

А вместо ГИСа – пламенный мотор...

Регулярные пресс-релизы группы компаний CSoft о победах в тендерах и успешно завершённых проектах по геоинформационной тематике избавляют нас от необходимости вновь и вновь повторять уже ставшие привычными, а для многих — и принятыми, тезисы о необходимости использования единого хранилища для пространственных и семантических данных на основе СУБД Oracle, о преимуществах инструментальной ГИС CS MapDrive, о развитости функционала ориентированных на конкретную предметную область семейств программных приложений (UrbaniCS — для градостроительного кадастра, UtilityGuide — для задач мониторинга и управления инженерными коммуникациями, EcologiCS — для задач экологического мониторинга).

Вместе с тем опыт внедрения в разных регионах и разных предметных областях выявил одну очевидную истину: "нельзя объять необъятное". То есть сколь бы ни были развиты структуры данных, сколь бы ни был изощрен функционал, всегда найдется что-то глубоко специфичное для данного клиента, не учтенное при проектировании, но, конечно же, жизненно необходимое для него, клиента...

Два примера.

Первый. В процессе внедрения ИСОГД в одном из регионов возникла специфическая задача учета выдачи разрешений на геодезические изыскания. При этом пространственным объектом, подлежащим учету, должны быть нарисованные "от руки" нетопологические контуры на единой топологической основе ИСОГД; в семантику требуется включить дополнительные справочники по лицензированным геодезическим фирмам; конечно, нужны отче-

ты; естественно, необходим и ретроспективный анализ... Задачу следует решать, однако вряд ли описанный подход можно тиражировать, это, скорее, дань традициям и сложившейся организационной структуре.

Приведенный пример иллюстрирует необходимость решения частной задачи в частной же постановке.

Другой пример. При паспортизации сетей водоснабжения и канализации города Душанбе с применением WaterGuide выяснилось, что в состав объектов, находящихся в ведении МУП "Душанбеводоканал", входят... айвовые деревья, которые обязательно нужно учитывать.

Здесь наглядно иллюстрируется возможность появления в единой базе данных специфических семантических, факультативных полей.

Что же делать? Нужно ли перегружать универсальные и тиражируемые программные средства уникальными узкоспециальными задачами, которые почти гарантированно не пригодятся в других регионах?.. Или же "плодить" многочисленные версии этих программных средств под города N-ск и M-ск, обрекая себя на медленную смерть в процессе их технической поддержки?.. Или на все запросы клиента говорить: "Вам это не нужно, мы точно знаем"?.. Вряд ли хоть один из перечисленных подходов приведет к желаемому результату. Но поставленная проблема требует решения... И такое решение существует!

Чтобы сотрудники заказчика сами могли быстро создавать дополнительные пользовательские приложения, вносить дополнительные семантические поля и справочники, не вторгаясь в разработанную и отлаженную структуру данных, предложена концепция, которую мы на-

зываем "движком", а официально — **CS GIS Engine**.

Этот программный продукт ориентирован прежде всего на две основные категории пользователей:

- коллективы независимых разработчиков, которые обладают определенным know-how (например, в сфере расчета показателей расхода энергоресурсов), но не имеют возможности интеграции своих знаний в существующие комплексные ГИС-решения от CSoft;
- организации, использующие один или несколько программных продуктов семейства UtilityGuide и UrbaniCS, но нуждающиеся в дополнительной функциональности, выходящей за рамки требований федерального или местного законодательства.

CS GIS Engine — почти готовое приложение, поскольку в нем по умолчанию реализуется большинство необходимых пользователю функций, в том числе работа с картой и отчетами. Таким образом, работать со своими данными заказчик может без необходимости предварительной настройки или программирования.

По понятным и уже обсуждавшимся причинам предполагается, что разрабатываемые на основе CS GIS Engine приложения ориентированы на применение единого хранилища для пространственных и семантических данных на основе мощной СУБД Oracle. Активно используются и другие основные преимущества СУБД Oracle, такие как система безопасности на основе пользователей, ролей и прав, а также возможность ретроспективного анализа данных на базе Oracle Workspace Manager.

Одним из основных понятий CS GIS Engine является конфигурация, включа-

ющая в себя информацию о соединении с базой данных, настройки карты, пользовательского интерфейса, а также образы форм и сценариев. Причем для каждого пользователя могут быть указаны свои варианты настроек пользовательского интерфейса, образы форм и сценарии. Очень важно, что в CS GIS Engine вся информация о конфигурации хранится в таблицах базы данных, к которой осуществляется подключение. Это позволяет пользователям работать в знакомом окружении независимо от рабочего места, с которого осуществлен вход в систему. Кроме того, значительно упрощается процесс развертывания системы и обновления ее конфигурации, поскольку при подключении к базе данных пользователь лишь выбирает требуемую конфигурацию, после чего для него автоматически загружаются соответствующие настройки, формы и сценарии.

К этому моменту заинтересованный читатель обязательно должен спросить: "А как же, собственно, выполняется настройка конфигурации CS GIS Engine?" Чтобы ответить на этот вопрос, рассмотрим процесс реализации упомянутой выше небольшой задачи учета выдачи разрешений на геодезические изыскания. Естественно, вся необходимая базовая картографическая информация, такая как границы округов, гидрография и строения, уже введена в ИСОГД и своевременно обновляется.

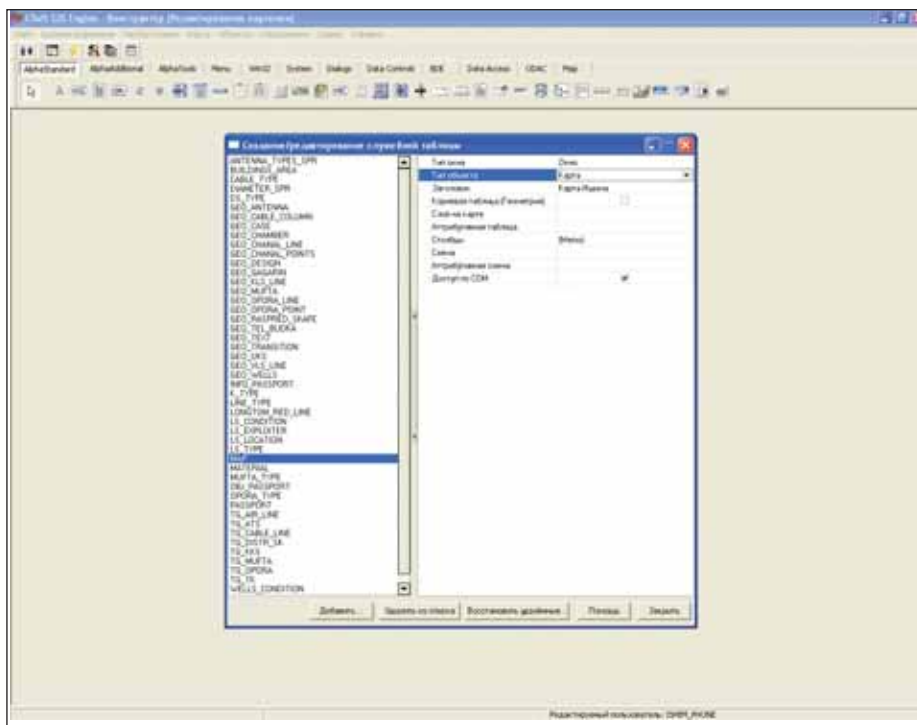
Вначале CS GIS Engine запускается в режиме редактора конфигурации. После ввода имени пользователя, пароля и службы Oracle создается или выбирается *Профиль конфигурации* — в нашем случае, например, *Геодезические изыскания*.

Для хранения конфигурации CS GIS Engine использует несколько таблиц основной схемы базы данных, которые создаются автоматически.

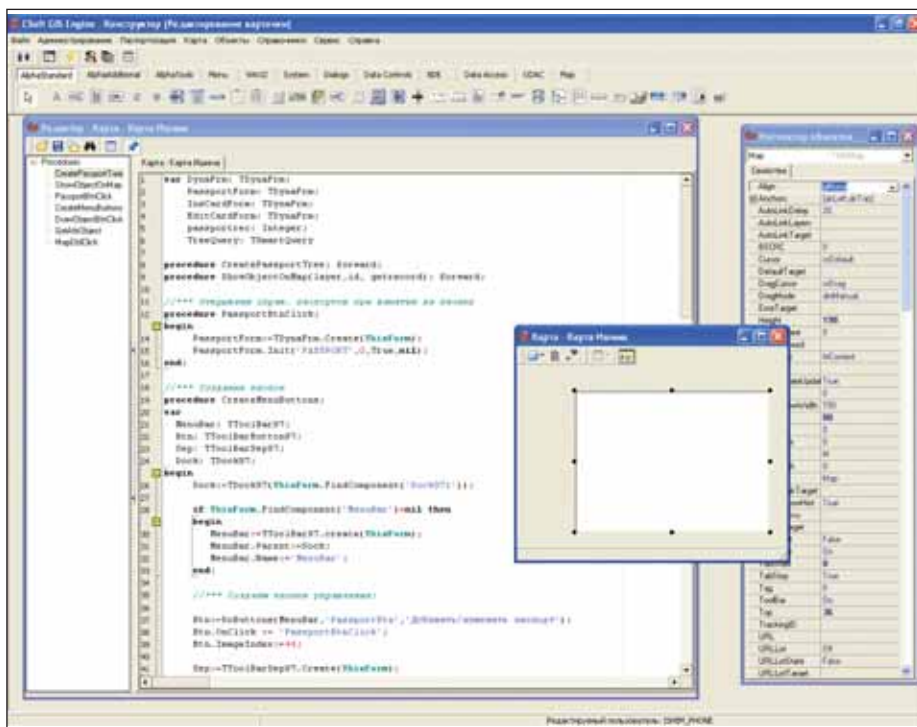
Теперь самое время приступить к изучению средств настройки конфигурации CS GIS Engine.

В приложениях ГИС работа традиционно строится вокруг одного или нескольких окон карты. GIS Engine полностью поддерживает этот подход, но не ограничивается им, позволяя при необходимости создавать конфигурации, которые вообще не используют окна карты.

Для каждого класса объектов, называемого также слоем, могут автоматически по умолчанию создаваться формы двух типов — форма карточки объекта и форма списка объектов. Первый тип формы служит для отображения информации одного объекта, а второй — для отображения нескольких объектов в списке. Однако автоматически генерируемые формы могут устроить пользователей только на первых порах. А как же



Создание объекта для окна карты



Форма карты в редакторе кода

изменить вид форм по умолчанию или, что еще лучше, создать собственные формы? Конечно же, с помощью конструктора в визуальном режиме! Впрочем, визуальным режимом создания форм сейчас уже никого не удивишь, но вот приятной неожиданностью для разработчика станет обилие компонентов, доступных пользователю для создания собственных форм: здесь присутствуют практически все стандартные компоненты, применяемые пользователями

сред разработки Borland. Да и встроенный язык программирования Pascal Script в значительной степени напоминает своего "именистого" предка.

Опробуем среду разработки в деле. Открываем редактор метаданных и создаем новый объект, который будет соответствовать новой форме с изображением карты.

Не забываем указать, что наша форма будет доступна в меню *Карта*. Именно из него выбираем нашу форму, пере-

носим на нее компонент карты и переключаемся в окно редактора кода, где с помощью редактора свойств устанавливаем нужные нам свойства.

Теперь перейдем к созданию формы разрешения на проведение геодезических изысканий. Снова открываем редактор метаданных и создаем новый объект для нашей формы. Но на этот раз сделаем так, чтобы форма объекта открывалась не из меню, а двойным щелчком мыши на границе участка, выделенного для проведения геодезических изысканий.

Переключаемся в окно редактора кода и размещаем на форме необходимые компоненты: поле ввода для регистрационного номера, названия организации, а также двух дат, ограничивающих срок действия разрешения. Причем для полей ввода номера документа и названия организации воспользуемся компонентами, связанными с источником данных (data-aware controls). Поэтому добавим на форму также компоненты запроса (query) и источника данных (data source) — знакомые операции для читателей, имеющих опыт работы с Borland Delphi. После размещения на форме всех компонентов по очереди установим их свойства с помощью редактора свойств и при необходимости добавим процедуры для обработки нужных событий формы.

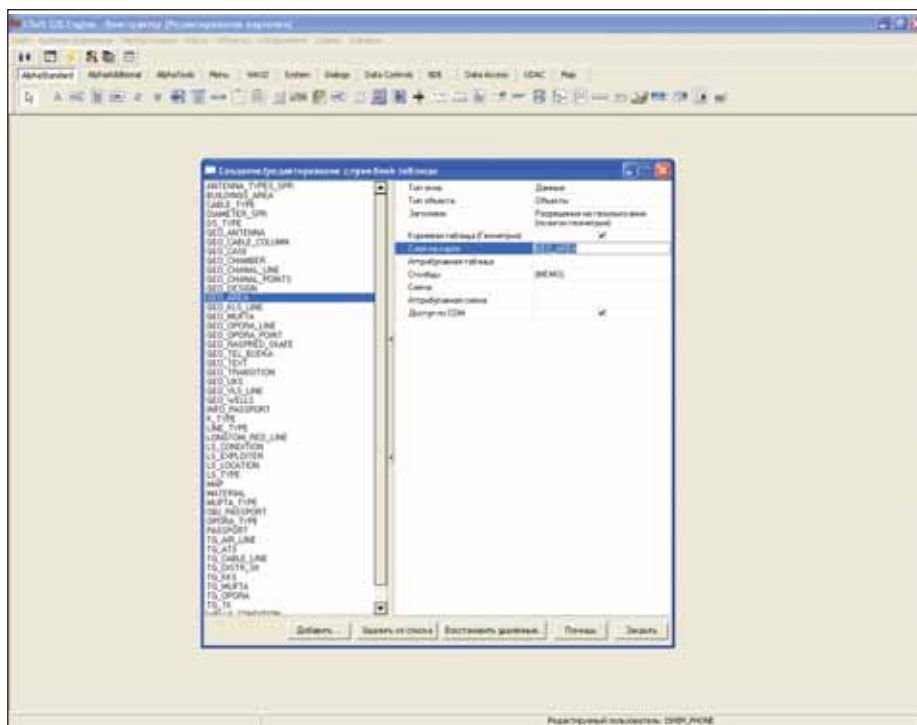
Чтобы форма участка открывалась двойным щелчком мыши на его границе в окне карты, снова откроем форму карты в редакторе кода и напишем свою процедуру обработки события двойного щелчка мыши. Ничего сложного!

Теперь займемся отчетом. С каждой формой CS GIS Engine по умолчанию связывается отчет по умолчанию. Воспользуемся этой возможностью для подготовки печатного образа документа разрешения. Как обычно, создадим соединение с базой данных и поместим на форму отчета нужные поля.

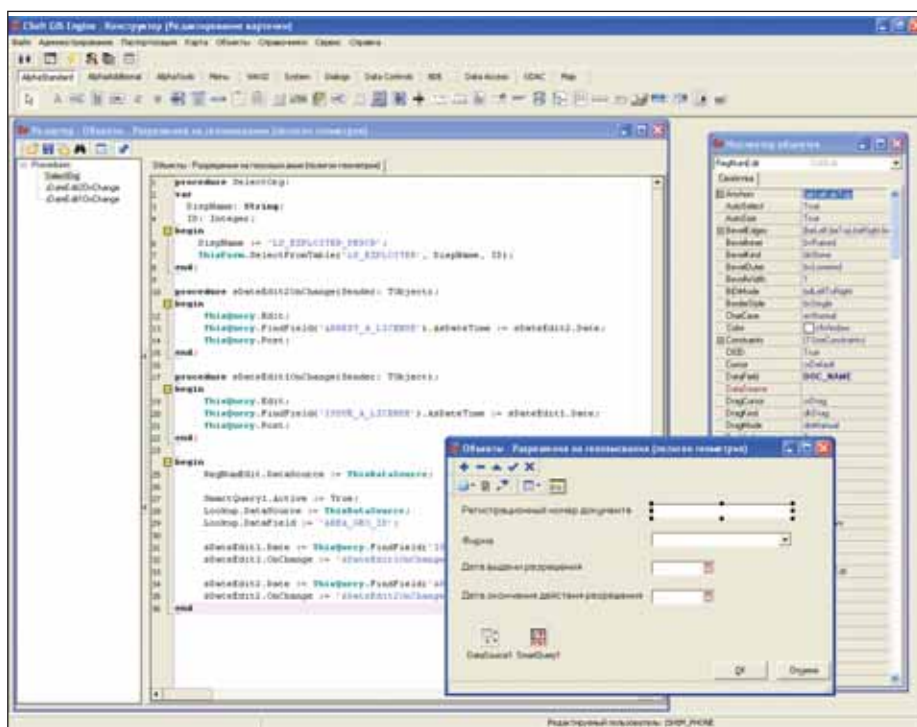
Нажав кнопку *Сохранить*, мы получим простую конфигурацию для работы с участками для геодезических изысканий.

Остается лишь опробовать созданную конфигурацию в работе. Для этого достаточно выбрать пункт меню *Администрирование* → *Запустить* — и приложение CS GIS Engine будет запущено в режиме выполнения конфигурации. Как и раньше, вводим имя пользователя и пароль, выбираем службу Oracle. В списке профилей указываем нашу конфигурацию и нажимаем кнопку *OK*.

Нажимаем кнопку отчета — и через некоторое время на экране появится окно предварительного просмотра, содержащее печатный образ необходимого нам документа.



Создание объекта формы участка



Форма участка в редакторе кода

Предложенный вариант настройки конфигурации предусматривает использование визуального конструктора совместно с языком сценариев и является основным механизмом расширения функциональности. Два других варианта расширения предполагают применение механизма COM для разработки собственных модулей (plug-in) и подключение внешних динамически загружаемых модулей (DLL). Кроме того, собственные формы конфигурации CS GIS

Engine также допускают управление с помощью механизма COM.

Попытаемся кратко обобщить сказанное и ответить на вопросы, которые неизбежно возникнут у читателей.

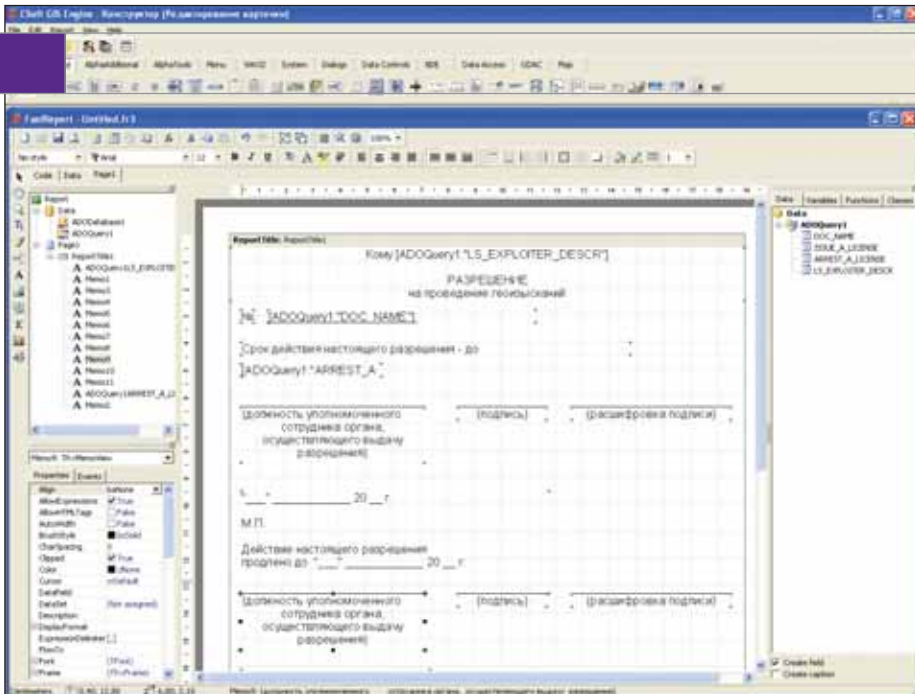
1. Использование CS GIS Engine позволит легко расширить возможности комплексных ГИС-решений от CSoft силами персонала заказчика, не нарушая интегральной целостности структур данных и базовых приложений.

2. "Сосуществование" базовых программных приложений от CSoft и приложений, созданных на основе CS GIS Engine, не требует дополнительных усилий по сопровождению программного комплекса, обеспечивая тем не менее его уникальность.
3. CS GIS Engine может успешно использоваться при интеграции ГИС-проектов в существующие системы заказчика благодаря богатому арсеналу средств взаимодействия с любыми стандартными информационными средами.

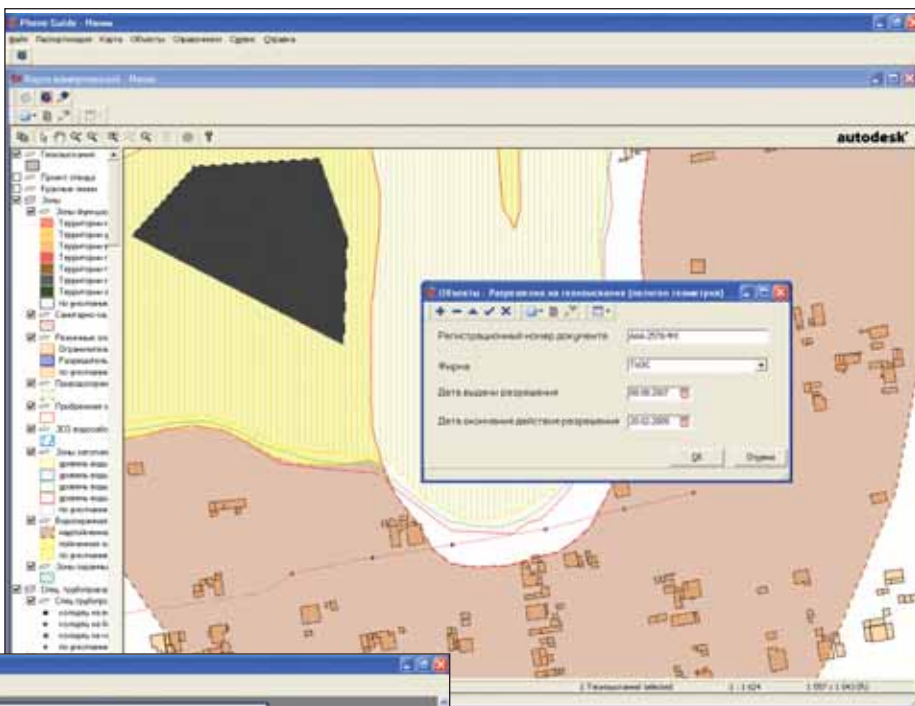
Таким образом, среда разработки пользовательских ГИС-приложений CS GIS Engine, доступная к поставке уже со II квартала текущего года, обеспечит быстрое и эффективное решение дополнительных задач, использующих данные, накапливаемые в процессе внедрения комплексных ГИС-проектов в области градостроения, муниципальных ГИС и систем мониторинга инженерных коммуникаций.

При этом гарантированно сохраняется целостность как структуры данных, так и самих данных, используется единый подход к администрированию системы и распределению прав доступа, отсутствует необходимость обращения за каждой "мелочью" к коллективу разработчиков CSoft и... расширяется круг людей, работающих для достижения единой цели: построения надежных, гибких и эффективных ГИС-проектов любого уровня сложности и детализации.

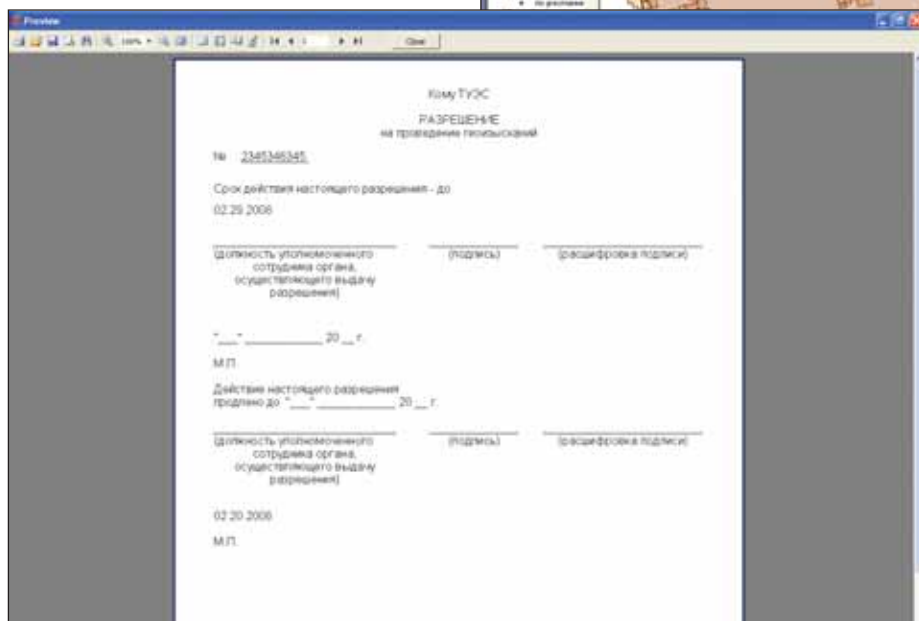
Александр Ставицкий,
директор по ГИС-направлению CSoft
Тел.: (495) 913-2222
E-mail: asta@csoft.com



Печатная форма документа в дизайнере отчета



Готовая конфигурация в работе



Печатная форма документа разрешения