

TechnologiCS 6 – разработка новой функциональности собственными силами

Покупая программное обеспечение, мы представляем себе определенный набор задач, который оно может решить. Но при этом прекрасно понимаем, что круг задач, возникающих при работе, например, с TechnologiCS, может оказаться столь широк, что его в полной мере не охватит и целый ряд решений, предлагаемых разработчиками.

Например, мы хотим ограничить интерфейс пользователя, убрав все лишнее и оставив только необходимое. Или, наоборот, добавить собственные команды и формы для расширения стандартных возможностей. Может возникнуть понятное желание автоматизировать ряд операций. Или использовать полезные свойства других продуктов – например, CAD-систем. Хотим мы одним нажатием клавиши публиковать в архиве сборочные единицы, спроектированные в CAD-системе, со всеми входящими в них деталями? Да так, чтобы все документы, созданные на основе деталей и сборок, были связаны между собой как в самой модели CAD-системы? А запускать требуемый режим TechnologiCS, работая в других приложениях? Конечно! В этой статье предлагается рассмотреть встроенный в TechnologiCS функционал, который как раз и позволяет исполнить наши пожелания.

Итак, какие общие задачи могут возникнуть?

- Ограничение/наращивание стандартного интерфейса системы.
- Автоматизация наиболее часто выполняющихся операций.
- Использование полезных свойств (функционала) других продуктов при работе в TechnologiCS.
- Использование функционала TechnologiCS в других продуктах.

Если в TechnologiCS требуется периодическое выполнение определенной последовательности действий, такую задачу можно автоматизировать с помощью скриптового модуля. Скриптовый модуль – это последовательность команд и функций, составленных на языке сценариев VBScript. Он выполнит практически те же действия, которые можно реализовать нажатием клавиш на клавиатуре или с помощью мыши. Для создания скриптового модуля применяется **Функционал разработки скриптовых модулей на языке VBScript** (рис. 1).

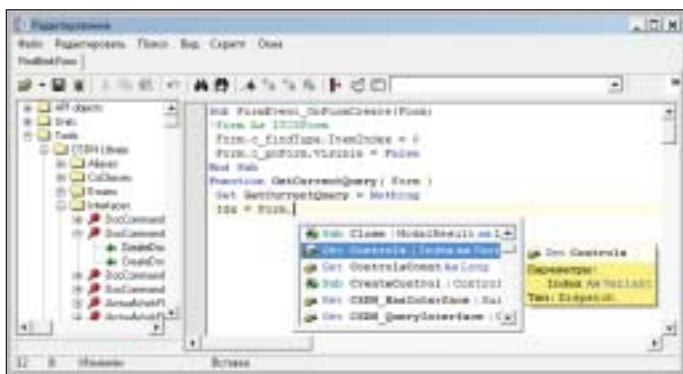


Рис. 1. Функционал разработки скриптовых модулей на языке VBScript

Чтобы пользователь не был ограничен стандартными формами и режимами TechnologiCS, предлагается использование **Дизайнера пользовательских форм** (рис. 2), который позволяет создавать собственные произвольные режимы с панелями инструментов, кнопками, выпадающими меню, элементами ActiveX, наборами данных, формами и иными объектами.



Рис. 2. Дизайнер пользовательских форм

Для упрощения работы с визуальными формами, скриптовыми модулями и отчетами разработан **Визуальный построитель запросов** (рис. 3), который может создавать собственные наборы данных и использовать их в различных режимах системы. Набор данных представляет собой запрос к объектам TechnologiCS, выполняемый так, как это было бы сделано с использованием структурированного языка SQL.

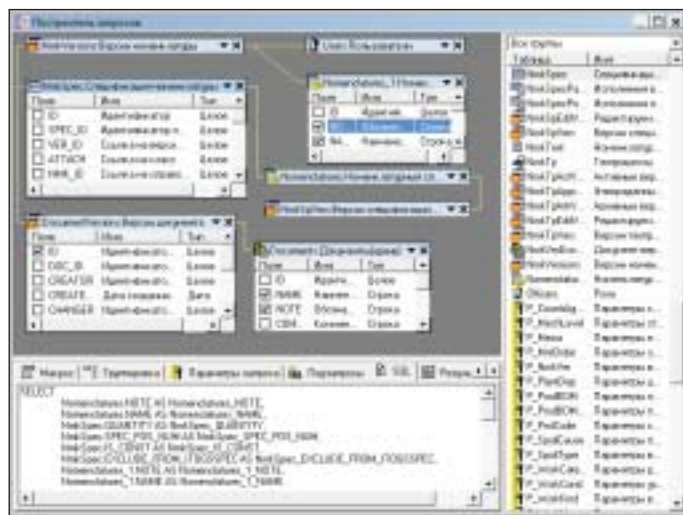


Рис. 3. Визуальный построитель запросов

Что касается интеграции TechnologiCS с другими системами, следует упомянуть о TechnologiCS API. Это интерфейсы, предоставляемые приложением для использования во внешних программных продуктах.

Далее необходимо решить вопрос о том, как преподнести конечному пользователю разработанные формы, модули и наборы данных. С такой задачей справляется **Дизайнер интерфейсов** (рис. 4). Он позволяет так модифицировать привычный интерфейс системы, что пользователь даже не заметит разницы между элементами стандартного интерфейса и элементами, разработанными самим пользователем. Например, можно подменять стандартные действия системы при вызове знакомого пользователю элемента меню или нажатии кнопки панели управления на запуск макроса или вообще убирать лишние элементы.

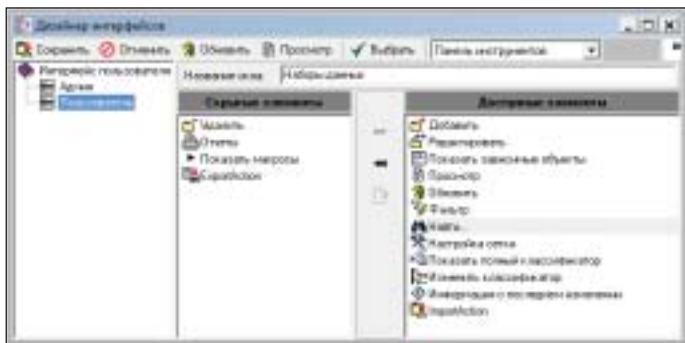


Рис. 4. Дизайнер интерфейсов

Рассмотрим перечисленный выше встроенный функционал TechnologiCS на таких примерах, как интеграция с NormaCS и расширенный поиск номенклатуры. Эти примеры представлены в демонстрационной базе данных дистрибутива TechnologiCS 6.

Допустим, нам требуется создать в архиве TechnologiCS документ NormaCS с автоматическим заполнением его атрибутов и сохранением связи с объектом NormaCS — чтобы его можно было открыть.

На стандартной панели TechnologiCS есть кнопка для вызова процедуры работы с NormaCS (рис. 5).

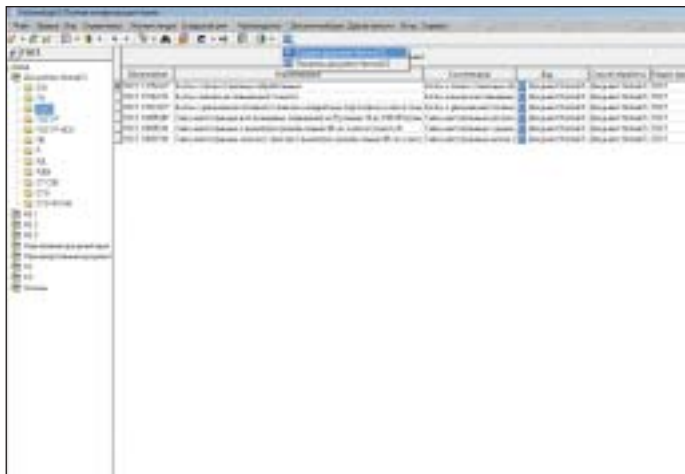


Рис. 5. Запуск процедуры создания документа NormaCS

Выбираем *Создать документ NormaCS*. Появляется диалоговое окно (рис. 6), созданное на основе пользовательской формы. Вводим, например, номер документа и нажимаем кнопку *Поиск*.

Открывается NormaCS с найденным по заданному критерию документом (рис. 7).

Выбираем в NormaCS команду *Копировать ссылку на документ* и закрываем систему.

Происходит перенос атрибутов документа NormaCS (наименование, обозначение, статус, дата утверждения и классификатор) в пользовательскую форму TechnologiCS (рис. 8).

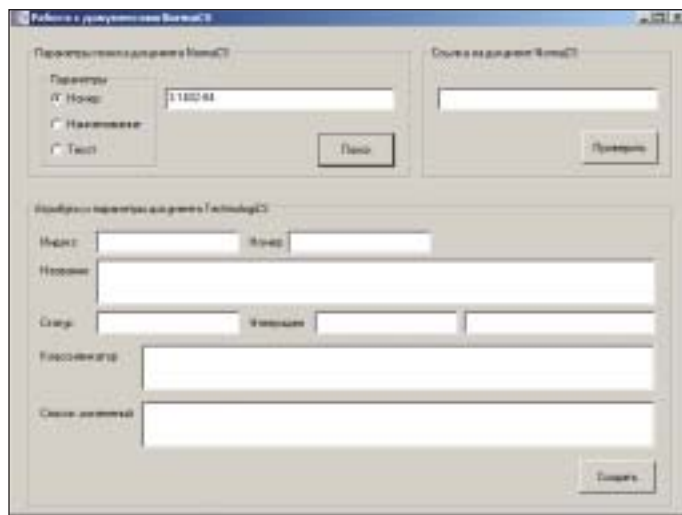


Рис. 6. Форма поиска документа NormaCS



Рис. 7. Результат работы процедуры. Открытие NormaCS

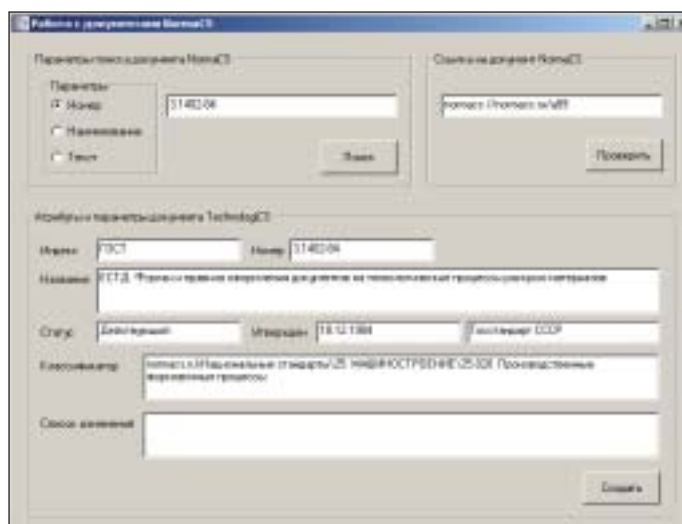


Рис. 8. Результат работы процедуры. Перенос атрибутов

Нажимаем кнопку *Создать*, выбираем рабочую группу для доступа к документу и получаем готовый документ в заданном классификаторе архива (рис. 9).

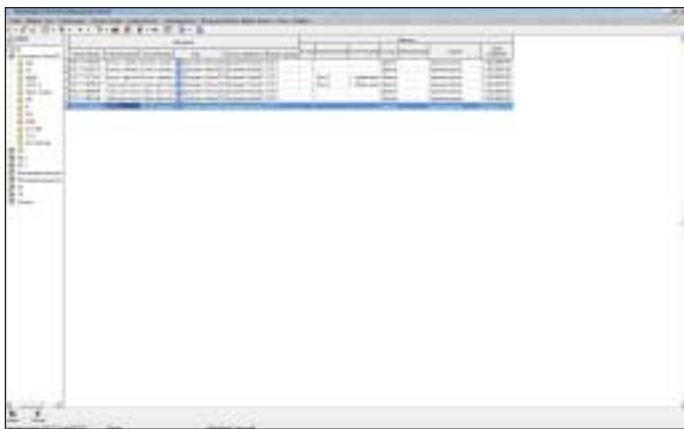


Рис. 9. Результат работы процедуры. Готовый документ

Для просмотра документа выбираем команду *Показать документ NormaCS* в меню панели TechnologiCS. Документ открывается (рис. 10).



Рис. 10. Результат работы процедуры. Открытие документа

Теперь последовательно рассмотрим инструментарий, при помощи которого это было реализовано.

Начнем с того, что данное расширение создано на основе скриптовых модулей. Содержание скриптового модуля определяется режимом системы. Эти модули содержат тексты кодов, использующихся в конкретных режимах. Работа с кодом ведется в редакторе скриптовых модулей, который представляет собой окно, разделенное на две части (рис. 11):

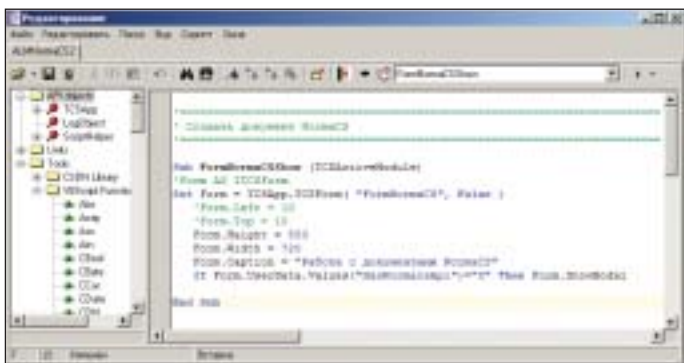


Рис. 11. Редактор скриптовых модулей

- дерево объектов (это API-объекты, присутствующие в контексте модуля; список всех модулей, созданных пользователями в системе; список стандартных функций VBScript);

- окно с текстом кода, в котором на нашем примере мы видим процедуры для запуска пользовательской формы.

Название нашей статьи звучит как "Разработка новой функциональности собственными силами". Но это не значит, что для создания расширений группе сопровождения придется отдавать все силы на их разработку. Не обязательно формировать и саму такую группу. Разработчики TechnologiCS постарались максимально облегчить решение подобных задач.

Начнем с того, что для удобства и значительного ускорения работы в редакторе скриптовых модулей предусмотрен функционал автодополнения (рис. 12).

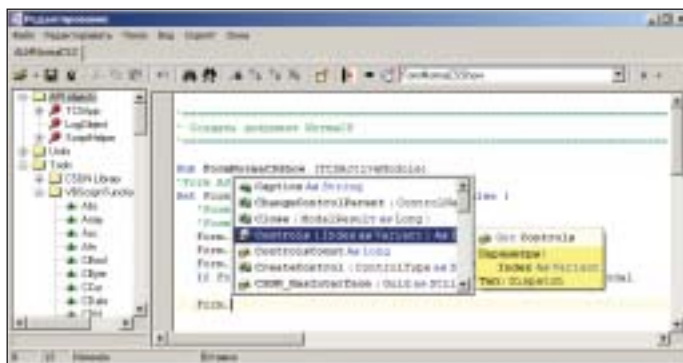


Рис. 12. Автодополнение кода

Печатаем текст, нажимаем сочетание клавиш "CTRL+Пробел" и получаем список доступных процедур, функций и переменных. В окне автодополнения при наведении курсора на объект доступно его дополнительное описание, которое может содержать параметры функций, возвращаемый тип переменной, комментарии. При этом вы также можете документировать собственные процедуры, функции и переменные, чтобы потом другой разработчик скриптового модуля мог получить по ним полное описание и использовать их в своем коде.

При вводе текста срабатывает фильтрация и по нажатию клавиши ENTER осуществляется вставка позиции в текст кода.

Кроме того, нет необходимости помнить свойства метода и их типы. С помощью "горячей" клавиши вы всегда можете вызвать подсказку и продолжить набирать текст (рис. 13).

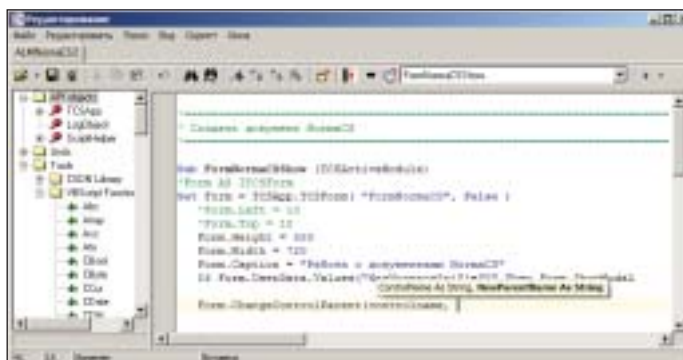


Рис. 13. Всплывающая подсказка

Если вас интересует дополнительная информация по конкретному объекту скриптового модуля, вы можете с помощью "горячей" клавиши автоматически позиционироваться в дереве объектов и вызвать справку с примерами по этому объекту (рис. 14) либо посмотреть список доступных свойств и методов для вашего объекта в дереве.

Таким образом, даже начинающему специалисту будет легко найти все необходимое для написания расширения.

Что делать, если человек слабо представляет себе, как организован API TechnologiCS? Или просто хочет глубже в нем разобраться? Или вообще не имеет большого опыта работы с языком VBScript?

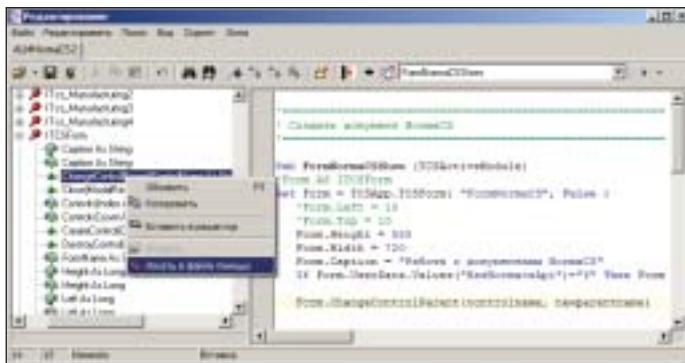


Рис. 14. Автоматическое позиционирование в дереве объектов

На этот случай разработчики предусмотрели такой интересный инструмент, как TCS API Explorer.

Вызвать обозреватель API можно непосредственно из редактора скриптовых модулей. По закладкам инструмента TCS API Explorer разнесены интерфейсы текущего режима системы (рис. 15). Например, закладка *IPModule* представляет собой одноименный интерфейс, содержащий свойства объекта, на котором в данный момент установлен курсор. В нашем случае мы запустили редактор из режима *Архив*, поэтому объектом исследования является документ архива.



Рис. 15. TCS API Explorer

Слева на закладке представлен перечень доступных свойств объекта, справа — их значения и прочие реквизиты. Вверху — наиболее часто используемые команды при работе со свойствами объекта. Выполнить команду над объектом можно непосредственно из TCS API Explorer.

На следующей закладке показан интерфейс всех возможных команд и действий над объектом режима системы (рис. 16), а также команды при работе с набором данных (сеткой), к которому он относится. Слева — перечень команд, справа — расшфровка, аналогично предыдущей закладке. Их также можно запустить и посмотреть, как они работают.

Мы рассмотрели свойства объекта, возможные действия над ним, далее выше по иерархии мы можем перейти к самому набору данных (сетке), в котором находится наш документ (рис. 17).

После того как мы просматривали и редактировали свойства, запускали команды, перемещались по набору данных (сетке), можно перейти на закладку *Log* и увидеть готовый текст кода согласно всем нашим произведенным в TCS API Explorer действиям над объектом (рис. 18). Например, мы редактировали свойства документа, сохраняли изменения, запускали карточку документа на редактирование. Теперь можно скопировать этот текст, вставить вместо символов "@" свои имена переменных и получить готовый текст кода для редактирования

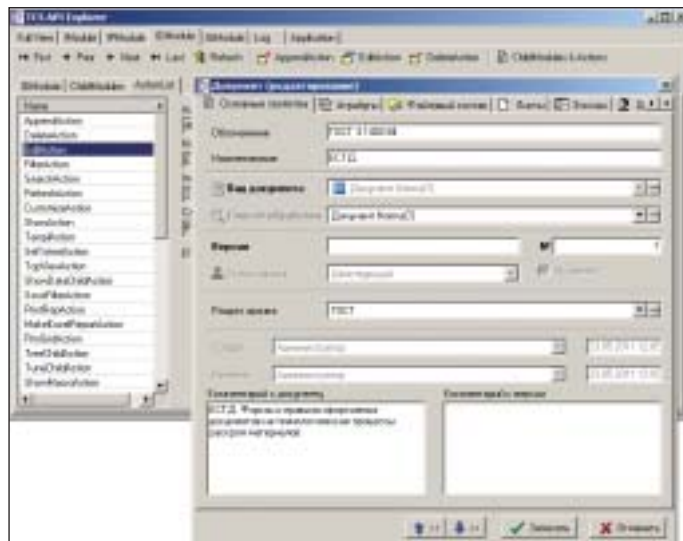


Рис. 16. TCS API Explorer. IDModule

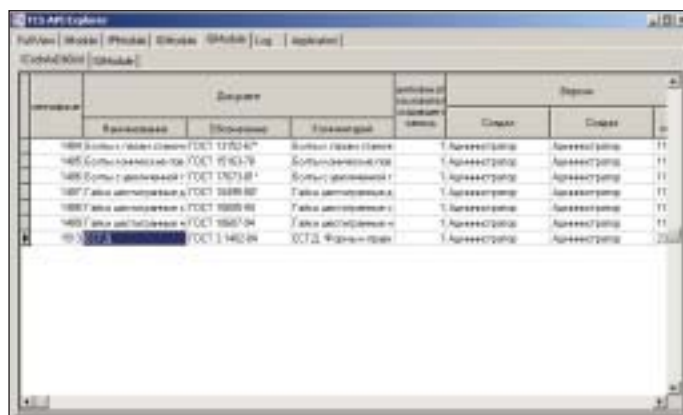


Рис. 17. TCS API Explorer. IGModule

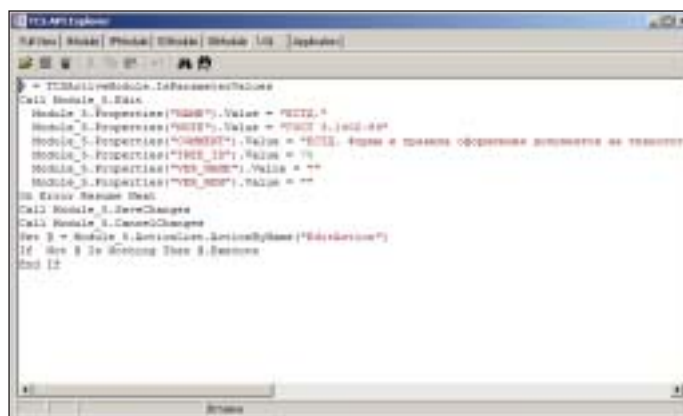


Рис. 18. TCS API Explorer. Log

свойств документа. Аналогичным образом мы можем составлять тексты кодов, находясь в любом режиме системы и не покидая при этом TCS API Explorer.

Итак, TCS API Explorer является инструментом не только визуализации объектов TechnologiCS API, но и создания скриптовых модулей. Кроме того, он может служить средством самообучения начинающего специалиста языку VBScript в рамках продукта TechnologiCS.

Мы рассмотрели инструмент, позволяющий быстро составить код. Перейдем к созданию пользовательской формы. Для этого необходимо запустить редактор форм, предназначенный для создания и редактирования пользовательских

форм ввода и отображения информации. Редактор форм представляет собой сочетание редактора кода и дизайнера форм. Переход к дизайнеру форм осуществляется непосредственно из редактора скриптового модуля. При запуске дизайнера форм перед нами появится окно с элементами управления и рабочим пространством (рис. 19). Панели управления представляют собой стандартный набор команд. Это свойства элементов, свойства формы, перемещение элементов, функция предварительного просмотра, дерево объектов и визуальные элементы формы. Визуальные элементы включают в себя надписи, переключатели, панели, кнопки, элементы ActiveX, наборы данных, меню, закладки и прочее (рис. 20). В общем, все что требуется для проектирования достаточно функциональных и сложных форм.

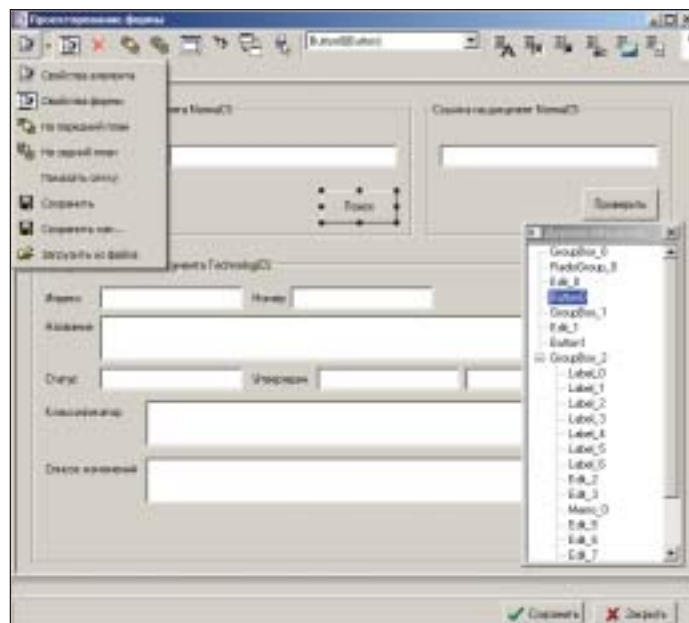


Рис. 19. Дизайнер форм. Панель управления

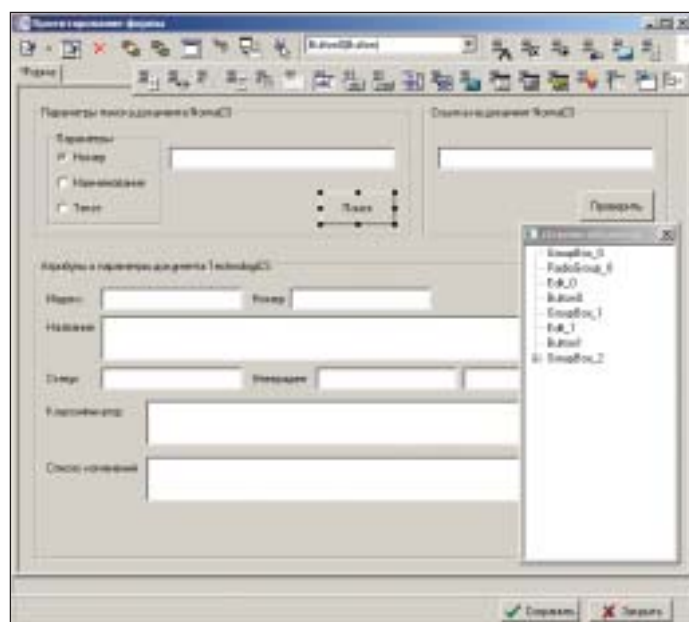


Рис. 20. Дизайнер форм. Визуальные элементы формы

Из таких элементов собран наш простой пример интеграции с NormaCS.

Вследствие появления элементов формы, в редакторе кода появляются дополнительные элементы управления. К ним относятся выпадающее меню со списком возможных событий

объекта (*События по объекту, Общие события, События формы*) и выпадающее меню *Объект* с перечнем элементов формы (рис. 21).

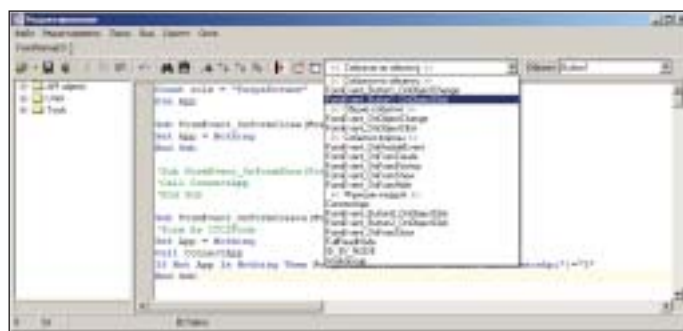


Рис. 21. Редактор кода. Меню выбора события

Таким образом, текст кода в редакторе форм делится на блоки из описаний событий по нужному элементу формы.

Мы разобрали, как получить текст кода скриптового модуля, содержащий необходимые процедуры по соединению с NormaCS, и рассмотрели пользовательскую форму, спроектированную в дизайнера форм. Осталось только предложить пользователю удобный вариант запуска нашего расширения. Для этого был разработан дизайнер интерфейсов.

Для запуска дизайнера интерфейсов необходимо перейти в интересующий нас режим или окно системы (в нашем случае это *Архив*) и нажать сочетание клавиш CTRL+ALT+F5. Появится окно, показанное на рис. 22.



Рис. 22. Дизайнер интерфейсов

Слева в дизайнера отображается классификатор интерфейсов — с интерфейсами, которые доступны пользователю для редактирования. При открытии дизайнера курсор позиционируется на интерфейсе, используемом в данный момент. Интерфейсы собраны в отдельном "Справочнике интерфейсов".

Сверху перечислены ряд команд и категории интерфейса, а ниже рабочее пространство разделено на две части: "Скрытые элементы" и "Доступные элементы" категории интерфейса. Между ними стрелки, с помощью которых осуществляется перенос элементов.

Категории предназначены для выбора настраиваемых элементов интерфейса. В нашем случае это команды TechnologiCS, доступные в режиме *Архив*, контекстное меню, панель инструментов, пользовательские макросы (служат для добавления на панель управления кнопок, выполняющих вызов пользовательской процедуры), наборы данных и пользовательские команды.

В нашем примере кнопки для запуска скриптового модуля решено было вынести на панель управления. Мы добавили их в "Доступные элементы" (рис. 23).

Что получилось в итоге? Стояла задача расширить функционал TechnologiCS, а именно позволить пользователю создавать в

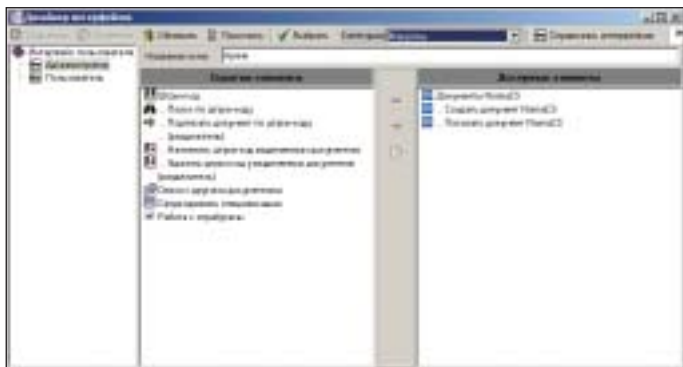


Рис. 23. Дизайнер интерфейсов. Добавление кнопок для запуска процедур

архиве и просматривать документы NormaCS, используя стандартные панели TechnologiCS для вызова процедур. Мы составили текст кода, редактор которого по мере своего развития все ощутимее снижает трудоемкость и увеличивает скорость написания кода, делает более понятным API TechnologiCS и вносит элемент автоматизации в процесс разработки интерфейса пользователя. Далее, в дизайнера, мы составили форму для представления информации пользователя в требуемом виде. Для удобства пользователя вынесли на стандартную панель меню запуска нашего расширения.

Но иногда возникают более сложные задачи, когда недостаточно перейти в нужный нам режим или открыть окно с необходимой информацией для выполнения ряда команд. Поскольку такого режима с необходимым набором данных просто нет. Что делать в этих случаях? Раньше можно было самому формировать наборы данных, используя возможности скриптовых модулей. Теперь в этом нам поможет Визуальный построитель запросов. Его работу мы рассмотрим на примере функции "Расширенный поиск номенклатуры".

Допустим, требуется расширить стандартные средства для поиска номенклатуры. Из стандартного выпадающего меню запускается скриптовый модуль "Расширенный поиск номенклатуры". Появляется окно, созданное в дизайнера форм, — так же, как в нашем примере интеграции с NormaCS (рис. 24). В окне мы видим элементы для задания критериев поиска. Это поиск по обозначению, наименованию, коду, поиск в определенных классах и панель, где будут отображаться результаты поиска. Выберем *Искать по наименованию* и найдем все корпуса в классе "Детали".

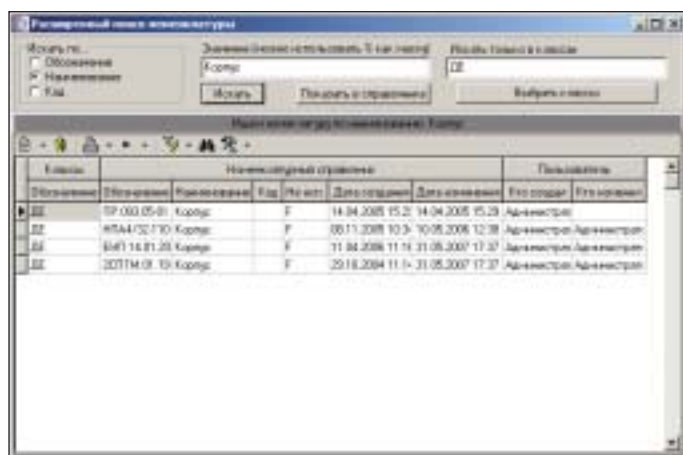


Рис. 24. Расширенный поиск номенклатуры

Обратим внимание, что в результатах не появился набор данных номенклатурного справочника. Результат представляет собой сложение отдельных значений справочника классов, отдельных полей режима номенклатуры и отдельных значений справочника пользователей.

Как это было получено? Теперь значительно проще, чем в предыдущих версиях.

Для дальнейшего расширения функциональности системы TechnologiCS можно создавать собственные наборы данных и использовать их в визуальных формах, скриптовых модулях, отчетах, интерфейсах и т.д. Такой набор представляет собой запрос к объектам TechnologiCS, выполняемый так, как это было бы сделано с использованием структурированного языка SQL. Сами наборы данных формируются в Визуальном построителе запросов (рис. 25).

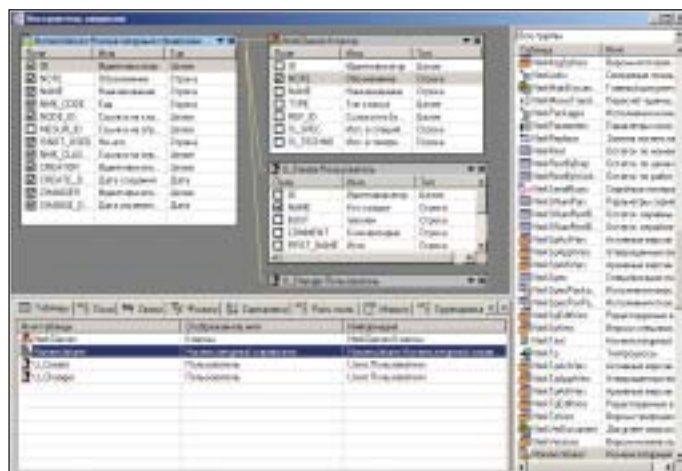


Рис. 25. Визуальный построитель запросов

Окно разделено на три части:

- рабочая область, где отображены выбранные нами таблицы и связи, наложенные между таблицами;
- область таблиц, где расположен список таблиц, доступных для выбора;
- область редактирования, позволяющая изменять набор данных с помощью фильтров параметров, подзапросов, расчетов, скриптовых модулей и т.д.

Пройдем основные этапы создания нашего запроса. Выбираем в списке таблиц группу "Номенклатура". Перетаскиваем таблицу "Классы" в рабочую область (рис. 26).

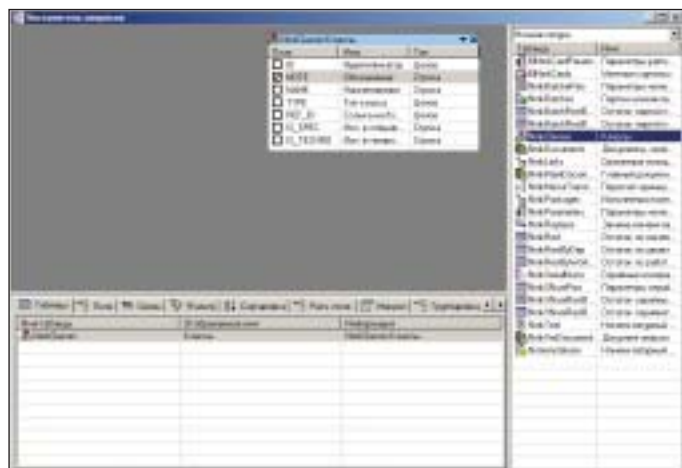


Рис. 26. Визуальный построитель запросов. Добавление таблицы

Таблица представляет собой перечень полей (их идентификаторы), их имена и типы. Флажками можно указывать, какие именно поля отображать либо не отображать в запросе. Убираем лишние. Выбираем из контекстного меню таблицы, связанные с таблицей классов. В нашем случае это "Номенклатура" (рис. 27).

Таблица с номенклатурой автоматически добавляется в рабочую область – со связью по умолчанию. Аналогично добавляем таблицы с пользователями (рис. 28).

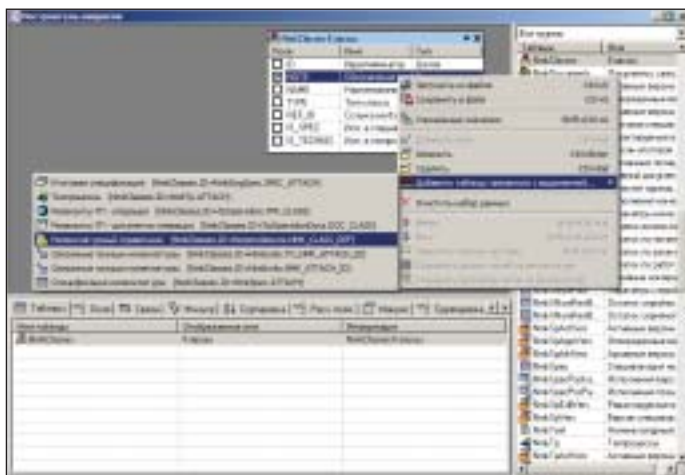


Рис. 27. Визуальный построитель запросов. Добавление связанной таблицы

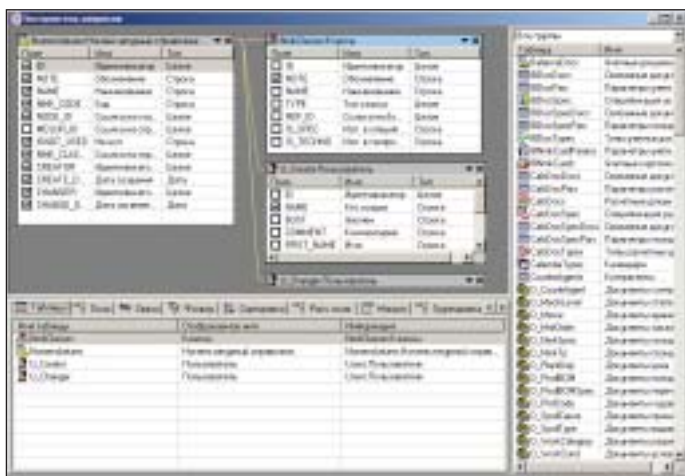


Рис. 28. Визуальный построитель запросов. Таблицы со связями

Внизу, в области редактирования, на закладке **Таблицы** отображается перечень наших таблиц. На другой закладке – перечень полей, участвующих в наборе данных (рис. 29). На следующей настраиваются связи и условия связей таблиц (рис. 30).

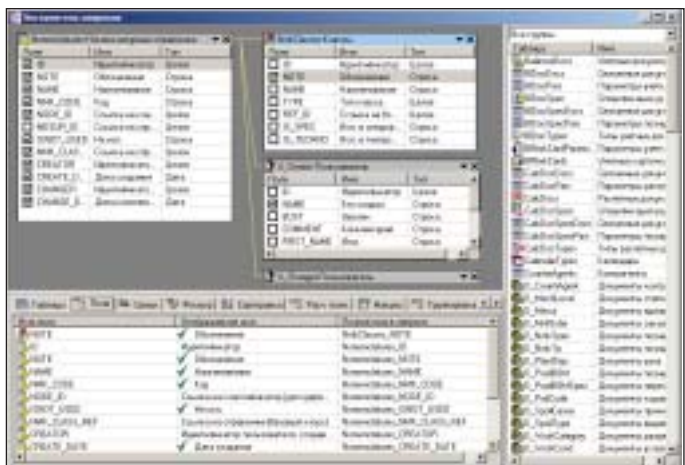


Рис. 29. Визуальный построитель запросов. Перечень полей таблиц

Далее устанавливаются фильтры. В данном случае нам необходимо фильтровать значения набора данных и зависимости от введенного слова в поле поиска на пользовательской форме.

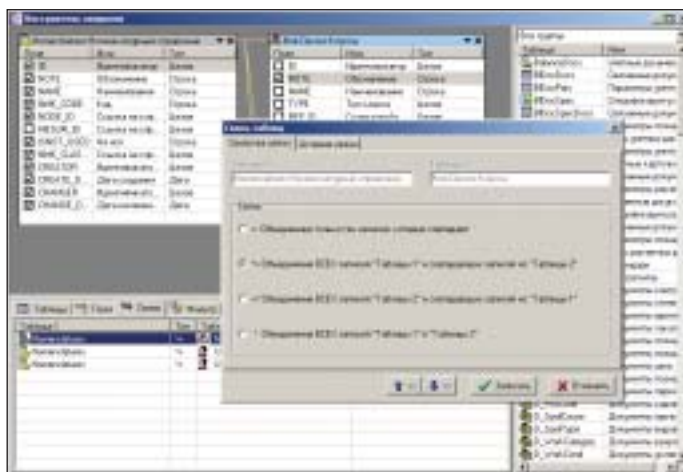


Рис. 30. Визуальный построитель запросов. Выбор типа связи между таблицами

Поэтому в качестве условия фильтрации зададим параметр, который будет передаваться из формы (ключевое слово поиска) в набор данных и фильтровать значения (рис. 31).

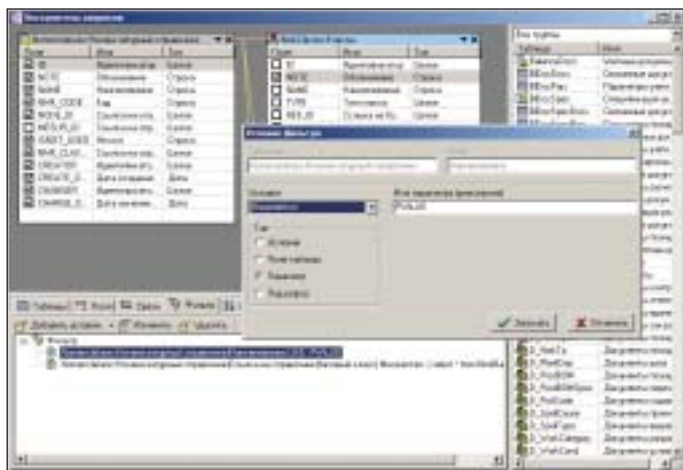


Рис. 31. Визуальный построитель запросов. Настройка фильтров

Результат наших настроек для наглядности отображается на закладке **SQL** в виде кода (рис. 32), а как будет выглядеть набор

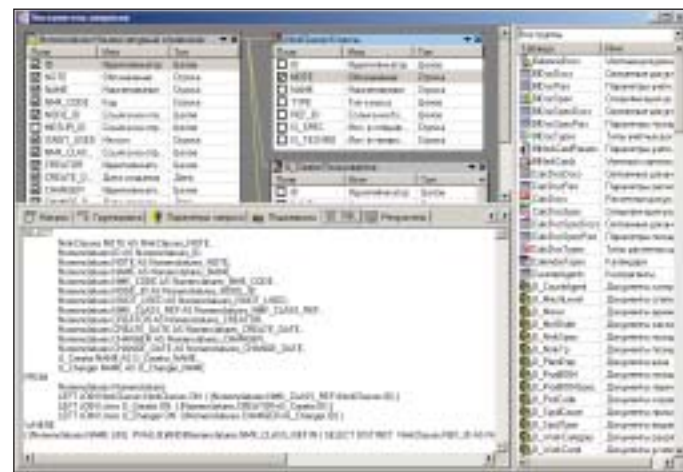


Рис. 32. Визуальный построитель запросов. Текст SQL-запроса

данных на форме показывает следующая закладка – **Результаты** (рис. 33). Результатом является таблица (набор данных), в которой мы выводим класс искомой номенклатуры, информацию о номенклатуре и создавших/редактировавших ее пользователях.

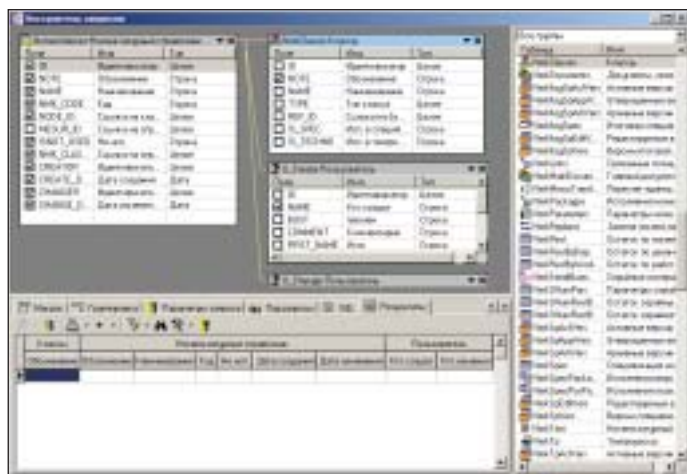


Рис. 33. Визуальный построитель запросов. Результат выполнения запроса

Помимо фильтрации и наложения различных связей в Визуальном построителе запросов можно использовать сортировку, расчетные поля с различными параметрами и агрегирующими функциями, скриптовые модули на VBScript, группировки данных и подзапросы. То есть необходимый набор инструментов, который может понадобиться при создании наборов данных.

Полученный набор данных вставляется в пользовательскую форму и, как результат, мы получаем еще одно расширение функционала TechnologiCS. Где форма создана на вкус и цвет конечного пользователя и запускается из удобного для него раздела меню. Ключевое слово поиска передается параметром в набор данных, а этот набор с установленным фильтром отображается на форме (рис. 34).

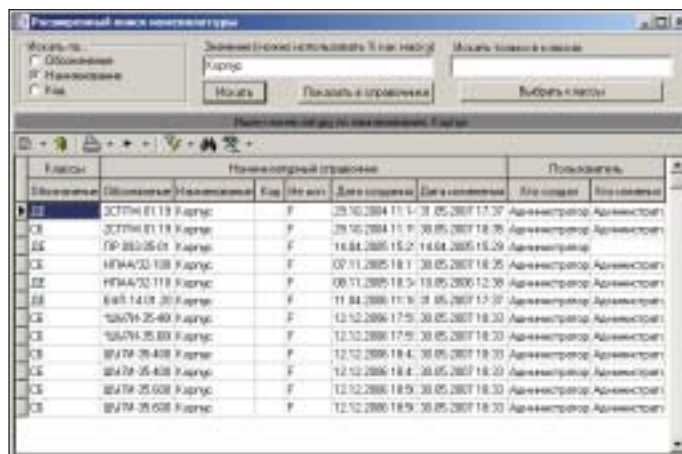


Рис. 34. Результат работы расширения

Целью всего сказанного было показать, что если круг задач, возникающих при работе с системой, оказался шире того, что обеспечен целым рядом решений, предложенных разработчиками программного обеспечения, может возникнуть необходимость расширить стандартный функционал. Но это не означает, что ради облегчения работы пользователя группа внедрения вынуждена будет тратить много сил и средств на разработку расширений. Разработчики TechnologiCS постарались сделать это раньше и упростить работу пользователей, обеспечив их всеми необходимыми инструментами.

Евгений Слинкин
CSoft Development Новосибирск
Тел.: (383) 346-0633
E-mail: e.slinkin@nsk.csoft.ru

БИБЛИОТЕКА НОРМАТИВОВ

- Реквизиты и тексты свыше 65 тысяч нормативных документов (ГОСТ, СНИП, СанПиН, РД, технологические карты и др.)
- Более 6 тысяч серий и типовых проектов
- Актуальность информации
- Удобство работы, легко настраиваемый интерфейс
- Интеграция с MS Office, конструкторскими программами
- Интеллектуальный поиск по растру с подсветкой найденного фрагмента
- Цитирование документа
- Расстановка гиперссылок
- Поддержка формата DWG
- Новый раздел "Разработчики ТУ"

