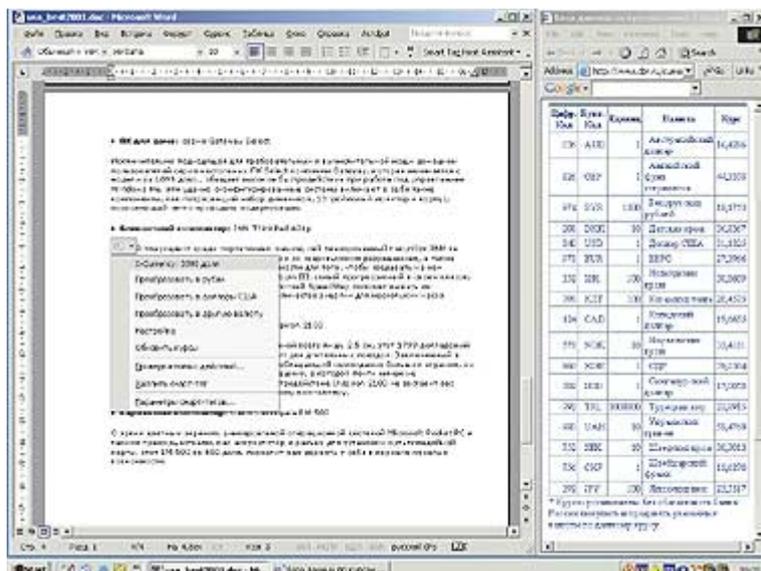


Рис. 3. Преобразование денежных величин с помощью смарт-тегов (контейнер X-Currency).

«Позвольте, но какая же тут революционность?» — словно вопрошает один из авторов критических статей по технологии смарт-тегов [5]. В самом деле, ведь если посмотреть на примеры применения смарт-тегов, то все они вроде бы могут быть довольно легко реализованы средствами уже имеющихся офисных приложений. По мнению другого автора [6], все то, о чем говорилось выше, делается с помощью известных средств программирования офисных приложений. Подводите курсор к выбранному слову, открываете нажатием «горячей клавиши» макрокоманду в контекстном меню и делаете все, что требуется. Оба автора, подробно излагая технические детали реализации смарт-тегов, словно сговорившись, недоумевают: зачем же нужен сам механизм?

Что ж, такое недопонимание простительно, и подобная точка зрения вполне имеет право на существование. Однако она игнорирует вполне очевидное преимущество технологии смарт-тегов по отношению к привычным макрокомандам офисных приложений (а ведь авторы вели речь именно о них). Как известно, макрокоманды опираются на простой принцип: сопоставление программного кода, написанного, как правило, на VBA, с конкретной комбинацией клавиш («горячие» клавиши).

Здесь вряд ли имеет смысл сравнивать функциональность и эффективность реализации смарт-тегов и макрокоманд (за счет применения компонентных технологий они имеют фактически одинаковые возможности для повышения упомянутых характеристик, хотя изначально некоторое преимущество все же на стороне смарт-тегов). Да и качество нынешней реализации технологии смарт-тегов никак не способно перечеркивать существенное преимущество ее использования по сравнению с макрокомандами.

Самое интересное, что режим выделения распознанных словокомплексов допускается отключать, и тогда чисто внешне работа смарт-тегов будет напоминать работу с макрокомандами. Кроме того, от большого числа «горячих» клавиш макрокоманд в принципе можно избавиться за счет реализации обобщенных макрокоманд с применением меню (т.е. промоделировав эту особенность смарт-тегов). Теперь все характеристики конкурирующих технологий приведены к одному знаменателю. Остается, пожалуй, последняя, ключевая: распознавание в случае смарт-тегов производится автоматически, а в случае макрокоманд — вручную. Ну а тут уже споры могут возникнуть в отношении удобства использования (что лучше: мышь или клавиатура) и качества реализации распознавания (в общем случае оно сводится к обработке запросов на естественном языке). А при работе с мнемоническими обозначениями (как тут не вспомнить венгерскую нотацию Саймони) «зацепиться» за текст распознавателю будет куда проще.

С системой Oberon я знаком без малого десять лет. Напомню, что команды в ней с точки зрения использования занимают промежуточное положение между смарт-тегами и макрокомандами. И надо сказать, что выполнять команды в Oberon без явного их выделения в тексте не совсем удобно. По сути, вы подобно саперу должны шаг за шагом прощупывать комбинации слов, чтобы выяснить, какие из них в данном контексте будут исполняться. Это может быть вполне приемлемо для программиста, для профессионала-виртуоза в обработке текстов, но массовому

пользователю не сулит ничего хорошего. Представьте себе на минутку, что было бы с нынешним Интернетом, если бы гиперссылки никогда не выделялись в тексте ни цветом, ни подчеркиванием. Кто нашел, куда нажимать, тот и молодец!

Очевидно, что для успешной работы смарт-тегов требуется, чтобы сервисные машины (engines), встроенные в соответствующие приложения (Word, Excel, Outlook, Internet Explorer, Acrobat и др.) и обеспечивающие управление обработчиками смарт-тегов, работали корректно. В принципе можно создавать и собственные сервисные машины, а не пользоваться штатными (если таковые для данного приложения имеются). Конечно же, идея и особенно нынешняя реализация технологии смарт-тегов далеки от совершенства. Но то, что это безусловный шаг вперед в развитии пользовательского интерфейса и собственно гипертекста, не вызывает никаких сомнений.

## Литература

1. Фрейберг П., Свейн М. Пожар в Долине. Группа ДАРНЭЛ, 2000.
2. Ичбиа Д., Кнеппер С. Билл Гейтс и сотворение Microsoft. Ростов-на-Дону: Феникс, 1997.
3. Wirth N., Gutknecht J. Project Oberon: The Design of Operating System and Compiler. Addison-Wesley, 1992.
4. Кэрролл Л. История с узелками. М.: Мир, 2000.
5. Малинин А. Смарт-теги в Microsoft Office фирменного и кустарного приготовления // Компьютер Пресс. 2001. № 11.
6. Колесов А. Smart Tags в офисных приложениях Microsoft // BYTE Россия. 2001. № 10.
7. Черняк Л. Ванневар Буш - царь науки // <http://www.computer-museum.ru/frgnhist/intern3.htm>
8. Черняк Л. Изобретатель мыши, но не только // <http://www.computer-museum.ru/frgnhist/mouse.htm>

## Web-ссылки

- Дуглас Энгельбарт - <http://www.bootstrap.org/dce-bio.htm>
- Тед Нельсон - <http://www.sfc.keio.ac.jp/~ted/>
- Чарльз Саймони - [http://www.edge.org/3rd\\_culture/bios/simonyi.html](http://www.edge.org/3rd_culture/bios/simonyi.html)
- Никлаус Вирт - <http://www.cs.inf.ethz.ch/~wirth/>
- Memex - <http://www.cs.brown.edu/memex/>
- XANADU - <http://www.xanadu.com/>
- Alto - <http://www.parc.xerox.com/hist-1st.html>
- Xerox Star - <http://xeroxstar.tripod.com/>
- Oberon System - <http://www.oberon.ethz.ch>
- Smart Tag Sales Kit - <http://www.exteria.ru/pr/20020118/>