

QUANTUMART



---

# QR7 Backend Explorer 2

## Руководство пользователя

v. 2.0

## Оглавление

Оглавление .....	2
Глоссарий .....	3
1 Общий обзор .....	4
2 Схема работы .....	5
3 Установка и удаление .....	5
4 Основы работы.....	6
4.1 Начало работы .....	6
4.2 Завершение работы.....	7
4.3 Настройка .....	7
4.4 Работа с соединениями .....	7
4.4.1 Создание соединения.....	7
4.4.2 Удаление соединения .....	9
5 Работа с объектами иерархии.....	9
5.1 Дерево проекта.....	9
5.1.1 Добавление и удаление элементов из дерева .....	10
5.1.2 Перемещение объекта страницы на уровень шаблона.....	11
5.1.3 Создание копий элементов. ....	11
5.2 Организация многопользовательской работы (система блокировок) .....	11
5.2.1 Get latest version .....	11
5.2.2 Check out .....	12
5.2.3 Check out recursive .....	14
5.2.4 Undo check out .....	14
5.2.5 Check in .....	14
5.3 Работа с несколькими элементами .....	15
5.4 Редактирование свойств объектов .....	15
5.4.1 Изменение имени элемента иерархии.....	17
5.4.2 Поля “Overridden / Overrides” .....	17
5.4.3 Изменение типа объекта.....	18
5.5 Редактирование параметров объекта.....	18
5.6 Использование ссылок .....	19
5.7 Редактирование форматов .....	20
5.7.1 Включение IntelliSense (“Update references”) .....	20
5.7.2 Drag-n-drop .....	21
5.7.3 Установка кода формата по умолчанию.....	21
5.8 Предпросмотр .....	21
5.9 Сборка.....	21
6 Локальная компиляция и отладка сайта.....	23
6.1 Папки ASP.NET .....	23
6.2 Файл web.config .....	23
6.3 Установка параметров для локального запуска сайта .....	24
6.4 Локальный запуск сайта.....	25
6.4.1 Запуск с использованием WEB-сервера Microsoft Visual Studio .....	25
6.4.2 Запуск с использованием внешнего WEB-сервера на примере IIS.....	26
7 Приложение.....	27
7.1 Структура хранения файлов на машине пользователя .....	27
7.2 Восстановление информации .....	28

## Глоссарий

- Add-In – см. QP7 Backend Explorer
- Backend – пользовательская система управления сайтом, основанная на QP7.Framework.
- CMS – Content Management System – Система Управления Контентом – общее название класса систем, позволяющих менять информацию (текстовую, графическую и т.п.) на сайте отдельно от систем её вывода.
- QP7 Backend Explorer – расширение Microsoft Visual Studio версий 2005 и 2008, позволяющее используя среду разработки работать с шаблонами, страницами, объектами и форматами сайта, основанного на QP7.Framework версии 7.6 и старше (младшие версии требуют QP7 Backend Explorer версии 1.6).
- QP7.Framework – CMS компании Quantum Art.
- IntelliSense – система автоматической разметки кода и всплывающих подсказок, интегрированная в Microsoft Visual Studio.
- Панель – область Microsoft Visual Studio, содержащая набор инструментов, объединенных в группу.
- Пользователь – человек, работающий с QP7 Backend Explorer.
- Сайт – совокупность веб-страниц, объединенных общим корневым адресом, и системой навигации. В данном документе рассматриваются только сайты, находящиеся под управлением CMS QP7.Framework.

## **1 Общий обзор**

QP7. Framework Visual Studio Add-In (Add-In) – это мощный программный продукт, позволяющий работать с шаблонами (templates), страницами (pages), объектами (objects) и форматами объектов (formats), находящимися под управлением CMS QP7.Framework, используя богатые возможности Microsoft Visual Studio.

Работая через Add-In, web-разработчик может пользоваться, к примеру, следующими инструментами, недоступными через WEB-интерфейс:

- Работа с сайтами, написанными на C# и VB.NET.
- Интерактивная проверка синтаксиса кода;
- IntelliSense – подсветка ключевых слов, а также система подсказок и авто-дополнений;
- Drag-n-Drop – возможность вставлять вызов элементов в код переносом мышкой из дерева объектов;
- Редактирование свойств объектов с помощью панели “Properties” Microsoft Visual Studio;
- Работа с несколькими соединениями одновременно;
- Возможность сложной многопользовательской работы (система рекурсивных блокировок);
- Выполнять локальную сборку WEB-сайта и пользоваться всеми преимуществами отладчика Microsoft Visual Studio;
- Использовать для локальной сборки (Build) и отладки как встроенный WEB-сервер Microsoft Visual Studio, так и внешний WEB-сервер, например, Internet Information Services (IIS).

## 2 Схема работы

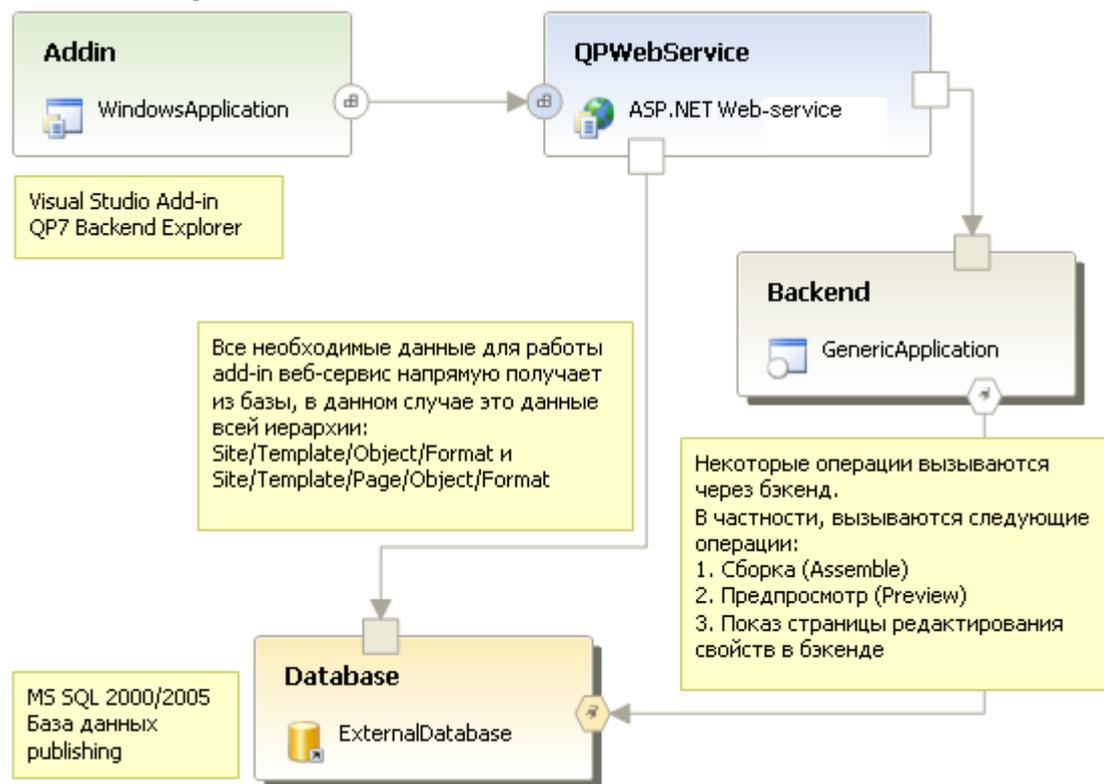


Рисунок 2-1

На рисунке 2-1 кратко показано место QP7 Backend Explorer в системе QP7.Framework. Как видно на рисунке, Add-in не обращается напрямую ни к базе данных, ни к CMS QP7.Framework. Все взаимодействие происходит только с WEB-сервисом, который либо самостоятельно обращается к базе данных, либо вызывает соответствующие функции CMS QP7.Framework.

## 3 Установка и удаление

Установка QP7 Backend Explorer осуществляется с помощью инсталлятора.

Деинсталляция может быть запущена одним из двух способов:

- Запуском инсталлятора. Если QP7 Backend Explorer уже установлен, инсталлятор предложит удалить его.
- С помощью диалога “Start | Control Panel | Add or Remove Programs” («Пуск | Панель управления | Установка и удаление программ»).

При удалении QP7 Backend Explorer файлы соединений (см. [Структура хранения файлов на машине пользователя](#)) могут быть удалены либо сохранены. Это определяется флажком в диалоге деинсталляции. В том случае, если файлы соединений сохраняются, работа с соединениями может быть продолжена при повторной установке Add-In в дальнейшем.

Для работы Add-In, кроме Visual Studio, требуется наличие доступа к установленному и настроенному веб-сервису QPWebService.

## **4 Основы работы**

### **4.1 Начало работы**

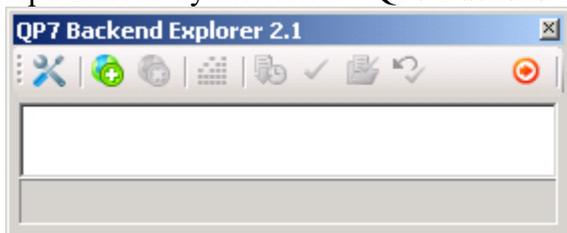
После установки Add-In в пункте меню “Tools” Visual Studio должен появиться новый пункт: “QP7 Backend Explorer 2” (рис. 4-1):



**Рисунок 4-1**

Если такой пункт меню не появился, прочтите документ “Known issues.txt”. Ярлык к нему можно найти в меню Start | Programs | Quantumart | QP7 Backend Explorer 2 | Known issues.

При вызове пункта меню “QP7 Backend Explorer 2” открывается новая панель (рис. 4-2):



**Рисунок 4-2**

Работа с данной панелью организована аналогично другим панелям Visual Studio (к примеру, Toolbox, Server explorer, etc.) Ее можно закреплять, перемещать, закрывать.

Закрытие панели не влечет прекращения работы Add-In. Закрытие панели означает просто её скрытие, для освобождения места на экране. В случае если панель была закрыта, доступ к ней всегда можно получить из меню Visual Studio - Tools | QP7 Backend Explorer 2.

В случае работы с одним соединением, при загрузке панели QP7 Backend Explorer это соединение будет сразу загружено. В случае одновременной работы с несколькими соединениями при загрузке панели QP7 Backend Explorer будут загружены только заголовки соединений, а загрузка данных произойдет только после выбора пользователем нужного соединения (рис. 4-3).



**Рисунок 4-3**

## 4.2 Завершение работы

Для завершения работы с Add-In, нажмите кнопку “Exit Add-In” . Если Вы собираетесь завершить работу с Visual Studio, сначала выйдите из QP7 Backend Explorer, а затем закрывайте Visual Studio.

В случае корректного завершения работы Add-In, при следующей загрузке будут открыты все соединения, файлы которых сохранены локально. Файлы соединения, при создании которого не был отмечен флажок “Save data offline” (см. «[Создание соединения](#)») не сохраняются локально.

Если у вас остались открытые файлы с несохраненными изменениями, будет показан диалог со списком таких файлов, в котором можно указать сохранять или отменить изменения, или продолжить работу с Add-In.

## 4.3 Настройка

При нажатии кнопки “Settings”  показывается диалог с тем же названием (рис. 4-4):

- “Location” определяет путь, по которому будут сохраняться файлы Add-In;
- “Block Solution Explorer, Class View and commands” позволяет блокировать появление на экране соответствующих окон Microsoft Visual Studio. По умолчанию этот параметр выключен.

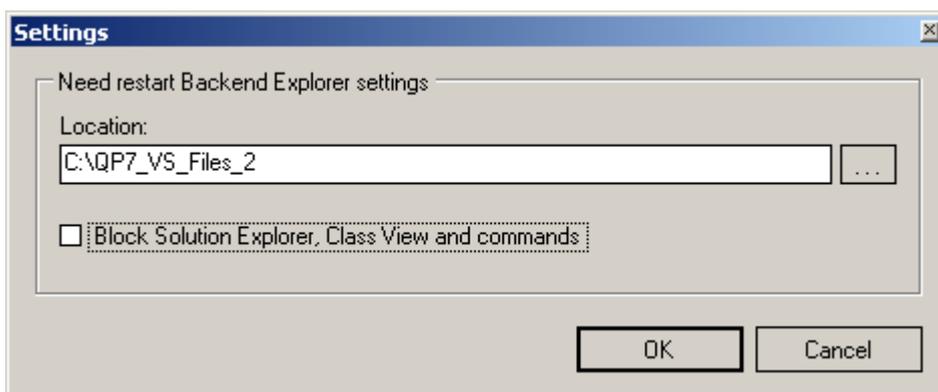


Рисунок 4-4

Для применения новых настроек необходимо перезапустить Visual Studio.

## 4.4 Работа с соединениями

QP7 Backend Explorer предоставляет возможность работы с неограниченным количеством соединений.

### 4.4.1 Создание соединения

Перед тем, как создавать в первый раз соединение в QP7 Backend Explorer, его можно настроить (см. [Настройка](#)). В противном случае будут использоваться настройки по умолчанию. В дальнейшем их можно изменить.

При нажатии на кнопку “Add new connection”  появляется панель создания нового соединения с QP7.Framework (рис. 4-5):

## Основы работы

### Работа с соединениями

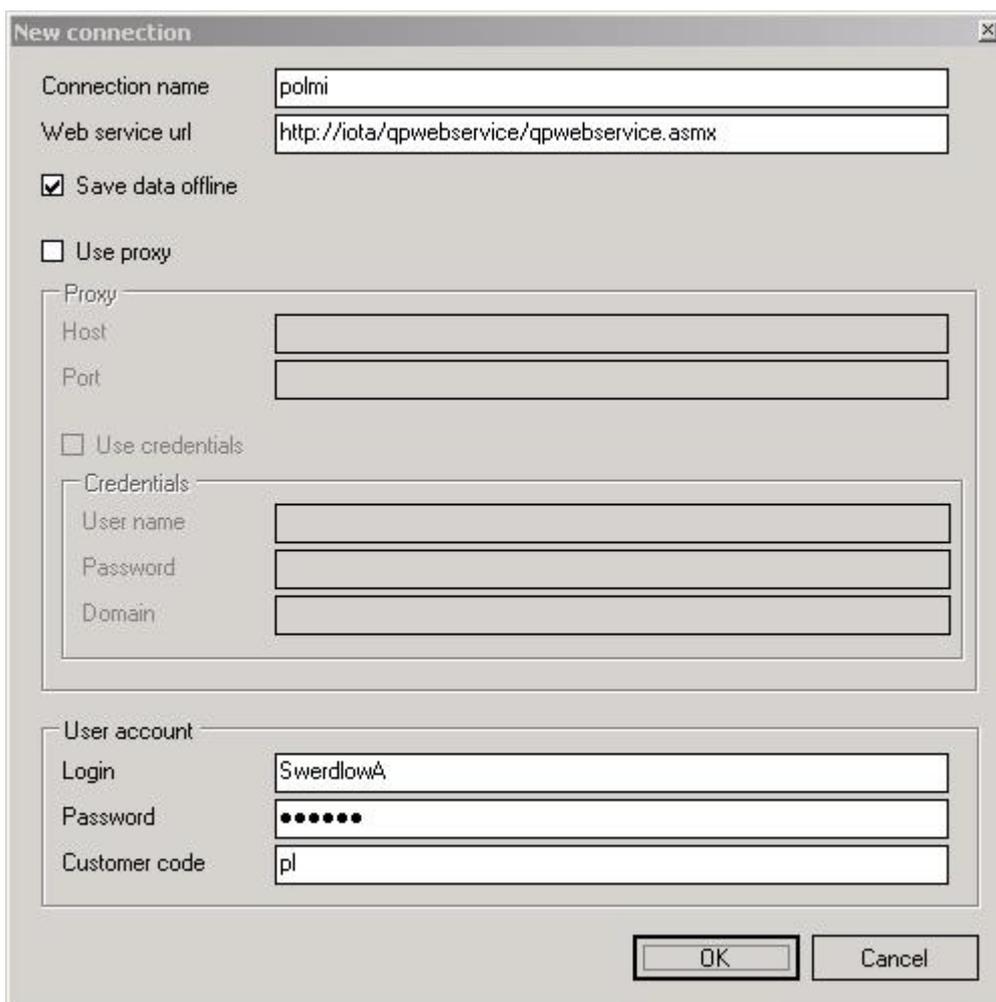


Рисунок 4-5

Панель содержит следующие параметры:

- “Connection name” – имя соединения, которое будет отображаться на панели QP7 Backend Explorer;
- “Web service url” – путь к файлу QPWebService.asmx WEB-сервиса;
- “Save data offline” – сохранять или нет файлы исходного кода страниц и объектов локально; если файлы сохранены, то соединение будет восстановлено при следующем запуске Add-In;
- “Use Proxy”, группа настроек “Proxy”, “Use credentials” и группа настроек “Credentials” – настройки доступа в Интернет;
- Группа настроек “User Account” – параметры доступа к QP7.Framework, доступ к которому настроен через данный веб-сервис.

После нажатия кнопки “OK” происходит загрузка данных из базы данных QP7.Framework, создание нового проекта (project) Visual Studio, генерация файлов исходного кода и присоединение их к созданному проекту.

Для создания соединения необходимо, чтобы версия Add-In не превосходила версию QPWebService, а та, в свою очередь, версию базы, т.к. невыполнение любого из этих

## Работа с объектами иерархии

### Дерево проекта

условий может привести к порче базы данных QP7.Framework в результате работы. В противном случае, пользователь увидит сообщение об ошибке версии (рис. 4-6).



Рисунок 4-6

После добавления в QP7 Backend Explorer соединения с QP7.Framework, на панели показывается дерево соединения (рис. 4-7):

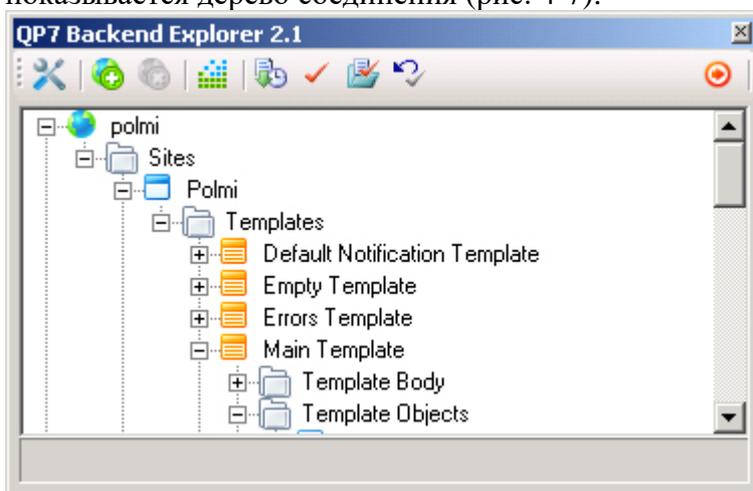


Рисунок 4-7

Для того чтобы выбрать соединение, нужно выделить в дереве его, либо любой зависимый от него (вложенный в него) элемент дерева.

#### 4.4.2 Удаление соединения

Чтобы удалить соединение, надо выбрать соответствующий элемент в дереве  , а затем нажать кнопку “Delete connection”  на панели инструментов QP7 Backend Explorer. Вместе с удалением соединения производится попытка удаления всех сохраненных локально файлов.

## 5 Работа с объектами иерархии

### 5.1 Дерево проекта

Дерево проекта представляется в виде набора вложенных группирующих и функциональных узлов.

Группирующие узлы показываются в дереве символом «папки» . К группирующим узлам дерева относятся узлы “Sites”, “Templates”, “Template body”, “Template pages” и другие узлы, присутствующие в любом соединении и определяющие его структуру. В контекстном меню таких элементов нет пункта “Open”, то есть их нельзя ни коим образом

## Работа с объектами иерархии

### Дерево проекта

---

открыть для редактирования. Их также нельзя удалить. Большинство таких узлов позволяет создать новый дочерний элемент. Любой группирующий узел содержит функциональные узлы.

Функциональные узлы можно редактировать. Большую часть функциональных узлов можно удалять.

Общая структура дерева такова:

- В корне дерева находятся соединения;
- Следующий уровень состоит из сайтов данного соединения
- На следующем уровне находится группирующий узел “Templates”, содержащий шаблоны сайта. Всегда присутствует шаблон с именем “Default Notification Template”. Каждый шаблон содержит три группирующих узла:
  - Template Body – определяется код шаблона:
    - Presentation – псевдо-HTML код, определяющий внешний вид страницы
    - Code Behind – код на C#, VB.NET
  - Template Objects – объекты шаблона
  - Template Pages – страницы шаблона, которые содержат в себе объекты страницы

Объекты шаблона и страницы в плане функциональности и вариантов использования одинаковы:

- Каждый объект может содержать форматы; любой формат объекта содержит вложенные элементы, через которые можно получить доступ к коду формата:
  - Presentation
  - Code Behind
- Объекты типа “Publishing Container” содержат еще одну вкладку: “Template Object Fields” или “Page Object Fields” (для объектов шаблона и страницы соответственно). Там содержатся все поля контента, включая служебные, с которым связан объект

#### 5.1.1 Добавление и удаление элементов из дерева

В некоторые узлы дерева, обозначенные символом «папки»  можно добавлять дочерние элементы. Для того чтобы добавить элемент, пользователь должен выбрать пункт “Create new element” в контекстном меню этого узла. Дальнейший диалог зависит от того, куда именно пытается добавить элемент пользователь.

При создании нового объекта, он всегда создается с форматом по умолчанию. Узлы “Presentation” и “Code behind” этого формата содержат код по умолчанию. Код различается - для форматов объектов типа “Publishing Container”, “Publishing Form” и форматов объектов всех остальных типов. Также он различается в зависимости от языка. Также формат может копироваться из родительского объекта, если он выбран и установлен соответствующий флаг в диалоге создания.

Шаблоны кода хранятся в папке QP7 Backend на сервере.

Для того чтобы удалить элемент дерева, пользователь должен выбрать пункт “Delete” контекстного меню этого элемента. Можно удалить практически любой элемент не являющийся папкой, за исключением сайта, дефолтного шаблона уведомлений, полей и представлений формата “Presentation” и “Code Behind”.

## Работа с объектами иерархии

Организация многопользовательской работы (система блокировок)

---

Вместе с элементом дерева удаляются и все зависимые от него (вложенные в него) элементы.

### 5.1.2 Перемещение объекта страницы на уровень шаблона.

Add-In позволяет также делать объекты страницы доступными из любого формата сайта, делая их объектами содержащего шаблона.

Чтобы сделать объект страницы объектом шаблона, нужно выбрать в контекстном меню объекта пункт “Promote to template level”. Объект страницы станет объектом шаблона, которому принадлежит страница.

Вызовы объекта из форматов объектов страницы останутся корректными.

### 5.1.3 Создание копий элементов.

Add-In позволяет легко создавать копии объектов и страниц шаблонов, либо шаблонов.

Чтобы скопировать один из этих объектов, нужно выбрать в контекстном меню этого элемента пункт “Create like”

При нажатии на эту кнопку появится диалог создания копии объекта, страницы или шаблона, аналогичный диалогу создания нового объекта или страницы.

Копируется только текущий элемент (если у него есть формат по умолчанию, то вместе с ним). Другие элементы (страницы копируемого шаблона, форматы копируемого объекта) не копируются.

## 5.2 Организация многопользовательской работы (система блокировок)

В Add-In реализована возможность одновременной работы над проектом нескольких программистов. Для этого создана система блокировок.

Ограничения текущей версии: все пользователи должны принадлежать к группе администраторов в QP7.Framework.

Многопользовательская работа в Visual Studio семантически аналогична работе с использованием системы контроля версий от компании Microsoft - Visual Source Safe. Пользователь может выполнить одно из четырех действий синхронизации: “Get latest version”, “Check out”, “Undo check out”, “Check in”. После каждого такого действия, если оно было рекурсивным (то есть применялось к выделенному элементу и всем зависимым от него, вложенным в него элементам) происходит обновление этих элементов дерева. В противном случае, происходит обновление только текущего элемента.

### 5.2.1 Get latest version

Для того чтобы получить текущую версию (со всеми изменениями, сделанными в нем на данный момент) некоторого объекта иерархии, пользователь должен выбрать нужный объект, открыть (щелчком на нем правой кнопки мыши) контекстное меню и выбрать в нем пункт «Get latest version» (рис. 5-1):

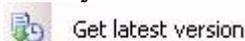


Рисунок 5-1

Если оставить флажок «recursive» в появившемся диалоге пустым, то будет обновлен только текущий элемент иерархии; если же отметить его, то будут также обновлены все зависимые от него элементы дерева (все элементы, вложенные в него) (рис. 5-2):

## Работа с объектами иерархии

Организация многопользовательской работы (система блокировок)

---



Рисунок 5-2

Другим способом получить текущую версию выбранного элемента, является нажатие кнопки «Get latest version» на панели инструментов Add-In. В этом случае по одному нажатию кнопки панели инструментов будет выполнено действие «Get latest Version» для выбранного в дереве элемента, аналогичное контекстному меню.

### 5.2.2 Check out

По умолчанию, соединение открывается только для чтения. Произвести какие-либо изменения невозможно.

Чтобы изменить какие-либо объекты шаблонов, пользователь должен сначала наложить на них блокировку. Для этого нужно в контекстном меню нужного объекта шаблона выбрать пункт «Check Out» (рис. 5-3):

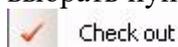


Рисунок 5-3

Флажок «recursive» работает аналогично флажку в диалоге «Get latest version» (рис. 5-4):

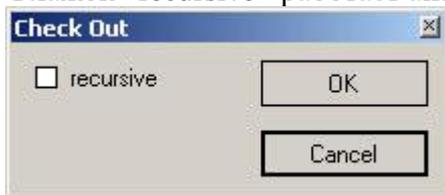


Рисунок 5-4

При выполнении «Check Out» в Visual Studio пользователя загружаются последние версии объектов иерархии.

После выполнения «Check Out» цвет значков заблокированных элементов изменяется на серый и к ним добавляется иконка замка (рис. 5-5):

## Работа с объектами иерархии

Организация многопользовательской работы (система блокировок)

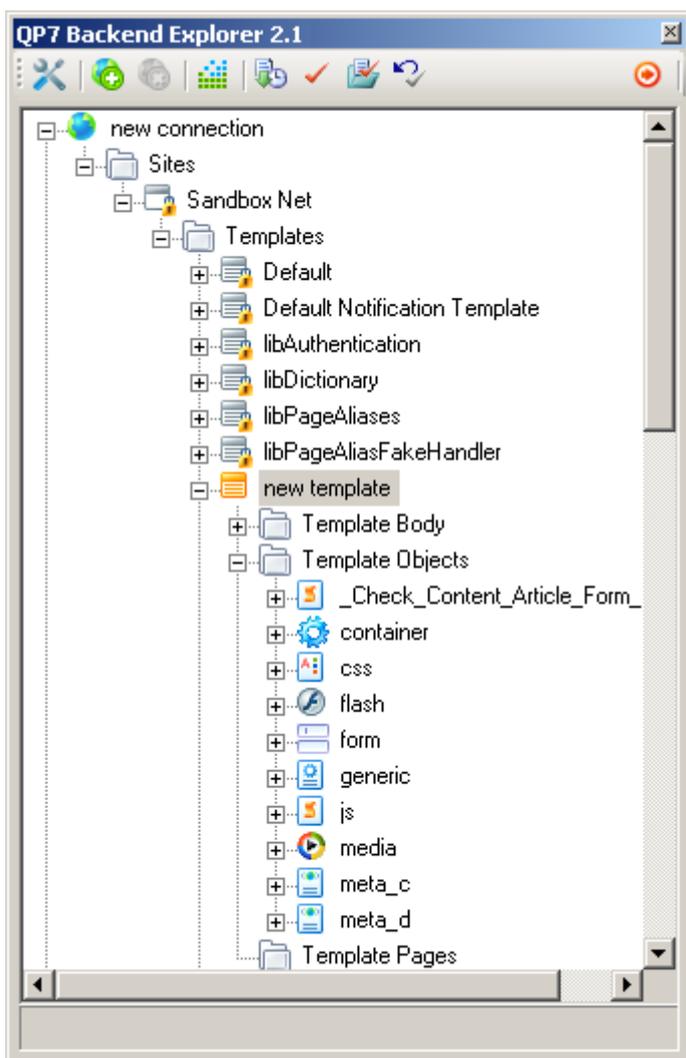


Рисунок 5-5

Заблокированные другими пользователями элементы также отображаются серыми иконками, но вместо значка замка на них появляется иконка пользователя (рис. 5-6):



Рисунок 5-6

Если элемент заблокирован другим пользователем, то в меню блокировки остается всего два пункта (рис. 5-7):



Рисунок 5-7

## Работа с объектами иерархии

Организация многопользовательской работы (система блокировок)

---

Также в этом случае останутся активными только две кнопки панели инструментов: “Get latest version” и “Check Out”.

При выборе пункта “Check out” сначала происходит обновление элемента иерархии; затем, если со времени предыдущего обновления объект был разблокирован, то произойдет его блокировка текущим пользователем; иначе будет произведено просто обновление, аналогично действию “Get latest version”.

Другим способом выполнить “Check Out” является кнопка на панели инструментов Add-In.  Она функционирует аналогично кнопке “Get Latest Version” этой панели: выполняет в один клик “Check Out” элемента, аналогично действию соответствующего пункта контекстного меню.

### 5.2.3 Check out recursive

Данный пункт контекстного меню (рис. 5-8) дополняет пункт “Check out”, позволяя в один клик выполнить Check Out выделенного элемента и всех зависимых от него (вложенных в него) элементов, минуя дополнительный диалог “Check out”.

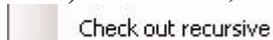


Рисунок 5-8

### 5.2.4 Undo check out

Отменяет блокировку элемента дерева. Диалог “Undo check out” работает аналогично предыдущим. Кнопка на панели инструментов  - также работает аналогично другим кнопкам.

При выполнении данной команды снимается блокировка элементов на сервере и производится откат всех изменений, сделанных пользователем на клиенте (в QP7 Backend Explorer).

### 5.2.5 Check in

Сохраняет изменения пользователя в базу данных. Сборка на сервере при этом не происходит.



Рисунок 5-9

Флажок “recursive” работает аналогично флажкам в диалогах “Check in”, “Get latest version”, “Undo check out”. Флажок “keep check out” (рис. 5-9) работает следующим образом: если оставить его неотмеченным, то обновленные объекты будут разблокированы; если же отметить его, то они останутся заблокированными.

Кнопка на панели инструментов  позволяет выполнить “Check In” элемента аналогично действию контекстного меню.

Если среди обновляемых файлов будут несохраненные, то QP7 Backend Explorer предложит их сохранить.

### 5.3 Работа с несколькими элементами

Возможно применение одного из действий “Check In”, “Check Out”, “Undo Check Out”, “Get Latest Version”, “Delete” к нескольким элементам, находящимся в дереве на одном уровне: нескольким страницам одного шаблона, нескольким объектам одной страницы и т.п. Для этого нужно выбрать сразу несколько элементов в окне QP7 Backend стандартным для Windows-приложений способом: выделением с нажатой кнопкой “Ctrl” или “Shift” и выбрать соответствующее действие в контекстном меню или на панели инструментов Add-In.

Все операции с множественными элементами выполняются рекурсивно.

### 5.4 Редактирование свойств объектов

Любой объект иерархии, кроме папки, обладает некоторыми свойствами. Эти свойства можно редактировать (если объект заблокирован) с помощью стандартной панели свойств Visual Studio 2005.

Эту панель можно открыть, выбрав пункт “Open” в контекстном меню нужного элемента иерархии (рис. 5-10):

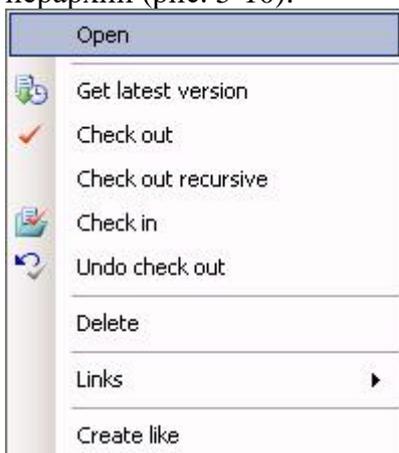


Рисунок 5-10

Если свойства текущего объекта не были показаны, то стоит еще раз выделить нужный элемент в дереве.

Свойства сайта, к примеру, выглядят на этой панели примерно следующим образом (рис. 5-11):

## Работа с объектами иерархии

### Редактирование свойств объектов

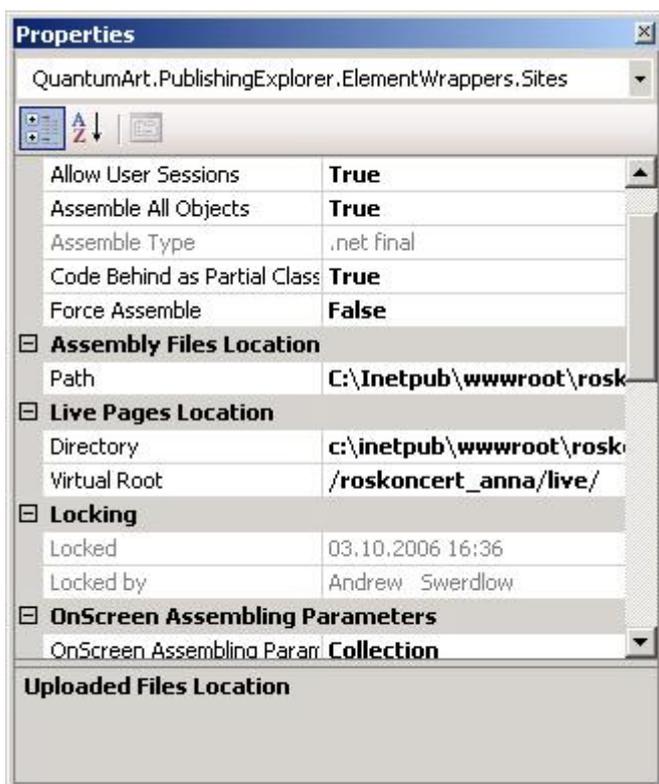


Рисунок 5-11

Работа со свойствами объекта иерархии происходит стандартными для Visual Studio методами.

В том случае, если свойства представлены не по категориям, а в алфавитном порядке, имеет смысл переключить представление нажатием самой левой кнопки этой панели (рис. 5-12):

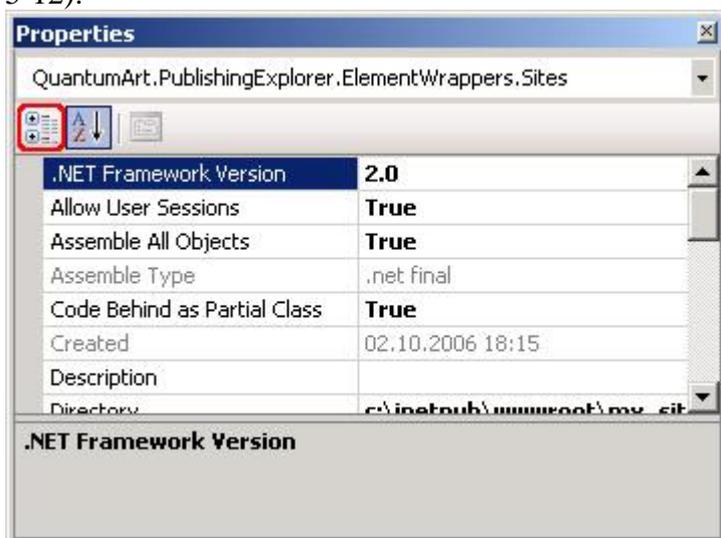


Рисунок 5-12

После изменения свойства (оно изменяется при нажатии клавиши Enter, а также при выходе из поля (клавиша Tab или действие мышью)) сразу же проверяется допустимость

данного значения. Если значение не допустимо, то выдаётся следующее предупреждение (рис. 5-13):

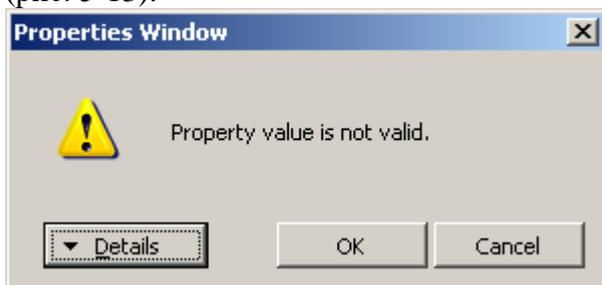


Рисунок 5-13

В данном случае, можно посмотреть название поля (кнопка Details), продолжить редактирование поля (кнопка OK), либо оставить старое значение в поле (кнопка Cancel).

#### 5.4.1 Изменение имени элемента иерархии

В дереве панели QP7 Backend Explorer показываются имена элементов иерархии, заданные одним из параметров “Format Name”, “Object Name”, “Page Name”, “Template Name”, “Site Name” (соответственно, для формата, объекта, страницы, шаблона или сайта). Поэтому при изменении этого параметра через панель “Properties” происходит обновление дерева. Как и при редактировании других параметров, это действие выполняется только на машине пользователя, для обновления имени элемента иерархии на сервере необходимо выполнить [“Check In”](#).

#### 5.4.2 Поля “Overridden / Overrides”

Add-In предоставляет возможность определить, переопределен ли объект шаблона на какой-либо из страниц. Если да – то в поле “Overridden” панели параметров Visual Studio будет стоять “True” (рис. 5-14).



Рисунок 5-14

Аналогичное поле есть и в параметрах объекта страницы (overrides). Такое поле показывает, что существуют объекты страницы, которые переопределяют объект шаблона.

### 5.4.3 Изменение типа объекта.

Add-In позволяет изменять тип объекта. Для того, чтобы изменить тип объекта, нужно раскрыть поле “Object Type” объекта, для которого сделан Check Out (рис. 5-15).

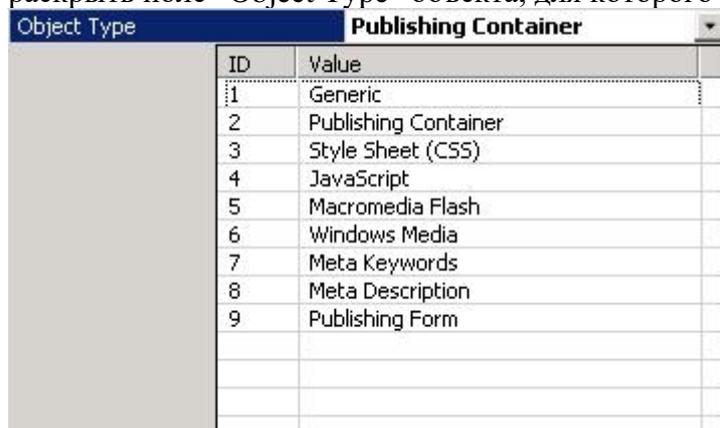


Рисунок 5-15

Данная операция является не откатываемой, т.е. операция “Undo Check Out” не отменит изменение типа объекта.

В случае изменения типа объекта на “Publishing Container” или “Publishing Form” пользователь увидит дополнительный диалог, в котором ему будет предложено выбрать контент, с которым будет связан данный объект (рис. 5-16).

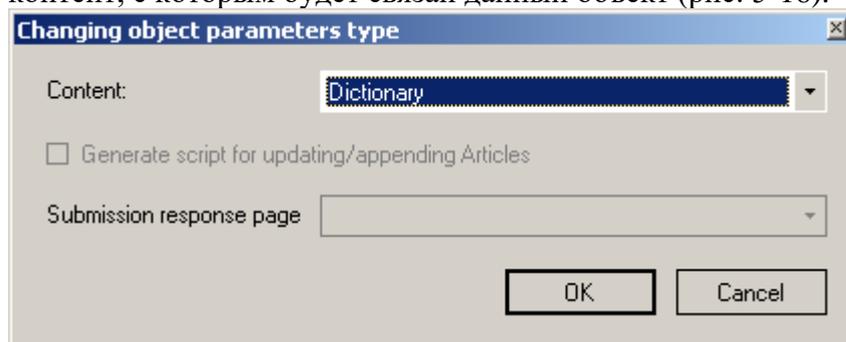


Рисунок 5-16

Для объекта типа “Publishing Form” пользователю будет также предложено выбрать страницу, на которую пользователь будет переадресован после окончания работы с формой (поле “Submission response page”), а также создать скрипт для работы со статьями контента в формате по умолчанию (флажок “Generate script for updating/appendng Articles”).

## 5.5 Редактирование параметров объекта

Панель свойств объекта для объекта любого типа содержит информацию об имени объекта, его формате по умолчанию, типе и других параметрах (рис. 5-17):



Рисунок 5-17

Часто используемые и сложно структурированные объекты, такие как “Publishing Container” и “Publishing Form” обладают дополнительными параметрами.

Получить доступ к ним можно, нажав на раскрывающую кнопку рядом с надписью Collection для параметра “Object Parameters”.

Для объектов любого типа, кроме “Publishing Container” и “Publishing Form” нажатие на раскрывающую кнопку не произведет никакого действия.

Поля диалога аналогичны соответствующим полям web-интерфейса QP7.Framework.

Работа с полями Default Values и Default order (диалог редактирования параметров объекта “Publishing Container”) аналогична работе с любым элементом управления Grid. Для вставки строки достаточно просто ввести какое-либо значение в строку помеченную звездочкой. Для удаления строки её необходимо выделить, нажав левой кнопкой мыши на самой левой колонке в данной строке, и нажать клавишу Delete.

## 5.6 Использование ссылок

Для взаимодействия с другими инструментами работы с проектом в контекстное меню Add-In встроены ссылки на эти инструменты (рис. 5-18).

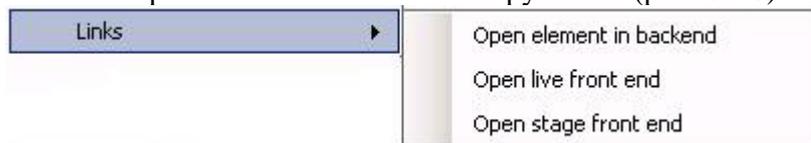


Рисунок 5-18

Ссылка “Open element in backend” позволяет открыть элемент для редактирования через стандартный веб-интерфейс QP7.Framework. При этом для доступа к Backend’у используются указанные пользователем параметры доступа и указанный в конфигурационном файле QPWebService адрес Backend’а.

Ссылки “Open live/stage front end” позволяют просмотреть front end сайта.

Для объектов типа “Publishing Container” в списке присутствует дополнительная ссылка (рис. 5-19)

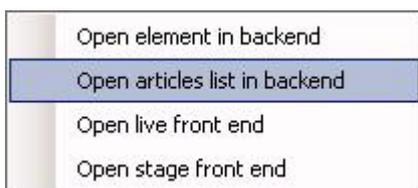


Рисунок 5-19

Ссылка “Open articles list in backend” позволяет открыть в QP7.Framework список статей контента, связанного с данным объектом. Ссылки открываются в браузере по умолчанию.

## 5.7 Редактирование форматов

Центральным моментом создания сайта на QP7.Framework является редактирование форматов объектов. И именно по удобству редактирования форматов объектов Add-In многократно превосходит стандартные средства.

Для того чтобы открыть формат на редактирование, нужно дважды кликнуть на него в дереве проекта, либо выбрать пункт “Open” в контекстном меню.

Если пользователь начинает редактировать формат, то на экране появляется диалог, в котором можно подтвердить либо отменить действие. В случае подтверждения автоматически выполняется не рекурсивная операция “Checkout” для данного формата.

### 5.7.1 Включение IntelliSense (“Update references”)

В случае если система IntelliSense Visual Studio 2005 не работает для типов, определенных в DLL-файлах сборок сайта (находящихся в папке “Bin”), нужно выбрать в контекстном меню этого сайта пункт “Update references” (рис. 5-20):

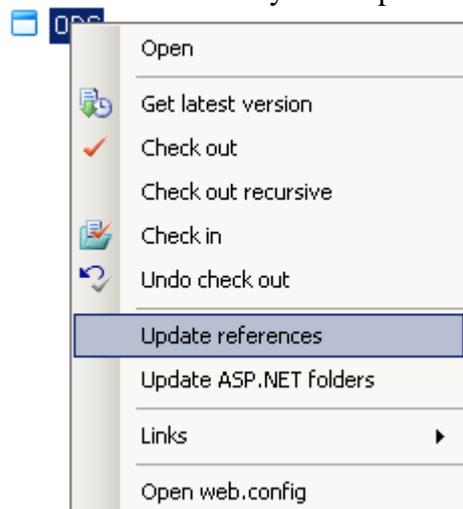


Рисунок 5-20

В частности, при выполнении данного действия подключаются все .NET сборки, находящиеся в файлах с расширением .dll, и находящиеся в папке “Bin” сайта.

При [создании соединения](#) автоматически выполняется данная операция.

Данная операция может понадобиться в том случае, если состав dll-ек или их содержание изменилось на сервере.

### 5.7.2 Drag-n-drop

В Add-In реализована операция Drag-n-drop. В данном случае – при переносе объекта из дерева в код вставляется вызов элемента иерархии.

Из любого формата можно вызвать объект любого шаблона или его форматы; из формата объекта страницы можно также вызвать объекты этой страницы или их форматы.

Drag-n-drop позволяет вставить в код вызовы только тех объектов, которые могут быть корректно вызваны из кода.

Синтаксис вызова объекта различен для Presentation и Code Behind.

Для Presentation: `<qp:placeholder calls="Имя объекта" runat="server"/>`

Для Code Behind: `ShowObject("Имя объекта", this);`

### 5.7.3 Установка кода формата по умолчанию

Add-In позволяет заменить текущий код формата на код по умолчанию.

Для того, чтобы заменить код Presentation или Code Behind на код по умолчанию, нужно выполнить Check Out содержащего их формата, а затем выбрать в контекстном меню элемента пункт “Set default” (рис. 5-21):

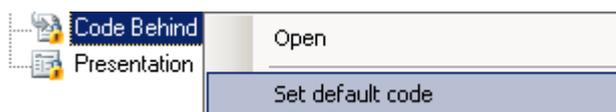


Рисунок 5-21

Код различается для Presentation и Code Behind, а также для формата объектов типа “Publishing Container” и формата объектов всех остальных типов. Также он различается в зависимости от языка формата.

## 5.8 Предпросмотр

В Add-In реализована такая функция QP7.Framework, как предварительный просмотр (preview). Для того чтобы включить режим предварительного просмотра объекта, пользователь должен выбрать в контекстном меню пункт “Preview”. Предпросмотр доступен для объектов и шаблонов и функционирует аналогично preview QP7.Framework.

В данном случае эта функция реализована как прямой вызов её из QP7.Framework. Поэтому в случае возникновения ошибки или неправильной работы проверьте работу данной функции в QP7.Framework.

## 5.9 Сборка

Чтобы сгенерировать исходный код страницы или объекта на WEB-сервере нужно, выбрав в дереве соответствующий элемент нажать кнопку «Assemble» панели инструментов Add-In . По идеологии текущей версии выполняется сборка только выбранного элемента и всех дочерних. Т.е., в отличие от предыдущих версий, не происходит пересборки всего шаблона при запросе на пересборку отдельного объекта и так же для страницы – пересборка отдельного ее объекта не требует пересборки страницы и всех объектов целиком.

Если нажать кнопку «Assemble» при выбранном сайте, то будет собран весь сайт, то есть все страницы всех шаблонов. Если же нажать её при выбранном соединении, то будут собраны все сайты данного соединения.

## Работа с объектами иерархии

### Сборка

В том случае, если сборке подлежат узлы, заблокированные пользователем, ему будет предложено выполнить check in этих объектов (рис. 5-22), если они были изменены:

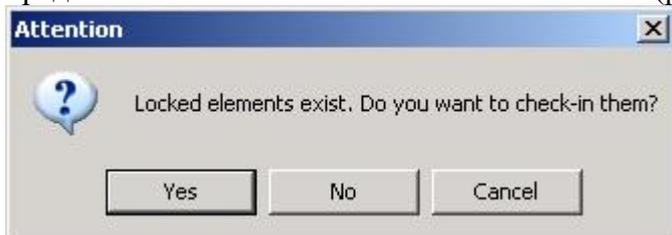


Рисунок 5-22

Если элемент является объектом или форматом шаблона, то проверяется на наличие заблокированных элементов весь текущий шаблон со всеми элементами, входящими в него. Если элемент является объектом или форматом страницы, то проверяется на наличие заблокированных элементов вся текущая страница со всеми элементами, входящими в нее. В остальных случаях проверяется текущий элемент. Если среди обновляемых файлов будут несохраненные, то QP7 Backend Explorer предложит их сохранить.

В данном случае функция «Assemble» реализована как прямой вызов её из QP7.Framework. Поэтому в случае возникновения ошибки или неправильной работы проверьте работу данной функции в QP7.Framework.

В процессе сборки сайта появится диалог с логом сборки сайта. Если нажать кнопку “Cancel”, то будет завершена текущая операция сборки, после чего сборка будет прекращена. Если во время процесса отмены нажать кнопку ”Terminate” (рис. 5-23), то Add-In прекратит взаимодействие с QPWebService, не дожидаясь ответа.

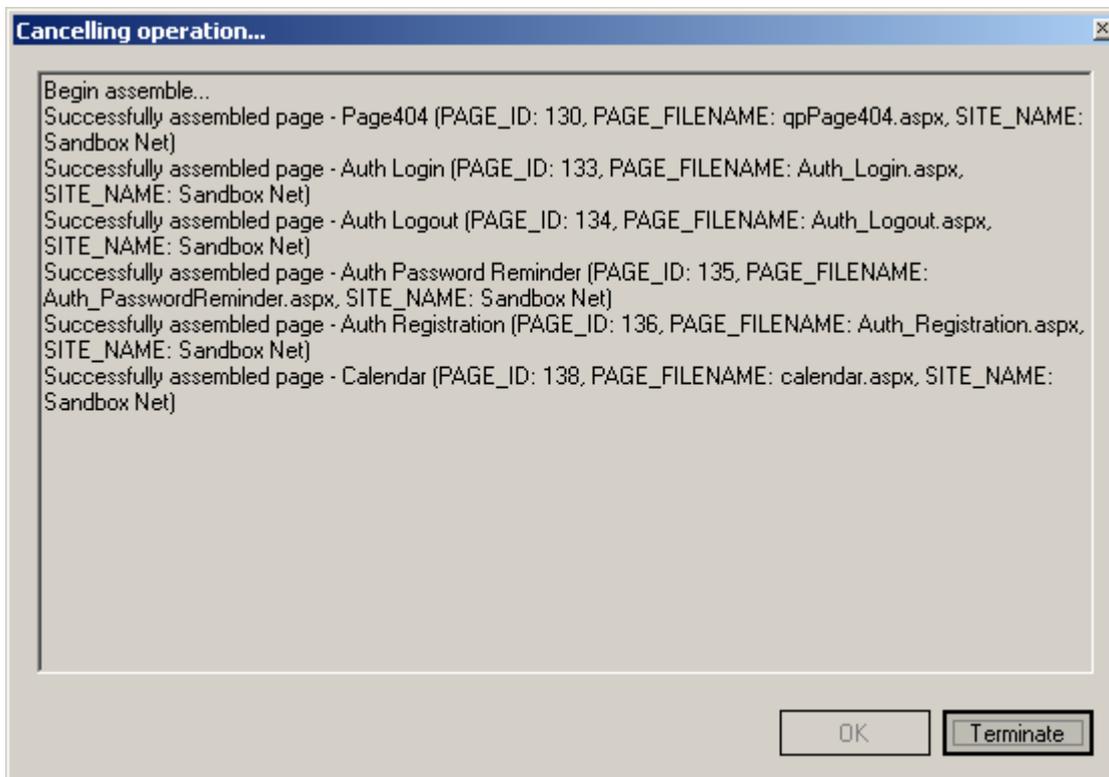


Рисунок 5-23

## 6 Локальная компиляция и отладка сайта

Главное отличие QP7 Backend Explorer версии 2 от предыдущих версий - создание на локальном диске разработчика файловой структуры сайта, полностью идентичной структуре на WEB-сервере клиента. При каждом изменении исходного кода через редактор, а так же при выполнении операций «Get Latest Version» и «CheckOut» он изменяется и на локальном диске. Это дает возможность выполнять локальную сборку (Build) и отладку сайта. При этом используются все возможности отладчика Microsoft Visual Studio - breakpoints, окна call stack, watch и остальные.

### 6.1 Папки ASP.NET

При [Создании соединения](#) все содержимое ASP.NET папок WEB-сайтов автоматически выкачивается Add-In'ом на локальный диск. Создаются папки App\_Code, App\_Data, App\_GlobalResources, App\_LocalResources, App\_WebReferences, App\_Browsers, если на WEB-сайте они содержат файлы. При необходимости обновить содержимое этих папок, выберите в контекстном меню сайта пункт «Update ASP.NET folders».

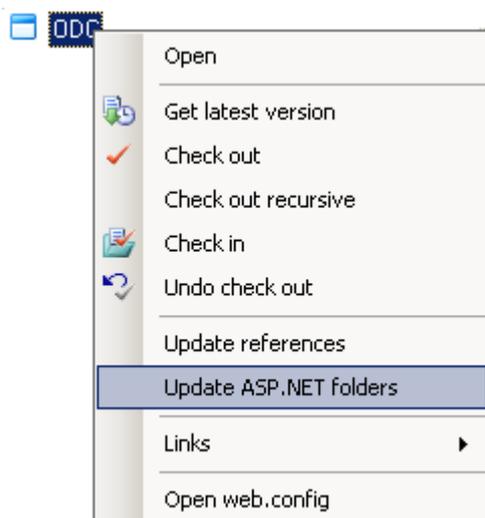


Figure 6-14

Наличие папки App\_Code так же необходимо для удачного выполнения локальной сборки WEB-сайта, если код форматов использует код, размещенный в этой папке.

### 6.2 Файл web.config

Во время компиляции и выполнения код страниц сайта обращается за данными к базе данных Quantum Publishing. Поэтому важно указать правильные параметры подключения к этой базе данных. Параметры подключение, как и прочие параметры сайта задаются в файле web.config. Открыть для редактирования в Visual Studio это файл можно либо указав его в окне Solution Explorer, либо выбрав пункт "Open web.config" в контекстном меню узла сайта в дереве Add-in.

## Локальная компиляция и отладка сайта

### Установка параметров для локального запуска сайта

---

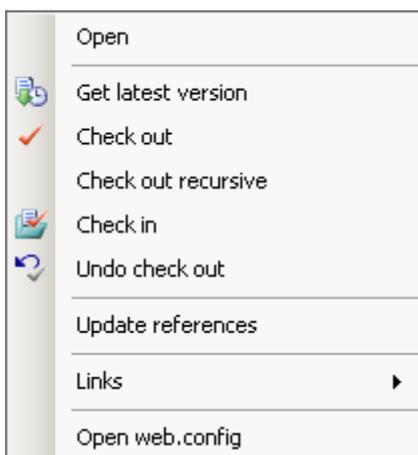


Рисунок 6-24

Нужно заполнить параметры соединения в ключах `appSettings/ConnectionString` и `connectionStrings/qp_database`.

Так же в секции `appSettings` при необходимости нужно заполнить ключи `MailHost`, `MainLogin`, `MailPassword`, `RelNotifyUrl`.

При первой сборке в `appSettings` помещается ключ `InternalExpirationTime`. Он управляет кэшированием внутренних данных `quantumart.dll` и по умолчанию имеет значение 0, т.е. кэширование отключено. Не следует без необходимости менять это значение, т.к. это может сказаться на результатах сборки сайта.

### 6.3 Установка параметров для локального запуска сайта

На странице свойств сайта - элемента дерева Add-in, можно указать путь к upload-ресурсам, которые будут использоваться при локальной сборке и отладке. Для этого нужно задать значения параметрам группы "Local Debug Parameters".

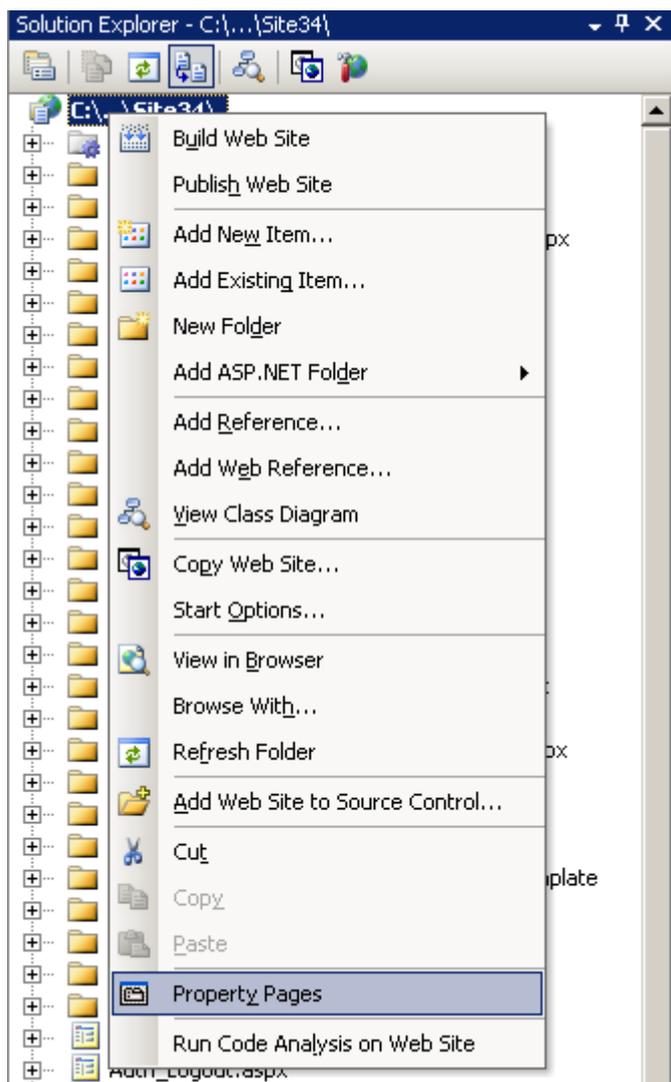


Рисунок 6-35

По роли они идентичны параметрам стандартным "Uploaded Files Location" в QP7 Backend.

Для того, чтобы указанные значения использовались страницами сайта нужно включить свойство "Use Local Upload Variables". Т.к. эти параметры хранятся не в базе данных QP7, а на локальном диске разработчика, для их изменения не требуется выполнять CheckOut сайта. Для использования этих значений регенерация исходного кода сайта так же не требуется.

## 6.4 Локальный запуск сайта

### 6.4.1 Запуск с использованием WEB-сервера Microsoft Visual Studio

Для запуска с использованием WEB-сервера Microsoft Visual Studio не требуется указывать ни каких дополнительных настроек. Все что необходимо - правильно указать параметры соединения к базе данных QPublishing (пункт 6.1) и нажать клавишу F5 (или выполнить команду Debug/Start Debugging). Будет выполнена компиляция всех страниц. В случае возникновения ошибок информация об этом будет выдана стандартными

средствами Microsoft Visual Studio - в окне Error List. В случае удачной компиляции будет открыто окно браузера со стартовой страницей сайта.

#### 6.4.2 Запуск с использованием внешнего WEB-сервера на примере IIS

Первым шагом следует создать для сайта виртуальную директорию. Для созданного из виртуальной директории сайта через оснастку управления IIS следует указать:

- Разрешения на чтение и выполнение скриптов.
- Включить использование .Net Framework версии 2 и выбрать соответствующий пул приложений.
- Включить для сайта Integrated Windows Authentication.

Далее в Microsoft Visual Studio изменить параметры сайта в окне Solution Explorer.

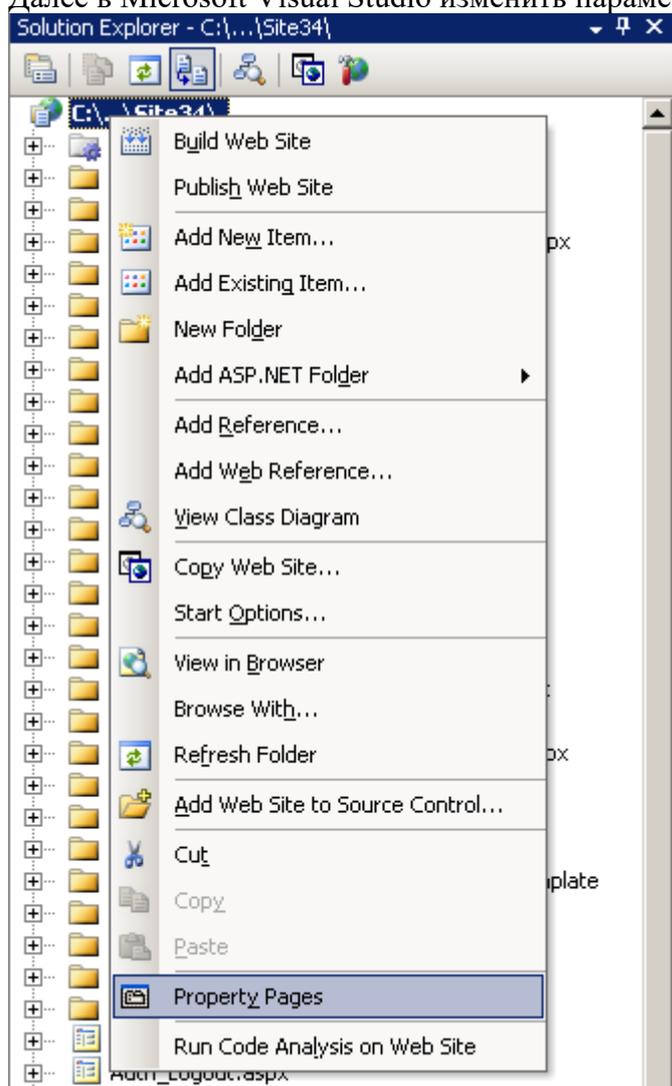


Рисунок 6-46

Укажите адрес WEB-сервера и имя стартовой страницы, как изображено на диалоге, подставляя свои значения.

## Приложение

### Структура хранения файлов на машине пользователя

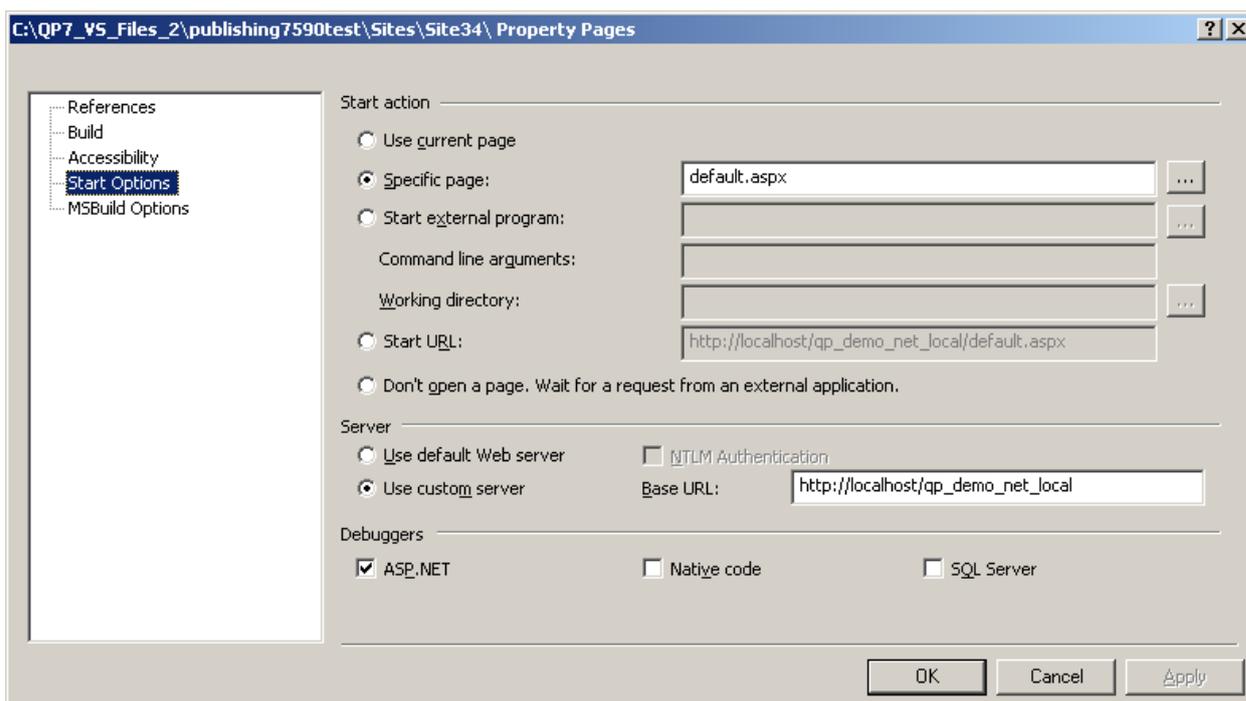


Рисунок 6-57

Т.к. адрес локального сайта помещается в исходный код страниц, после завершения ввода всех параметров нужно произвести регенерацию исходного кода всего сайта. Для этого для сайта выполняется рекурсивная операция "Get Latest Version". Эту операцию нужно выполнять любой раз, после изменения используемого WEB-сервера.

По завершении всех указанных действий можно нажать клавишу F5 (или выполнить команду Debug/Start Debugging). Будет выполнена компиляция всех страниц. В случае возникновения ошибок информация об этом будет выдана стандартными средствами Microsoft Visual Studio - в окне Error List. В случае удачной компиляции будет открыто окно браузера со стартовой страницей сайта.

## 7 Приложение

### 7.1 Структура хранения файлов на машине пользователя

Файлы всех соединений хранятся в указанной в настройках QP7.Backend Explorer папке. Файловая структура внутри этой папки выглядит следующим образом:

В корне папки лежат папки соединений с теми именами, которые пользователь вводит при создании соединения. Они отображаются в дереве Add-In. Также в корне папки лежит файл с именем RecentList.xml, в котором хранится информация для вывода подсказок при создании соединения.

В случае если при создании соединения был снят флаг "Save data offline", то эти данные хранятся только пока открыто соединение.

В каждой папке соединения хранятся следующие файлы:

- "имя соединения.sln" – файл solution Microsoft Visual Studio
- "имя соединения.xml", "имя соединения.xsd" – файлы с текущими данными для наполнения дерева

- “имя соединения-webServiceParams.xml” – файл с параметрами для доступа к веб-сервису

В этом файле хранятся в открытом виде login/password/customer code для доступа к веб-сервису, поэтому файлы соединений необходимо хранить в безопасном месте.

Дальнейшая файловая структура повторяет файловую структуру WEB-сайта. Это главное отличие от предыдущих версий QR7 Backend Explorer.

## **7.2 Восстановление информации**

В случае если в процессе работы с QR7 Backend Explorer произошла какая-либо ошибка, могут быть повреждены какие-либо файлы.

Если QR7 Backend Explorer не может загрузить соединение, а данные, сохраненные на локальной машине и не загруженные на сервер, терять не хочется, то рекомендуется выполнить следующие действия:

1. Выяснить, в каком именно месте происходит ошибка (какой файл её вызывает)
2. Переместить папку соединения в другое место на диске
3. Через Add-In удалить старое соединение
4. Создать новое соединение с тем же именем и параметрами
5. Выйти из Add-In
6. Заменить файлы созданного соединения старыми файлами (нужные, либо все, кроме тех, которые вызывают ошибку)
7. Запустить Add-In и попробовать продолжить работу.