

3. Косачев Ю. В. Исследование устойчивости динамической модели финансово — промышленной корпоративной структуры / Ю. В. Косачев // Эконом. и мат. методы. — 2000. — Т.36, № 1. — С. 126—142.
4. Гаращенко Ф. Г. Вступ до аналізу чутливості параметричних систем: Навчальний посібник. / Ф. Г. Гаращенко, О. Ф. Швець — К. : Видавничо-поліграфічний центр «Київський університет», 2006. — 115 с.
5. Башняков О. М. Практична стійкість та структурна оптимізація динамічних систем. / О. М. Башняков, Ф. Г. Гаращенко, В. В. Пічкур — К. : ВПЦ «Київський університет», 2000. — 197 с.

Work is sanctified to the construction of mathematical models of activity of financially-industrial structures. As the last the systems of differential come forward that develop. It is such systems which have switching and in which the dimension of phase space can change in course of time. On an example four a stage model of financial groups it is shown as such systems appear. Similar economic models lately become more actual and require the detailed research. It is shown, as possible mathematically to formulate (as optimization tasks) some economic problems, an answer for which is extraordinarily important for specialists.

Key words: *mathematical modeling, financially-industrial structure, system of differential equalizations that develops.*

Отримано 11. 05.10

УДК 004.5

Н. Н. Глибовец, д-р физ.-мат. наук, професор,

Л. О. Шыпович, магистр

Национальный университет «Киево-Могилянская академия». г. Киев.

СТАНОВЛЕНИЕ ТЕХНОЛОГИИ WEB 3.0

Проанализировано развитие Web-технологий и становление технологии Web 3.0. Выделены основные отличия технологии Web 3.0. Продемонстрировано использование новых подходов на примере создания Web-документа на базе онтологических объектов без HTML-разметки.

Ключові слова: *Web-(технології, ресурс, додаток, документ), онтологія.*

Вступление. Все чаще появляются сообщения о появлении того или иного Web-ресурса. На собрании 2007 года O'Reilly Web 2.0 Expo представителей Microsoft, Mozilla, Opera и Google принято решение, которое может стать судьбоносным. Речь идет об объединении усилий разработчиков по усовершенствованию работы Web-приложений. Основными проблемами данного направления названы медлительная и нестабильная работа Javascript, а также проблемы

безопасности. Очевидно, в дальнейшем разработчики браузеров сосредоточатся на решении этих проблем путем оснащения новыми возможностями Javascript и его стандартизации, особенно строгой стандартизации DOM-модели [1]. Там же зафиксирована формальная зрелость решений Web 2.0 и официально провозглашен *термин Web 3.0*, под которым понимают новое поколение онлайн-приложений с функциональностью, отвечающей требованиям времени. На нижнем уровне в платформу Web 3.0 входят программные интерфейсы систем конкретных поставщиков. Они объединены на втором уровне с помощью XML-технологий. На третьем прикладном уровне такие стандартизированные услуги объединяются провайдером конечных систем в целостные продукты [2].

Термин Web 3.0 и раньше появлялся в публикациях, но воспринимался скорее как научная перспектива. Со становлением технологии Web 2.0 стало ясно, что идеологи Web 3.0 не так уж далеки от реальности. На данном этапе эта технология находится в фазе зарождения. Поэтому целесообразно упорядочить и систематизировать разнообразную информацию о технологии Web 3.0 и выписать единый стандарт для Web-ресурсов этого поколения.

Развитие технологий Web. В 1990 году создан первый в мире Web-браузер, названный WorldWideWeb [3], способный отображать простые HTML-страницы и положивший начало технологии, которую несколько позже назвали Web 1.0.

В 1999 году появились блоги. Появление сайта Blogger.com превратило статический Интернет в средство активного общения. Появились и другие ресурсы, значительно отличавшиеся от предшественников. Новые ресурсы уже имеют такие технологии, как Web-сервисы, Ajax, синдикацию и дистрибуцию контента. Все большее распространение получают фолксономии, социальные сети, wiki. Проявились значительные отличия между Web-ресурсами нового и предыдущего поколений.

Чтобы понять разницу между ресурсами Web 1.0 и Web 2.0, целесообразно рассмотреть примеры. В таблице 1 представлены Web-ресурсы с подобными сервисами.

Таблица 1.

Ресурсы

Web 1.0	Web 2.0
Doubleclick	Google AdSense
Ofoto	Flickr
mp3.com	Napster, Gnutella
Britannica Online	Wikipedia
Персональные сайты	Блоги
Altavista.com, Rambler.ru, AOL.com, Google.com и др.	Answers.yahoo.com, Otvety.mail.ru

Охарактеризуем некоторые примеры таблицы 1 более детально.

Известный сервис Ofoto позволяет достаточно удобно манипулировать изображениями. Пользователи имеют возможность загружать большое количество изображений, сохранять их упорядоченными в папках со сложной иерархией. Данный Web-ресурс предоставляет даже определенный ограниченный инструментарий модификации изображений. Но становится очевидным ограниченность услуг. Пользователи желают не только сохранять свои изображения, но и предоставлять доступ к своим изображениям другим пользователям или группам пользователей с ранжированием прав доступа к материалам, разрешением вставки комментариев и т.п. Примером такого ресурса является известный сервис Flickr. Помимо описанных свойств этот ресурс также предоставляет пользователям возможность интегрирования экаунта (account) с другими Web-сайтами, то есть делает возможным обмен информацией между экаунтом пользователя на различных ресурсах.

Достаточно ярким примером отличия технологий Web 1.0 и Web 2.0 является организация сервисов хранения и систематизации универсальных знаний. Довольно длительное время популярным ресурсом был сайт известной энциклопедии Britannica.

Ресурс Britannica Online является сетевой энциклопедией. Со временем он все больше теряет популярность, поскольку не успевает пополнять информацию, которая появляется в современном мире стремительными темпами. Решением является интенсивное привлечение пользователей к пополнению контента, его редактирование и рецензирование. Это и сделано разработчиками Wikipedia. Интернет-сообщество создает контент на данном ресурсе, обрабатывает его, поддерживает в актуальном состоянии.

Еще одним интересным явлением в Web 2.0 является приход социальных поисковых систем на смену классическим. В социальной сети можно мгновенно получать ответы на вопросы от специалистов. Возможна дальнейшая поддержка диалога с экспертом, что также является полезным. Известными примерами социальной сети есть ресурсы Answers.yahoo.com и русскоязычный Otvety.mail.ru.

Сравнение технологий Web. Сравнить технологии Web 1.0, Web 2.0 и Web 3.0 можно по разным критериям. Мы выбрали следующие: участники, программное обеспечение (ПО), подход к организации контента, основные события, ценность и стоимость. Таблицы 2—6 характеризуют применение этих критериев.

Таблица 2.

Участники

Web 1.0	Web 2.0	Web 3.0
1. Разработчик и пользователь 2. Автор контента и читатель	1. Пользователь как собеседник 2. Читатель как соавтор 3. Сообщество	1. Пользователь принадлежит к группе разработчиков 2. Читатель как эксперт данных 3. Сообщество

В отличие от предыдущих по критерию участия основной идеей Web 3.0 является ранжирование пользователей и создание так называемых групп экспертов для упорядочения и оценки информации, поступающей от многочисленных пользователей. На самом деле эта идея не нова. Она успешно используется на многих форумах.

Таблица 3.

Программное обеспечение

Web 1.0	Web 2.0	Web 3.0
1. Писалось для ПК 2. Товар 3. Закрытые коды, API 4. Лицензирование, продажа 5. Привязка к устройствам 6. Направленность на изобретение 7. Запланирован релиз 8. Для просмотра контента нужна браузер	1. Пишется для Web 2. Сервис, применение 3. Открытые коды, API 4. Может быть бесплатным 5. Поверх устройств 6. Поиск применения 7. “Вечная бета” 8. Альтернативные устройства восприятия	1. Пишется исключительно для Web 2. Сервис, применение 3. Открытые коды принадлежат сообществам разработчиков 4. Бесплатное 5. Отсутствие привязки к устройствам или платформам 6. Реализация изобретения и поиск нового 7. Постоянная разработка открытыми сообществами разработчиков 8. Контент может представляться и отдельным применением

Контент на этом уровне используется не только как информация для предоставления пользователю, но и как данные, которыми могут обмениваться отдельные приложения. Основной идеей программной поддержки является вынесение всех приложений на уровень Web-приложений и преобразование Web-браузера на «окно» в любом применении (поскольку все ПО будет бесплатным).

Доступность ПО через браузер с любого компьютера, на данный момент актуальна.

Компания Microsoft заявляет что Web-технологии являются главным направлением развития на следующую пятилетку. Adobe приобретает компанию Macromedia и готовит к выпуску Web-версию программы Photoshop. Компания Google продвигает свой офисный пакет. Такое ПО имеет очевидные преимущества: централизованное хранение данных; доступность данных и средств их обработки с любого компьютера, подключенного к Интернету; совершенная система поставки ПО, по условиям которой у пользователя отпадает необходимость установки обновлений в системе и т.д. Проблемность использования указанного подхода заключается в том, что Интернет не всегда доступен в нужном объеме а за конфиденциальность, сохранность и стабильность информации никто не несет ответственности.

Обе проблемы мог бы решить онлайн-версия жесткий диск с удобным Web-интерфейсом и синхронизацией для оффлайновой работы, а также полный набор утилит для резервного хранения, шифрования данных, также, возможно, совместимость с мобильными устройствами, корпоративным ПО и т.п. [4].

Сейчас наблюдается не противостояние, а «слияние» двух платформ Web и Desktop. Имеем многочисленные примеры значительного распространения гибридных приложений, совмещающих свойства Web и Desktop приложений. Фактически ни одно из Web-приложений не может работать без Desktop-программы [4, 5]. Вспоминая Web-платформы, зачастую имеют ввиду браузер. Но этот же браузер поддерживает установку расширений и плагинов, которые в свою очередь являются настоящими Desktop-программами, хотя и созданы для работы в Web.

Аналогичное можно сказать и об обновлениях Desktop-widgets (виджеты). Они основаны на HTML и JavaScript как Web-приложения, однако имеют доступ к ресурсам локального компьютера и могут работать в оффлайне без потребности в Интернете.

С другой стороны, большинство современных Desktop-программ работает через Интернет. Например, фотоменеджер iPhoto — типичная Desktop-программа, но с подключением к ней специального модуля может считывать и генерировать RSS-feeds (RSS-фиды). Современные текстовые редакторы понимают HTML, могут генерировать Web-сайты и обновлять блоги. Графические редакторы могут сохранять изображения на удаленных серверах. Google Desktop ищет почтовые сообщения прямо с рабочего стола. Прослеживается тенденция к расширению функциональности Desktop-программ в сторону Web.

Таблиця 4.

<i>Контент</i>		
Web 1.0	Web 2.0	Web 3.0
1. Плата поставщику контента, использование добровольцев	1. Достояние одного сразу становится доступным остальным	1. Использование баз знаний
2. Односторонние ссылки	2. Автоматические двусторонние ссылки	2. Связь между объектами
3. Формы представления: персональные страницы, статический сайт	3. Формы представления: блоги, динамический сайт	3. Формы представления: свободный доступ к объектам
4. Адрес имеет страница сайта	4. Адрес имеет микроэлемент контента	4. Адрес имеет объект
5. Источник — ум производителя контента	5. Источник — коллективный разум, общественное мнение	5. Источник — коллективный разум, мнение экспертов
6. Меню навигации для работы с данными этого сайта	6. Интерфейс для работы с данными всей сети	6. Интерфейс для работы с данными глобальной базы знаний
7. Копирайт	7. «Свободная» лицензия	7. «Свободная» лицензия
8. Для восприятия контента необходимо посещение сайта, переходя по ссылке, адресу, закладке	8. Для восприятия контента не нужно посещать сайт — чтение RSS-ленты	8. Отпадает необходимость Web-сайта, бесплатное оформление представления объектов

В технологии Web 3.0 контент будет собираться, упорядочиваться и храниться в базах знаний, свободный доступ к которым будет иметь каждый пользователь. Знания в таких базах должны быть оформлены в виде упорядоченной совокупности объектов, описанных с помощью онтологий.

Таблиця 5.

<i>События</i>		
Web 1.0	Web 2.0	Web 3.0
1. Заказ и изготовление ПО	1. Сотрудничество через отдел технической поддержки ПО	1. Разработка ПО свободными сообществами разработчиков
2. Публикация контента авторами и восприятие его читателями	2. Взаимодействие, добавление свойств, ценностей, создание совместного контента каждым участником	2. Редактирование группами экспертов объектов, созданных каждым участником
3. Обращение к третьему лицу — посреднику для привлечения его ресурсов	3. Самообслуживание основанное на партнерской архитектуре сервиса — сервис лишь посредник между концами, который использует их собственные ресурсы	3. Самообслуживание основанное на партнерской архитектуре
4. Крупные, немногочисленные соглашения	4. Мелкие многочисленные транзакции	4. Мелкие многочисленные транзакции

При сравнении с основными событиями в Web 2.0 и Web 3.0 почти отсутствуют различия, за исключением управления контентом. Доступ к управлению контентом имеют специально сформированные группы экспертов, которые могут оценивать материалы в зависимости от собственного рейтинга, а также редактировать контент.

Таблица 6.

Ценность и стоимость

Web 1.0	Web 2.0	Web 3.0
1. Вся ценность в ПО: кто владеет ПО — тот зарабатывает 2. Интернет ценен как источник информации	1. Вся ценность в БД и сервисе: кто владеет БД и сервисом работы с ним — тот зарабатывает 2. Интернет ценен как способ коммуникации	1. Вся ценность в БЗ: кто контролирует БЗ — тот зарабатывает 2. Интернет ценен как источник коммуникации (доступ к другим субъектам)

С каждым годом информация изменяется, преобразуется и потребляется все легче, дешевле и быстрее. Это требует изменений от компаний, работающих на рынке. Организации, которые смогут справиться с таким потоком необобщенной информации в режиме реального времени, получат огромное конкурентное преимущество — аналогичное тому, которое получили компании, сумевшие первыми использовать хранилища данных и получить из них новые знания для принятия решений. Такие организации смогут находить новые возможности и реагировать на них подобно тому, как программы-трейдеры используют разницу курсов. В результате это вполне сможет стать новым стратегическим преимуществом ИТ. Стратегическим информационным активом при этом будет не программное обеспечение, автоматизирующее процесс, а знания (храняемые системой) корпоративных сообществ и интеграция данных и знаний в бизнес-процессы.

Отличия Web-страницы. Web-страница в условиях Web 1.0 была HTML-файлом, который также мог включать данные в формате XML1.0 и стиль отображения основных тегов в таблице стилей (CSS 1.0). Даже по определению Web-страница понималась исключительно как HTML-файл. В этом файле содержались и данные, и структура оформления, и стиль отображения. Даже при незначительных изменениях оформления страницы, необходимо было делать изменения в HTML-коде. Web-сайты в начале развития были небольшие по количеству страниц, поэтому такой подход к созданию отдельных Web-страниц вполне удовлетворял потребности. Но с дальнейшим развитием сети и ростом ее популярности возникла необходимость оперирования большими объемами информации, частого обновления самих данных, структуры или стилей отображения ин-

формации. Возникла необходимость отделения составляющих Web-страницы в отдельные части. На этом этапе оформление Web-страницы вынесли в таблицы стилей CSS, а данные редко отделяли (имеется в виду в XML-файл) от структуры отображения. Но в Интернет приходило все больше пользователей, появилась необходимость не только в выделении данных, но и обмена данными между разными приложениями. Формат XML удовлетворял эти потребности больше всех, почему и принят как стандарт обмена данными.

Именно поэтому в Web 2.0 основной частью Web-страницы является HTML-файл, основанный на стандарте XHTML, обязательными элементами также являются описание стилей страницы и собственно данные в формате XML 2.0. Эти обязательные элементы уже непосредственно упомянуты в новом определении Web-страницы. Составными частями являются разметка блоков информации (в формате XHTML), схема цветов и шрифтов, а также оформление блоков (CSS), собственно данные (XML).

Такие Web-страницы являются достаточно гибкими в плане изменения оформления и обмена данными. Благодаря новому подходу к организации Web-страницы стал возможен запуск многих Web-сервисов. Благодаря этому Web-страницы получили динамику, которая обеспечивается самими пользователями. Поскольку данные генерируются всей общиной пользователей, Интернет стал не только огромным хранителем информации, но и источником актуализированной и более упорядоченной информации. Стали больше внимания уделять созданию новых сервисов. Сеть Интернет перестала зависеть от финансовых вложений отдельных лиц или групп. В условиях Web 2.0 данные в сети создаются и принадлежат всему сообществу. Даже при отсутствии финансирования конкретного проекта он может поддерживаться и развиваться только пользователями. Именно поэтому сеть Интернет стала еще и источником объективной и непредвзятой информации, поскольку разработчики (владельцы) ресурсов все меньше зависят от финансовых давлений, которые ранее могли использоваться как инструмент манипуляций для «правильного» освещения информации.

Все же внесенные изменения оставили трудоемким процесс создания новой Web-страницы или реструктуризации существующей. При изменении разметки блоков приходится существенно изменять XHTML-файлы. Поэтому возникла идея отказа от разметки Web-документов. По сути, сами данные уже имеют определенную структуру (организация XML-файла), стиливое оформление вынесено в таблицы стилей (CSS). Остается лишь решить проблему объединения данных в структурные блоки.

На первый взгляд эта проблема кажется довольно сложной, но если проанализировать информацию в сети Интернет, можно выделить определенные виды данных. Новая информация, появляющаяся в сети, имеет структуру, повторяющую уже ранее использованную. Именно по этому принципу строятся онтологии. Этот же принцип можно использовать для создания Web-страниц, а браузеру передать не только информацию, но и тип, к которому она принадлежит. По сути, браузеру не обязательно знать, как будет выглядеть страница в целом, если есть установленный набор шаблонов отображения, которые можно расширять с помощью шаблонов отображения определенных структурных блоков.

На первый взгляд такой подход может показаться сужением возможностей Web-дизайнеров, но даже в дизайне есть определенные требования к разметке, которые необходимо соблюдать. А с помощью такого подхода можно упростить создание страницы, и не только не ограничить, а наоборот расширить возможности Web-дизайнеров. Ведь, тогда создавать Web-страницы можно будет не только используя собственную фантазию, а пользуясь разработками тысяч и сотен тысяч других разработчиков. Процесс реструктуризации Web-страниц занимает меньше времени, ведь необходимо будет лишь изменить представления. Процесс реструктуризации Web-страниц будет также занимать меньше времени, ведь необходимо будет лишь изменить тип отображения. Стили оформления (CSS) будут приписывать конкретным объектам данных, а не структурным элементам, как это делается на данный момент. Такой подход может действительно упростить генерирования новых данных и реструктуризации старых. Кроме того вся информация будет отвечать определенным структурам (описанным в шаблонах), таким образом будет решена проблема с благоустройством огромного количества информации, представленной в сети. Идея построения и использования семантического Web станет ближе.

Итак, в Web 3.0 предлагается отказаться от HTML-разметки, а использовать только данные и стиль их оформления. Поскольку все данные будут объектами определенных баз знаний, то соответственно будут иметь свою онтологию. На основе онтологии и таблицы стилей и предлагается оформлять отображения конкретного объекта.

Создание Web-страницы без использования HTML-разметки. Приведем пример создания синтаксического анализатора для преобразования XML-документа и таблицы стилей в формате CSS в полноценную Web-страницу формата HTML. Изменение структуры отображения должна происходить только с помощью изменения типа отображения в CSS-файле.

Поскольку на данный момент не существует браузера, который мог бы отображать данные без использования HTML-разметки, было решено создать синтаксический анализатор для преобразования в HTML-формат данных из XML, используя оформление, заданное в таблице стилей CSS.

При разработке синтаксического анализатора можно использовать фасадную модель (facade pattern) проектирования. Анализатор представим в виде трех структурных частей: анализатор, описание возможных структур отображения, реализация структур отображения («квази-онтологии»).

Благодаря использованию такой модели процесс добавления новых структур отображения достаточно прост. Необходимо лишь добавить реализацию новой структуры отображения. Отметим, что в этом случае не возникает никаких ограничений самой реализации. Разработчик получает от анализатора два массива данных: один — с данными, другой — со стилями отображения структурных частей данных. Для создания новой структуры отображения можно проводить любые манипуляции с этими массивами.

После добавления новой структуры отображения необходимо указать об этом в описании существующих структур, то есть просто добавить возможность выбора добавленной модели отображения. Например, описание трех структур отображения *table*, *news1*, *list-ol*, реализации которых находятся соответственно в функциях *get_tree()*, *show_items1()*, *get_tree_list()* может быть таким:

```
switch ($type){
    case 'table':
        echo "<table>";
        for ($i = 0; $i < count($data); $i++)
        {
            get_tree($data[$i], $css);
        }
        echo "</table>";
        break;

    case 'news1':
        echo "<div id='container'>";
        for ($i = 0; $i < count($data); $i++)
        {
            show_items1($data[$i], $css);
        }
        echo "</div>";
}
```

```
        break;
    case 'list-ol':
        echo <ol>;
        for ($i = 0; $i < count($data); $i++)
        {
            get_tree_list($data[$i], $css);
        }
        echo </ol>;
        break;
```

Асоциативные массивы *\$data* и *\$css* соответственно содержат данные и стили описания структурных частей данных.

Для возможности описания структурных частей данных предложены некоторые изменения с требованиями оформления CSS-файла. Например, представление данных задается в начале CSS-файла и отделяется от собственно стилей тремя звездочками (***) . Тогда представление должно быть указанным в поле *output-type* . А в поле *key-el* необходимо указать название элемента, на основе которого будет происходить деление на структурные части.

Разных вариаций отображения может быть столько, сколько их будет представлено в реализации отображений. На XML-файлы никаких ограничений не накладывается.

Заключение. Учитывая вышеизложенное можно выделить такие преимущества Web 3.0 по сравнению с предшественниками. Он проще и дешевле, поддерживает всеобъемлющий и круглосуточный онлайн, предоставляет полный контроль над информацией, и стандартизацию оформления данных.

С развитием Web 3.0 люди не будут «входить» в онлайн, как они это делают в наши дни, а будут постоянно там находиться. Тенденция прослеживается уже сейчас, поскольку много приложений Web 2.0 требуют присутствия и мгновенного реагирования. Очевидно, такая «зависимость» пользователей от Интернета будет увеличиваться в дальнейшем. Постоянный доступ будут обеспечивать мобильные устройства. Будут разработаны системы управления всеми видами персональной информации, в том числе невербальной. Появятся новые системы защиты информации, а также новые алгоритмы ранжирования информационных источников по различным параметрам.

Все основные объекты будут описаны в онтологиях, поэтому для создания нового экземпляра объекта не будет потребности в описании структуры объекта, а лишь в его визуальном представлении. Очевидно, что исчезнут и проблемы с кроссплатформностью и кроссбраузерностью [6]. Установятся единые (для каждой категории объектов) стандарты представления данных.

Стандартизація оформлення даних приведе до більш простого, швидкого пошуку та упорядкування інформації. Привнесення цих аспектів зовсім можливо вже в найближче час. На прикладі створення синтаксического аналізатора була показана можливість стандартизації даних за рахунок визначених онтологій, що робить можливим повне відокремлення даних від представлення та від джерела інформації. Відмова від навантажувальної вже зараз HTML-верстки пришвидшить упорядкування інформації в базах знань, що приведе до значущого упорядкування знань, якими володіє людство.

Список использованной литературы:

1. Havenstein Heather. Microsoft & Mozilla end browser wars: IE, Firefox, Opera & Google Reader teams unite. — PC Advisor. — Режим доступу : <http://pcadvisor.com/news/index.cfm?NewsID=9031>.
2. Fawzi Marc. Web 3.0 Definition, — Wordpress. — Режим доступу : <http://evolvingtrends.wordpress.com/Web-30>.
3. Alizar. Тим Бернерс-Ли не понимает, что такое Web 2.0. — Habrahabr. — Режим доступу : <http://www.habrahabr.ru/archive/text/2006/09/04>.
4. Toxziс. Обобщенная идея гибридизации приложений. — Habrahabr. — Режим доступу : http://www.habrahabr.ru/blog/i_am_clever/10236.html.
5. Simmons Brent. The end of “desktop vs. web apps”. — Inessential.com. — Режим доступу : <http://inessential.com/?comments=1&postid=3406>.
6. Havenstein Heather. Microsoft & Mozilla end browser wars: IE, Firefox, Opera & Google Reader teams unite. — PC Advisor. — Режим доступу : <http://pcadvisor.com/news/index.cfm?NewsID=9031>.

In the paper Web and Web 3.0 technology development analysis is made. The main differences of Web 3.0 from the previous technologies are outlined. The usage of new approaches are demonstrated on a Web-document creation on the basis of ontology objects without HTML-marks.

Key words: *Web-(technologies, resource, application, document), ontology.*

Отримано 23.06.10