

Министерство образования и науки Российской Федерации
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет информатики
Кафедра теоретических основ информатики

ДОПУСТИТЬ К ЗАЩИТЕ В ГАК
зав. кафедрой ТОИ, д.т.н., проф.
_____ Ю.Л. Костюк
« ___ » _____ 2004 г.

Снежко Виктор Валерьевич

**РАЗРАБОТКА ПОДСИСТЕМ ПРОЕКТИРОВАНИЯ
ИНЖЕНЕРНОГО ОБУСТРОЙСТВА ДЛЯ САПР IndorCAD/Road**

Дипломная работа

Научный руководитель,
Профессор кафедры ТОИ, д.т.н. _____ А.В. Скворцов

Исполнитель,
студ. гр. 1491 _____ В.В. Снежко

Электронная версия дипломной работы помещена
в электронную библиотеку. Файл
Администратор

Томск 2004

Реферат

Дипломная работа, 62 стр., 29 рис., 15 источников, 1 прил.

САПР, ИНЖЕНЕРНЫЕ КОММУНИКАЦИИ, INDORCAD, ПРОЕКТИРОВАНИЕ ДОРОГ, ТЕХНОЛОГИЯ СОМ, DELPHI, ИНЖЕНЕРНОЕ ОБУСТРОЙСТВО

Объект исследования – системы автоматизированного проектирования, принципы построения их элементов.

Цель работы – разработка технологии проектирования инженерных коммуникаций и дорожной разметки для САПР автомобильных дорог IndorCAD/Road.

Метод исследования – экспериментальный.

Разработаны подсистемы проектирования инженерных коммуникаций и дорожной разметки для САПР IndorCAD/Road, исследованы принципы построения САПР.

Модуль находится в стадии промышленной эксплуатации в составе САПР автомобильных дорог IndorCAD/Road в ООО «ИДЦ Индор» (г. Томск), ОАО «Томгипротранс» (г. Томск), ГУП «АлтайИндорПроект» (г. Барнаул), ОАО «КрасИндорПроект» (г. Красноярск), ООО НППФ «Краснодаравтосервис» (г. Краснодар), ОАО «Каздорпроект» (г. Астана, Казахстан)

Содержание

ВВЕДЕНИЕ	4
1. ОБЗОР СУЩЕСТВУЮЩИХ СИСТЕМ И ПОСТАНОВКА ЗАДАЧИ	6
1.1. Проектирование окружения объектов универсальными средствами САПР IndorCAD/Road	6
1.2. Проектирование инженерных коммуникаций с помощью других систем	7
1.3. Проектирование элементов инженерного обустройства с помощью других систем	8
1.4. Постановка задачи	10
2. ОБЩЕЕ ОПИСАНИЕ САПР INDORCAD/ROAD	12
2.1. Возможности системы	12
2.2. Структура системы	14
<i>План</i>	14
<i>Продольный профиль</i>	15
<i>Поперечный профиль</i>	15
<i>3D-вид</i>	16
3. МОДУЛИ РАСШИРЕНИЯ САПР INDORCAD/ROAD.	18
3.1. Описание общей структуры модуля расширения для САПР IndorCAD	18
3.2. ActiveX - интерфейс программирования модулей расширения САПР IndorCAD	20
3.3. Методика разработки модулей расширения	24
4. ПОДСИСТЕМА ПРОЕКТИРОВАНИЯ ИНЖЕНЕРНЫХ КОММУНИКАЦИЙ	29
4.1. Структура подсистемы	29
4.2. Реализованная функциональность	37
4.3. Применение обобщённых алгоритмов	38
5. ПОДСИСТЕМА ПРОЕКТИРОВАНИЯ ДОРОЖНОЙ РАЗМЕТКИ	41
5.1. Структура подсистемы	41
5.2. Реализованная функциональность	46
ЗАКЛЮЧЕНИЕ	48
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	49
ПРИЛОЖЕНИЕ 1. ПОДСИСТЕМА ПРОЕКТИРОВАНИЯ ИНЖЕНЕРНЫХ КОММУНИКАЦИЙ. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ.	50
П 1.1. Начало работы	50
П 1.2. Создание и удаление коммуникаций	51
П 1.3. Настройка отображения коммуникаций	54
П 1.4. Выделение коммуникаций	55
П 1.5. Параметры коммуникаций	58
П 1.6. Редактирование коммуникаций	60

Введение

Современной тенденцией в САПР является разработка специализированных систем для проектирования дорог, трубопроводов, ЛЭП, генеральных планов и т.п. Особенность поколения систем, разрабатываемых и существующих в настоящее время, заключается в том, что необходимо обеспечить работу с объектами проектирования, на базе которых динамически создается, а в дальнейшем развивается и дополняется новыми данными пространственная модель местности. На основе данных из этой модели генерируются чертежи, разрезы (профили), а также решаются многие прикладные задачи.

Важным при проектировании объектов является их окружение: рельеф, строения, инженерные коммуникации, элементы инженерного обустройства. Так, и при проектировании новых дорог, и при создании проектов реконструкции/ремонта существующих жизненно необходим доступ к информации о расположении в окрестности проектируемой дороги зданий и строений, ограждений, озеленения, а также о расположении инженерных коммуникаций. Это позволяет оценить объём работ и принять решение о том, как спроектировать работы таким образом, чтобы минимизировать затраты ресурсов на решение задач, прямо не связанных с проектированием собственно дороги (снос зданий и сооружений, удаление озеленения, временное отключение участков инженерных коммуникаций).

Помимо всего прочего, на более поздних стадиях проектирования объектов появляются задачи актуального отражения других элементов инженерного обустройства: дорожных знаков, разметки, ограждений, столбиков и т.д. При этом необходимо решать задачи не только отображения этих объектов, но и производить различные расчёты.

В то же время, современные системы автоматизированного проектирования не всегда обеспечивают удобный доступ к информации об окружении проектируемых объектов на всех этапах проектирования. Кроме того, многие процессы проектирования, которые могли бы быть автоматизированы, после чего занимали бы на порядок меньше времени, до сих пор часто выполняются вручную.

Данная работа направлена на решение задач, связанных с проектированием инженерных коммуникаций и элементов инженерного обустройства проектируемых объектов, а также на органичное встраивание подсистем проектирования окружения в существующую систему моделирования рельефа.

Данная дипломная работа состоит из пяти глав.

В первой главе проводится анализ современного состояния проблемы, а также рассматриваются подходы к проектированию информации об окружении проектируемых объектов.

Во второй главе рассматриваются возможности и структура системы IndorCAD/Road.

В третьей главе обсуждаются принципы разработки модулей расширения САПР IndorCAD/Road.

В четвёртой и пятой главах рассматриваются разработанные автором подсистемы проектирования инженерных коммуникаций и дорожной разметки для САПР IndorCAD/Road.

В заключении сформулированы основные выводы по работе.

Основным результатом данной дипломной работы является создание подсистем проектирования инженерных коммуникаций и дорожной разметки для системы автоматизированного проектирования IndorCAD/Road.

1. Обзор существующих систем и постановка задачи

1.1. Проектирование окружения объектов универсальными средствами САПР IndorCAD/Road

В составе САПР IndorCAD/Road имеются инструменты для проектирования произвольных точечных, линейных и площадных объектов. С их помощью возможно решение многих задач проектирования. Например, достаточно эффективно реализуется создание планов окружения проектируемых дорог, частью которого являются коммуникации, и инженерного обустройства, в том числе дорожной разметки. Однако создавать многие документы и чертежи проекта приходится вручную, несмотря на то, что автоматизация создания этих документов возможна. Так, наиболее трудоёмкая для ручного исполнения операция – это создание продольных и поперечных разрезов (профилей) существующего рельефа по оси (или перпендикулярно ей) проектируемой трассы. В ГОСТ Р 21.1701-97 «Правила выполнения рабочей документации автомобильных дорог» содержится требование отображать на чертежах и продольных, и поперечных профилей места прохождения инженерных коммуникаций. Существующая на момент принятия решения о необходимости разработки подсистемы проектирования инженерных коммуникаций САПР IndorCAD/Road не позволяла этого. На цифровую модель местности не наносилась информация о глубине прохождения коммуникаций – она заносилась по результатам изысканий в отдельные ведомости и никак не фигурировала в модели. Это ни в коей мере не мешало создавать из модели планы местности, но профили было создавать невозможно. И это приходилось делать вручную, на что у инженеров-проектировщиков уходила масса времени.

Аналогичная ситуация сложилась и с процессом проектирования инженерного обустройства: дислокации дорожной разметки, дорожных знаков, различного рода ограждений, столбиков и т.д. Существующая САПР не позволяет эффективно проектировать эти объекты инженерного обустройства.

Однако эти проблемы поддаются решению. САПР IndorCAD/Road имеет широкие возможности расширения, основанные на модели подключаемых модулей и на компонентной модели объектов (COM) фирмы Microsoft [9]. Использование этих концепций делает возможным безболезненное расширение системы. Для этого

необходимо разработать подсистему, которая обладает необходимой функциональностью, и встроить её в САПР IndorCAD/Road.

1.2. Проектирование инженерных коммуникаций с помощью других систем

В настоящее время на рынке САПР наблюдается широкое разнообразие систем проектирования инженерных коммуникаций. Их можно разделить на два класса:

1. Специализированные системы проектирования инженерных коммуникаций. К этому классу относятся системы:

- ИНЖКАД - программный комплекс для проектирования наружных коммуникаций.
- «WinELSO» - комплекс программ для проектирования и расчета систем силового электрооборудования и электроосвещения.
- PVS – комплекс модулей для САПР AutoCAD, служащий для построения продольных профилей наружных коммуникаций.

Все эти системы позволяют решать задачи, связанные с проектированием инженерных коммуникаций, а также производить сложные инженерные расчёты, однако все из них настолько узкоспециализированы, что не позволяют, например, работать со всеми видами инженерных коммуникаций. Кроме того, отсутствие интеграции с широко распространёнными системами автоматизированного проектирования дорог чрезвычайно затрудняет их использование в дорожной отрасли.

1. Системы автоматизированного проектирования дорог, включающие инструменты для проектирования коммуникаций. Кроме рассмотренной выше САПР IndorCAD/Road, в эту категорию попадает САПР Credo, разработанная в минской компании «Кредо-Диалог». Этот комплекс программ имеет длинную историю и позволяет решать множество разнообразных задач проектирования, от задач обеспечения дорожного строительства до проектирования инженерных коммуникаций и производства расчётов по ним. Однако его пользовательский интерфейс оставляет желать лучшего, в чём можно убедиться при просмотре демонстрационных роликов, доступных на сайте компании – <http://www.credo-dialogue.com>. Например, подсистема проектирования

инженерных коммуникаций выполнена в качестве внешнего модуля к САПР «AutoCAD», и процесс проектирования представляет собой борьбу с интерфейсом программы, для победы в которой приходится постоянно переключаться между полями ввода, создаваемыми из AutoLISP-подпрограмм, такими же меню, а также меню самого AutoCAD'а и регулярно всплывающими диалоговыми окнами, в которые необходимо вручную вводить параметры для каждого проектируемого колодца (нет возможности задать эти параметры один раз по умолчанию). Кроме того, процесс ввода данных не только затруднён, но и ненагляден – координаты колодцев необходимо вводить вручную, так как в этой подсистеме отсутствует визуальная привязка к цифровой модели местности.

1.3. Проектирование элементов инженерного обустройства с помощью других систем

Разнообразие средств проектирования элементов инженерного обустройства на современном рынке САПР не столь велико. В основном инженеры-проектировщики поступают следующим образом. Сначала с помощью высокоуровневой САПР подготавливается цифровая модель местности, проектируемого объекта и его окружения. На следующей стадии проектирования из него генерируются отчётные документы: ведомости, таблицы, продольные и поперечные профили. Кроме этого, генерируется основной отчётный документ – план, который представляет собой совокупность чертёжей, доступных для редактирования в системах подготовки чертежей – IndorDraw, AutoCAD и других. После этого чертежи редактируются, и элементы инженерного обустройства – не только дорожная разметка, но и дорожные знаки, столбики, ограждения – заносятся в них вручную с помощью указанных выше систем.

У этого подхода есть как достоинства, так и недостатки. Неоспоримое достоинство – простота подхода. Однако этим достоинства и ограничиваются. Потому что стоит немного изменить проект – сместить или, тем более, повернуть ось проектируемой дороги, изменить окружение – чертёж необходимо генерировать повторно. А при этом теряет смысл ранее произведенное проектирование элементов инженерного обустройства. Потому что на вновь сгенерированном чертеже они (естественно) отсутствуют, приходится делать заново. У инженеров-

проектировщиков даже выработались стратегии обхода этого недостатка – проектирование происходит поэтапно, они стремятся минимизировать количество генераций чертёжей из модели. Однако всё это – борьба с симптомами, не затрагивающая основную болезнь – недостаточную автоматизацию системы, работающей с моделью.

Существуют и альтернативные специализированные системы проектирования дислокации дорожной разметки. Наиболее распространённая из них так и называется – «Система проектирования схем дислокации дорожных знаков и разметки» – и разработана в ФГУП СНИЦ «Росдортех». Система обеспечивает ввод и отображение данных о характеристиках проезжей части (километр начала и конца участка, количество полос на участке или количество полос в начале и конце участка отгона, о дорожной разметке (километр начала и конца участка, расположение на полосах движения, тип), о дорожных знаках (пикетажное местоположение, код по ГОСТ, наименование по ГОСТ, расположение относительно оси дороги, об объектах на дороге и их характеристиках, необходимых для проектирования). Однако в силу узкой специализации системы невозможно при проектировании видеть план местности, на которой проектируется дорога. При этом не только пропадает возможность привязывать элементы инженерного обустройства ко многим объектам местности, но и исчезает наглядность проектирования. Так, вся дорога представляется в виде совокупности прямых участков, которые не только внутри системы являются прямыми, но и отображаются в совокупности в виде одной непрерывной прямой полосы (даже если дорога не прямая, как обычно и бывает). Поэтому использование этой системы неприемлемо.

В настоящее время разрабатывается ещё одна система, которая имеет возможность обеспечивать проектирование дислокации дорожной разметки. Это т.н. «Расчетная Информационная Среда «Автодороги»», разработанная в тверской фирме «Компалекс». Эта среда является обычной дорожной информационной системой, к которой добавлены функции дислокации дорожных знаков и разметки, отсюда её недостатки: только базовые возможности отображения, отсутствие информации о многих объектах окружения. Кроме того, модуль проектирования дорожной разметки в настоящее время находится в стадии разработки, поэтому его обзор невозможен.

Помимо всего этого, необходимо отметить общий недостаток рассмотренных систем – все они узкоспециализированы и не позволяют производить весь комплект проектных работ или хотя бы приблизиться к этому. Отсутствует крайне удобная

возможность отображения местности в 3D-виде. Системы не только не обладают сколько-нибудь значимыми достоинствами по сравнению с САПР IndorCAD, но и имеют серьёзные недостатки, затрудняющие их использование при проектировании дорог. Поэтому любая разработка на основе этих систем не была бы экономически оправдана. Гораздо эффективнее доработать систему, в которой есть почти всё необходимое.

1.4. Постановка задачи

Таким образом, принимая во внимание всё вышесказанное, было принято решение доработать систему автоматизированного проектирования IndorCAD/Road: разработать для неё необходимые для облегчения проектирования подсистемы. В данную дипломную работу вынесены две подсистемы, как наиболее приоритетные и позволяющие в наибольшей степени облегчить труд инженеров-проектировщиков. Первая из них – подсистема проектирования инженерных коммуникаций в виде модуля для САПР IndorCAD/Road, который бы позволял:

1. Проектировать инженерные коммуникации всех необходимых видов универсальным образом;
2. Обеспечивать при этом наглядность процесса – осуществлять проектирование прямо на изображении плана местности, генерируемого динамически из цифровой модели местности;
3. Обеспечивать учёт параметров коммуникаций – диаметров труб, глубин колодцев, напряжений электрических кабелей и т.д.
4. Обеспечивать экспорт данных о расположении инженерных коммуникаций в продольные и поперечные профили, построенные по трассам проектируемых новых и реконструируемых дорог, а также в план;
5. Обеспечивать отображение коммуникаций в окне трёхмерного вида местности САПР IndorCAD;

Вторая подсистема – подсистема проектирования дорожной разметки, которая также в немалой степени могла бы автоматизировать труд инженеров-проектировщиков. Эта подсистема должна обладать следующими функциями:

1. Обеспечивать проектирование всех типов дорожной разметки в соответствии с ГОСТ 51256-99 «Разметка дорожная. Типы и основные параметры. Общие технические требования».
2. Позволять проектировать разметку прямо на плане местности с нанесённой трассой проектируемой дороги.
3. Обеспечивать учёт параметров разметки, привязки к проектируемой трассе и отображение в зависимости от этих параметров.
4. В перспективе – обеспечивать отображение разметки в окне трехмерного вида местности САПР IndorCAD.

2. Общее описание САПР IndorCAD/Road

2.1. Возможности системы

Система IndorCAD/Road позволяет проектировать автомобильные дороги всех категорий на стадии их строительства, реконструкции и ремонта. Проектирование в системе IndorCAD/Road можно разбить на следующие основные этапы:

- ввод исходных данных;
- построение цифровой модели местности (ЦММ);
- построение цифровой модели проекта (ЦМП):
 - проектирование плана трассы (трассирование, разбивка трассы),
 - проектирование продольного профиля трассы,
 - проектирование верха земляного полотна,
 - проектирование поперечных профилей и дорожной одежды;
- визуальная оценка проекта по его трехмерному изображению (возможна на любом этапе проектирования);
- подготовка проектной документации (чертежи, ведомости, таблицы).

Отличительной особенностью системы IndorCAD/Road является принцип единой модели дороги, то есть любые изменения в одной из проекций трассы (плане, продольном или поперечных профилях) мгновенно отображаются в остальных проекциях и 3D-виде. Такой подход позволяет отслеживать результаты выполненных изменений в режиме реального времени и получать непротиворечивые проектные решения.

Инструментальные средства системы позволяют:

- обрабатывать геодезическую информацию, полученную различными методами (нивелирование, тахеометрическая съемка, GPS-съемка);

- на основе обработанных данных формировать цифровые модели местности, редактировать их, отображать в различных представлениях (изолинии, твердотельная модель, уклоны и т.д.) для визуального анализа;
- выполнять трассирование автомобильной дороги (построение в плане кривой, описывающей траекторию дороги с учетом ограничений на допустимые радиусы поворотов) как с применением классических геометрических элементов (дуг окружностей и клотоид), так и современных инструментов вычислительной математики (кривых Безье 3-го и 5-го порядков);
- осуществлять параллельный перенос трассы, изменять азимут ее начального направления;
- выполнять разбивку трассы на поперечные профили (дискретизацию отметок трассы с заданным интервалом);
- проектировать продольный профиль трассы классическим методом или с помощью сглаживающих сплайнов;
- проектировать верх земляного полотна (виражи), в том числе и с учетом сплайновой природы трасс;
- конструировать дорожную одежду и поперечные профили как типовые, так и индивидуальные;
- отображать 3D-вид существующей и проектируемой поверхностей;
- отображать на экране все проекции проектируемого объекта одновременно;
- объединять проекты;
- формировать чертежи, ведомости и таблицы для последующего их редактирования соответственно в IndorDraw и MS Excel;
- используя богатый ActiveX-интерфейс, создавать собственные модули расширения и надстройки системы для выполнения частных задач.

2.2. Структура системы

Система IndorCAD/Road состоит из пяти основных компонент: план, продольный профиль, верх земляного полотна, поперечный профиль, 3D-вид. Данные всех компонент взаимосвязаны и любые изменения, выполненные в плане, продольном или поперечном профиле мгновенно отображаются во всех окнах. Ниже приведено краткое описание этих компонент.

План

Окно плана является основным в системе IndorCAD/Road. В нем отображаются в виде плана все объекты проекта: растры, поверхности (ЦММ, ЦМП), трассы и другие. Здесь осуществляется построение ЦММ по исходным данным, проектирование плана трассы и построение ЦМП.

Для построения ЦММ и ЦМП в системе IndorCAD/Road используется триангуляционная модель Делоне с ограничениями. Эффект "псевдо 3D" (изменение цвета треугольников триангуляции в зависимости от направления уклонов и источника света), изолинии и градиенты стоков позволяют визуально оценить рельеф построенных поверхностей. Используя инструменты для создания и редактирования структурных и ситуационных линий, можно представить ЦММ в виде карты (топоплана).

Трассирование дороги и разбивка трассы осуществляется через главное меню системы IndorCAD/Road и/или специальную панель инструментов. Проект может содержать любое количество трасс (основная, вспомогательные, примыкания, пересечения и др.). Каждой трассе соответствует карточка объекта, где можно задать индивидуальные параметры трассирования. В системе IndorCAD/Road реализованы как классические схемы трассирования дорог посредством тангенциального хода с закруглениями типа "клотоида - круговая кривая - клотоида", так и схем на основе кривых Безье 3-го и 5-го порядков. Последние позволяют осуществлять пространственное трассирование дорог, что особенно эффективно при проектировании транспортных развязок.

Объекты проекта (растры, поверхности, элементы поверхностей, трассы) могут одновременно или поочередно отображаться на экране. Это позволяет визуально анализировать полученные решения.

Проект может быть оформлен в соответствии с существующими требованиями оформления топографических планов и проектов автомобильных дорог. Для этого в

системе IndorCAD/Road предусмотрено достаточно большое число вспомогательных инструментов для работы с точками, линиями, полигонами, трассами и т.п. Для оформления этих объектов применяется технология Eх-шрифтов отображения условных знаков.

Продольный профиль

Проектирование продольного профиля трассы осуществляется в редакторе продольного профиля. Продольный профиль трассы представляет совокупность дискретных точек проектной линии, которые должны удовлетворять ограничениям на максимальные допустимые уклоны и минимальные вертикальные радиусы изгибов трассы.

При проектировании продольного профиля в системе IndorCAD/Road можно использовать классический или сплайновый метод. При использовании сплайнового метода система выполняет автоматический поиск наиболее подходящего (оптимального) решения с учетом различных ограничений, накладываемых на точки проектной линии (допустимые вертикальные перемещения точек). Оптимизация проектной линии сглаживающими сплайнами особенно эффективна при реконструкции и ремонте автомобильных дорог, когда в узкой полосе варьирования требуется найти наиболее плавное очертание проектной линии, проходящей через заданные точки. Следует отметить, что IndorCAD/Road является единственной в России системой, которая обеспечивает сплайн-сглаживание с последующим ручным редактированием проектной линии.

Для удобства работы в редакторе продольного профиля автоматически строятся графики рабочих, проектных и интерполированных отметок, графики кривизны, уклонов и другие. Если на плане нанесены скважины, то в продольном профиле отображаются их геологические разрезы. Для учета состояния грунтов в месте планируемого строительства дороги в системе IndorCAD/Road предусмотрены специальные инструменты для ввода информации по геологическим колонкам и построения графиков залегания грунтов.

Поперечный профиль

Проектирование поперечных профилей трассы осуществляется в редакторе поперечного профиля. При моделировании проектной поверхности можно использовать

уже существующие модели из библиотеки типовых решений по поперечным профилям либо строить новые. Проектная линия профиля может быть автоматически доведена до пересечения с существующей или интерполированной поверхностью. С помощью именованных структурных линий можно моделировать конструкции дорожной одежды существующей и проектируемой дороги. В качестве элементов дорожной одежды используются прикромочные лотки, бордюры, основания под бордюры и другие. Отметим, что модель любого поперечного профиля можно поместить в библиотеку, либо применить к другим участкам трассы. В этом редакторе также проектируются полосы временного и постоянного отвода земель, интерполированной поверхности и срезки растительного слоя.

3D-вид

Одной из важных задач, выполняемых при проектировании автомобильных дорог, является визуальная оценка проектного решения. При этом оцениваются такие параметры решения, как достаточная пространственная видимость трассы, видимость на поворотах и примыканиях, правильность расстановки и видимость дорожных знаков, отсутствие частых изгибов трассы в плане и продольном профиле (по вертикали), отсутствие чрезмерно длинных прямых перегонов и другие.

В системе IndorCAD/Road для визуальной оценки решений разработан модуль трехмерной визуализации, позволяющий достаточно реалистично представить проект вместе с инженерным обустройством, зелеными насаждениями и другими объектам, расположенными вдоль дороги. Проектирование таких объектов осуществляется в окне плана проекта, но вместе с условными обозначениями объектов на плане формируются их 3D-аналоги. В число стандартных трехмерных элементов входят коммуникации, дорожные знаки, ограждения, здания и деревья. Дорожные знаки и деревья являются самостоятельными объектами плана. В виде зданий и ограждений могут отображаться полигональные и линейные объекты.

Просмотр трехмерного вида проекта выполняется в окне 3D-вида. Пользователю предоставляется возможность интерактивного перемещения в пространстве по свободной траектории и "проезда" по трассе, когда отображается вид на дорогу с точки зрения водителя.

Результаты пролета над дорогой или проезда по ней могут быть записаны в видеофайл AVI для последующей демонстрации без системы IndorCAD/Road. Такой

приём с демонстрацией видеофильма может быть очень полезен, например, при защите выполненных проектов у заказчика.

3. Модули расширения САПР IndorCAD/Road.

В данной главе описывается методика разработки модулей расширения для САПР IndorCAD. Сначала описывается общая структура модуля расширения САПР IndorCAD, затем описывается необходимая часть интерфейса прикладного программирования, затем эта информация обобщается.

3.1. Описание общей структуры модуля расширения для САПР IndorCAD

Модуль расширения для САПР IndorCAD/Road должен представлять собой динамически подключаемую библиотеку (DLL) [7], которая экспортирует 3 функции и процедуры:

```
function GetModuleDescription:PChar; stdcall;  
procedure IndorCADPluginInit(const aApp:IApplication); stdcall;  
procedure IndorCADPluginDone; stdcall;
```

Первая из них – процедура IndorCADPluginInit – вызывается, когда модуль только только загружен системой. При этом модулю передаётся указатель на класс, реализующий СОМ-интерфейс IApplication (описания интерфейсов см. в следующем пункте), который предоставляет необходимую функциональность для того, чтобы модуль смог сообщить, что он действительно является модулем расширения (для этого необходимо знать описание интерфейса IApplication), для регистрации модуля в приложении и настройки параметров регистрации, а также служит средством доступа ко всем остальным частям приложения.

Процедура IndorCADPluginDone предназначена для уведомления модуля расширения о том, что работа приложения завершается и необходимо освободить ресурсы и вызывается при завершении работы системы IndorCAD.

Функция GetModuleDescription предназначена для выдачи описания модуля и используется для внутренних нужд САПР IndorCAD/Road.

Рассмотрим подробнее, что происходит при регистрации модуля расширения в приложении. Модуль может регистрироваться на одном из трёх уровней – на уровне проекта, на уровне слоя проекта и на уровне трассы [15]. Уровень определяет, какие уведомления будут приходить модулю. Например, модуль, регистрирующийся на уровне слоя проекта, будет получать уведомления о том, что создан новый слой. Механизм таков:

- Ядром САПР IndorCAD вызывается функция «IndorCADPluginInit» модуля;

- Ей передаётся в качестве параметра указатель на класс, реализующий интерфейс IApplication,
- У этого класса вызывается метод AddDataPlugin. Его сигнатура выглядит следующим образом:

```
function AddDataPlugin(
    PluginDataPlacement: PluginDataPlacementConstants;
    const DataPlugin: ICADDataPlugin;
    UserData: Integer;
    const DataSignature: WideString
): Integer; safecall;
```

При вызове этого метода необходимо передать следующие параметры:

- константу, определяющую уровень, на котором будет регистрироваться модуль расширения;
 - фабрику класса (см. также [2]) – в параметре DataPlugin – которой будут приходить уведомления о создании новой сущности приложения (проекта, слоя или трассы) и которая при этом будет должна создать экземпляр класса-коллекции объектов и вернуть его в качестве результата;
 - в случае необходимости – любые свои данные через параметр UserData;
 - через параметр DataSignature – сигнатуру своего модуля – уникальную строку, которая могла бы идентифицировать модуль в системе.
- Дальнейшие обращения к модулю будут происходить через созданную фабрикой коллекцию.

Каждая коллекция должна поддерживать интерфейс ICADObject, с помощью которого осуществляются основные обращения к модулю. В основном класс, поддерживающий этот интерфейс, должен реализовывать различные служебные методы, в числе которых, например, присутствуют: Get_GUID(), возвращающий уникальный идентификатор объекта, Get_Name() – возвращающий читаемое человеком имя объекта, Set_Name, позволяющий изменить его, и другие. Наиболее полезная функциональность интерфейса – он позволяет организовывать древовидную структуру контейнеров. Это реализуется с помощью методов Get_ChildCount(), Get_IChild(), а также

MakeTreeStructure(). Метод Get_ChildCount() возвращает количество дочерних объектов, метод Get_IChild – сами дочерние объекты по номеру.

Если модулю необходимо реализовать какую-то дополнительную функциональность (кроме описанной выше), то основной контейнер его объектов (возвращаемый фабрикой класса при создании новой сущности приложения) и/или дочерние контейнеры могут реализовать другие интерфейсы в дополнение к ICADObject.

3.2. ActiveX - интерфейс программирования модулей расширения САПР IndorCAD

Модули расширения САПР IndorCAD программируются при помощи COM-интерфейсов (подробное описание можно найти в [8]). Основные интерфейсы, необходимые при разработке, следующие:

Интерфейс ICADDataPlugin

Данный интерфейс поддерживают объекты, которые хранят (или могут создавать) данные подключаемого модуля. Обычно эти объекты представляют собой фабрики классов-коллекций объектов, обрабатываемых в модуле. Объект, поддерживающий данный интерфейс, должен реализовать одну единственную функцию CreateInstance(), на вход которой передаётся контекст, в котором должен быть создан объект (это может быть проект, слой или трасса), текущий проект, а вернуть эта функция должна вновь созданный класс-коллекцию объектов модуля.

Интерфейс ICADActionResponder

В системе IndorCAD используется событийная модель управления подсистемами. События могут приходиться при нажатии кнопки на панели инструментов, при выборе пункта меню и осуществлении прочих действий. Данный интерфейс используется для управления откликом на события. Объект, который желает получать уведомления на такого рода события, обычно автоматически подписывается на них при создании собственного элемента управления (функциям создания элементов управления передаётся указатель на реализацию данного интерфейса – этим реализациям уведомления впоследствии и рассылаются). Объект, реализующий данный интерфейс, должен реализовать две функции: OnUpdate и OnExecute. Когда ядро вызывает первую функцию, этим самым оно запрашивает у модуля статус данного элемента управления: должен ли он в данный момент быть разрешённым к нажатию и/или включенным. Вызов второй

функции – OnExecute – это уведомление модуля о том, что событие только что произошло (например, нажата кнопка) и необходимо его обработать.

Интерфейс ICADTreeObject

Этот интерфейс предназначен для обеспечения отображения древовидной структуры объектов в отдельном окне. Класс, поддерживающий интерфейс, должен реализовать несколько методов: DrawToTreeView(), который позволяет модулю самому отрисовать как текст, так и (при необходимости) любой другой графический объект; Get/SetCheckedInTree – узнать/изменить пометку о том, что объект выбран в дереве, и Get/SetExpanded – узнать/изменить атрибут «дерево раскрыто».

Интерфейс IPropertyStreamed

Предназначен для облегчения реализации свойства сохраняемости объектов в постоянном (persistent) хранилище. Объект, который хочет быть сохранён в проекте, обязан поддерживать этот интерфейс. Для этого необходимо реализовать два метода: LoadFrom и SaveTo, которые вызываются при загрузке проекта и при его сохранении, соответственно. В качестве параметров обоим методам передаётся указатель на реализацию интерфейса IPropertyStream, который имеет удобные методы, обеспечивающие загрузку и сохранение большого количества типов данных: ReadBoolean()/WriteBoolean(), ReadInteger()/WriteInteger(), ReadDouble()/WriteDouble(), ReadDate()/WriteDate(), ReadString()/WriteString() – для булевских значений, целочисленных переменных, переменных с плавающей точкой, переменных типа дата/время и строк соответственно. Интерфейс также предоставляет возможность сохранять структурированные данные: он имеет методы CreateDir(), OpenDir() и Close(), которые позволяют сохранять любые структуры в отдельных каталогах. Когда необходимо сохранить структурированные данные, вызывают метод CreateDir(), затем Write-методы нужных типов данных для нужных переменных, и по окончании записи – Close(). При необходимости выполнить чтение из каталога его сначала необходимо открыть функцией OpenDir(), затем прочитать нужные данные Read-методами, после чего закрыть каталог функцией Close().

Интерфейс ICADObjectDraw

Объект, поддерживающий этот интерфейс, может сам себя отрисовывать в окне приложения. Для этого одним из параметров метода DrawTo передаётся указатель на реализацию интерфейса IPaintCanvas, который предоставляет необходимую

функциональность для отрисовки различных графических примитивов. Также имеются методы RealToScreen/ScreenToReal – для пересчёта реальных координат в экранные, а также GetDC – для получения контекста устройства, при необходимости осуществить низкоуровневые операции по отрисовке.

Рисунок 1 содержит пример того, как это выглядит с использованием подсистемы проектирования коммуникаций.

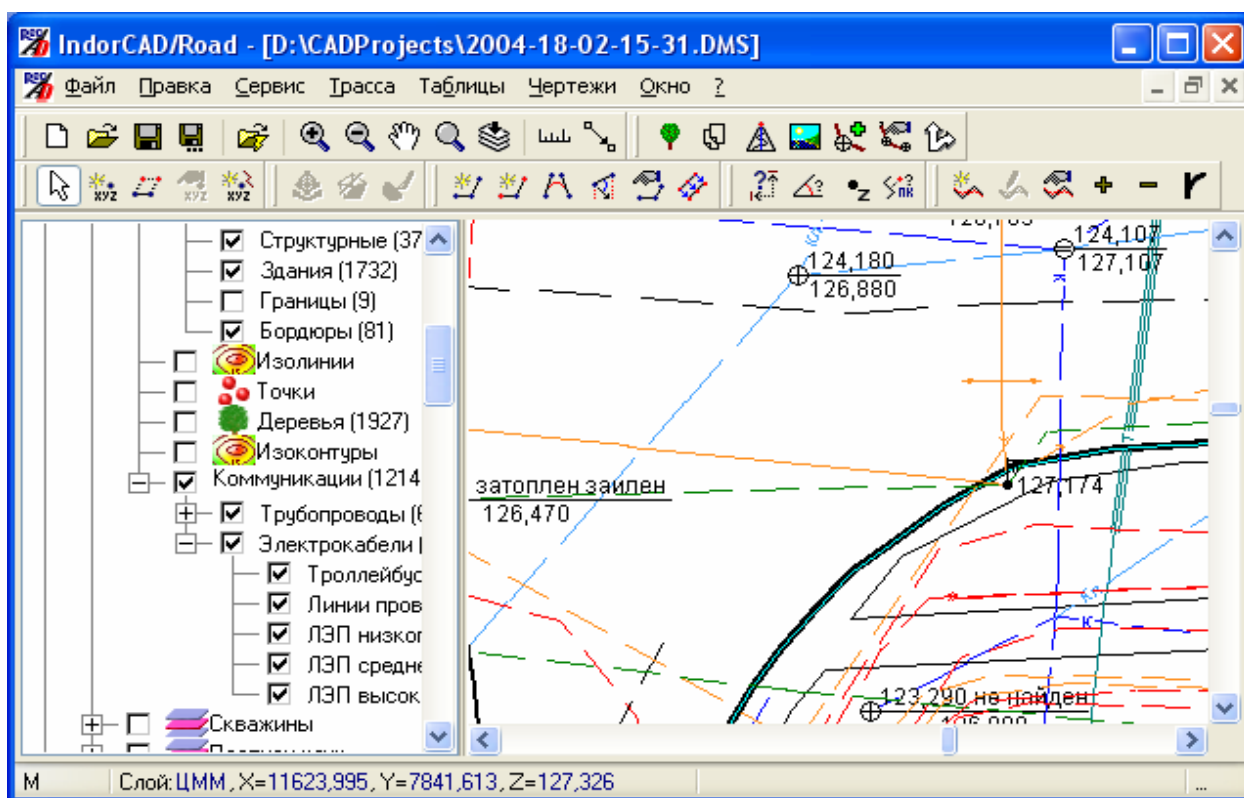


Рис.1. Фрагмент проектируемого объекта со включенными коммуникациями

Интерфейс IProfileDrawObject

Объект, поддерживающий этот интерфейс, может себя отрисовывать в окне продольного и поперечного профиля заданной трассы. Реализация интерфейса должна содержать два метода – DrawLongProf и DrawTransProf, которые предназначены для рисования продольного и поперечного профилей трассы, соответственно. Для этого первым параметром каждому методу передаётся указатель на трассу, профиль которой необходимо отрисовать, а вторым – указатель на реализацию интерфейса IPaintCanvas, который предоставляет необходимую функциональность для отрисовки различных графических примитивов. С их помощью и отображаются условные изображения объектов модуля.

Фрагмент отображённого профиля можно посмотреть на рисунке 2:

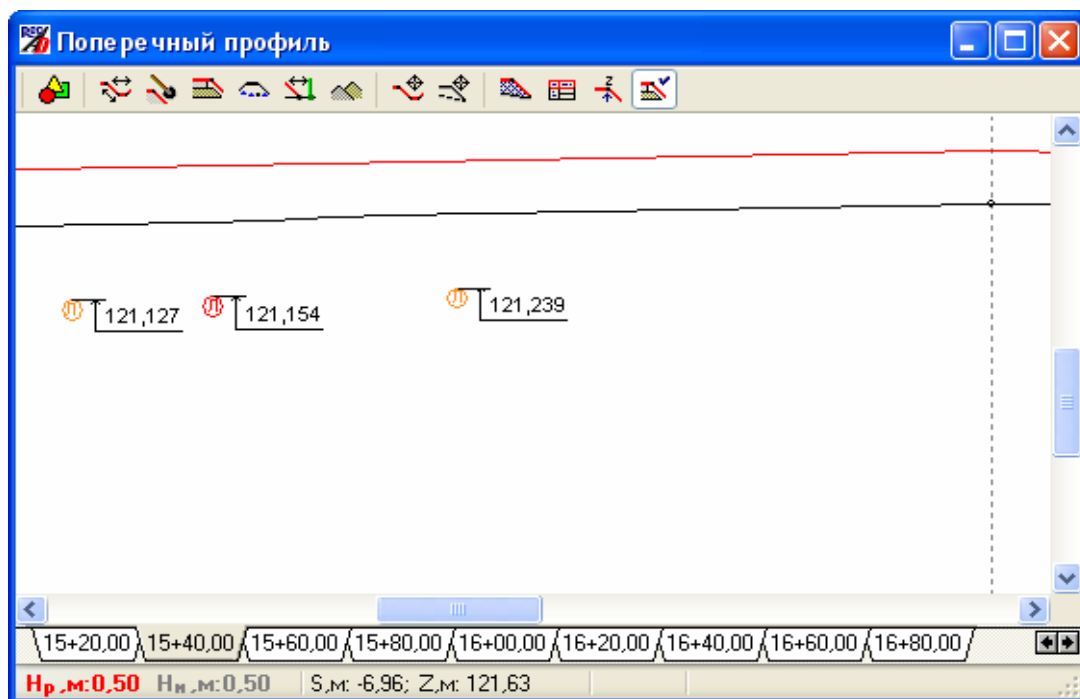


Рис. 2. Фрагмент поперечного профиля с отображёнными коммуникациями

Интерфейс IExportableToPlan

Объект, поддерживающий этот интерфейс, может экспортироваться в чертёж плана местности. Реализации необходимо отрисовать себя таким же образом, как и в окне приложения, только уже не с помощью методов интерфейса IPaintCanvas, а записать данные о себе непосредственно в файл чертежа. Обычно это делается не напрямую, а с помощью интерфейса, способного абстрагировать работу с разными форматами чертежей – IDrawingList (лист чертежа), указатель на реализацию которого передаётся методам интерфейса IExportableToPlan.

Интерфейс IExportableWithProfile

Объект, поддерживающий этот интерфейс, может экспортироваться в чертёж профиля трассы (продольного и/или поперечного). Для этого необходимо реализовать методы: GetTransBounds, ExportLongitudinalProfilePage и ExportTransverseProfilePage. Первый метод – GetTransBounds – позволяет получить диапазон изменения координат объектов, которые будут экспортированы в чертёж поперечного профиля. Остальные два метода позволяют объекту непосредственно экспортироваться в чертёж продольного и поперечных профилей, соответственно.

Рисунок 3 показывает, как выглядит сгенерированный чертёж:

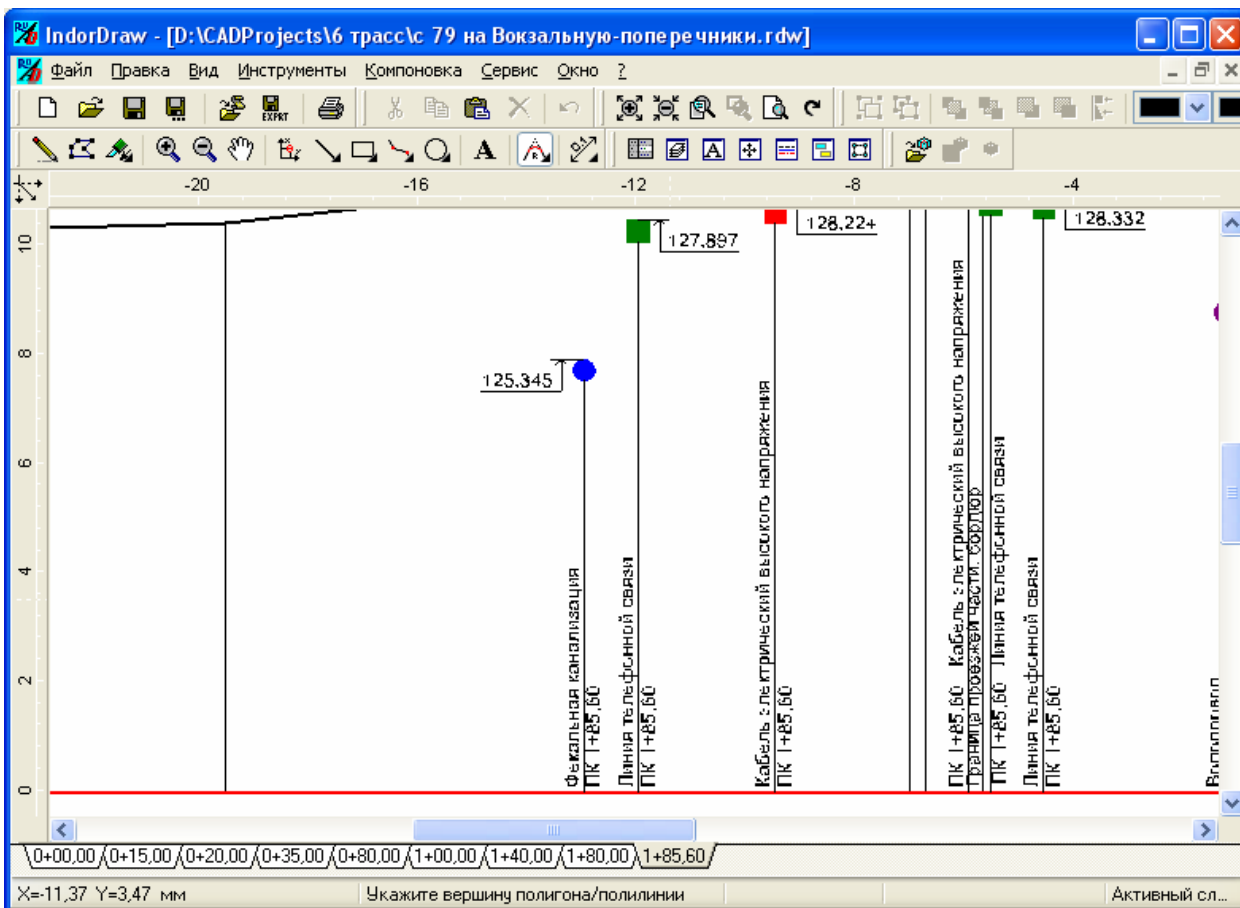


Рис. 3. Фрагмент сгенерированного чертежа поперечного профиля

Сгенерированный чертёж с помощью чертёжной системы можно модифицировать, подготовить к печати и отпечатать. Однако модификация чертежа – низкоуровневая операция проектирования, и этого следует избегать

Интерфейс IObject3D

Объект, поддерживающий этот интерфейс, может отображать себя в окне трёхмерного вида приложения. Для этого он должен реализовать 2 метода – DrawToD3D и Free3DResources. Метод DrawToD3D позволяет отрисоваться на Direct3D-устройство, указатель на реализацию которого передаётся в качестве параметра метода. Метод Free3DResources позволяет освободить выделенные по требованию (см. также lazy evaluation в [1], [3], [5]) в процессе рисования ресурсы. Он вызывается при закрытии 3D-вида.

3.3. Методика разработки модулей расширения

Таким образом, учитывая вышеизложенное, разработка модулей расширения для САПР IndorCAD/Road выглядит следующим образом.

Сначала необходимо создать ядро модуля: файл проекта динамически подключаемой библиотеки, подключить к ней описания интерфейсов, описать фабрику классов-коллекций и добавить три главные экспортируемые функции:

```
library Markup;  
  
uses  
  
    // Описания интерфейсов  
    CADObjs_TLB in '..\..\..\IDL\CADObjs_TLB.pas',  
    ActiveX,  
    ...;  
  
type  
    // фабрика классов-коллекций.  
    TMyCollectionCreator = class(TInterfacedObject, ICADDataPlugin)  
    protected // ICADDataPlugin  
        // фабричный метод, должен создать объект IndorCAD.  
        function CreateInstance(  
            const Context: IUnknown;  
            const Project: IProject;  
            UserData: Integer  
        ): ICADObject; safecall;  
  
    end;  
  
function TMyCollectionCreator.CreateInstance(  
        const Context: IInterface;  
        const Project: IProject;  
        UserData: Integer  
    ): ICADObject;  
  
var  
    Route: IRoute;  
  
begin  
    // к примеру, объекты модуля будут создаваться в  
    // контексте трассы  
    if Supports(Context, IRoute, Route) then  
        Result := TMyPluginRootCollection.Create(Route);  
  
end;  
  
function GetModuleDescription: PChar; stdcall;  
  
begin  
    Result := 'Мой новый модуль расширения САПР IndorCAD';  
  
end;
```

```

procedure IndorCADPluginInit(const aApp:IAApplication); stdcall;
var
    Creator: ICADDataPlugin
begin
    // Создаём фабрику классов-контейнеров
    Creator:= TMyCollectionCreator.Create(aApp);

    // Просим ядро зарегистрировать нашу фабрику класса.
    // Константой PluginDataPlacementRoute задаём, что
    // объекты модуля будут «жить» на уровне трассы
    aApp.AddDataPlugin(PluginDataPlacementRoute, Creator, 0);
end;

procedure IndorCADPluginDone; stdcall;
begin
    // Тут обычно освобождаются разные глобальные вещи.
    // По-хорошему, их быть не должно
end;

```

Обычно также возникает необходимость добавлять в систему свои элементы управления и обрабатывать события от них. В этом случае нужно при инициализации модуля добавить элементы управления:

```

var

    Act: ICADAction;

begin

    // ... то, что тут и было раньше
    // Создаём IndorCAD-объект «Событие», назначаем ему константу
    // MyFirstButtonID (она позже будет приходить при вызове
    // функций OnCreate и OnUpdate в параметре UserData)
    Act:=aApp.CreateAction(Creator, MyFirstButtonID);
    // Задаём ему всплывающую подсказку
    Act.Hint:='Создание объектов моего модуля';
    // Создаём объект управления – кнопку на панели инструментов,
    // на основе IndorCAD-объекта «Событие»
    aApp.CreateButton(GetModuleDescription, Act, ButtonStyleCheck,
0);
end;

```

Кроме того, чтобы всё это заработало, необходимо создать объект, отвечающий на события.

```
type
  TResponderObject = class(TInterfacedObject,
    ICADActionResponder)
private
  FApplication: IApplication;

protected // ICADActionResponder
  procedure OnUpdate(
    const Action: ICADAction;
    UserData: Integer
  ); safecall;
  procedure OnExecute(
    const Action: ICADAction;
    UserData: Integer
  ); safecall;
public
  constructor Create(const Appy: IApplication);
end;

procedure TResponderObject.OnUpdate(
    const Action: ICADAction;
    UserData: Integer);

var
  Cmd: ICADCommand;
  ActiveRoute: IRoute;
begin
  try
    ActiveRoute:=FApplication.CurrentDocument.Routes.ActiveRoute;
    Cmd := FApplication.CurrentDocument.Command;
  except
    // при желании исключение можно и запротоколировать.
    Action.Enabled := False;
    Action.Checked := False;
    exit;
  end;

case UserData of
  MyFirstButtonID: begin
    // устанавливаем активность кнопки
    Action.Enabled := ActiveRoute <> nil;
    // делаем кнопку нажатой, только если текущий режим -
    // именно тот, который должен быть связан с кнопкой.
```

```

        Action.Checked := (Action.Enabled) and (Cmd<>nil) and
            (TObject(Cmd.Handle) is TMyModeHandler);
    end;
    // Аналогично обрабатываются остальные кнопки
end;
end

procedure TResponderObjectCreator.OnExecute(
        const Action: ICADAction; UserData: Integer);
var
    CurrentRoute: IRoute;
begin
    try
        CurrentRoute :=
FApplication.CurrentDocument.Routes.ActiveRoute;
    except
        exit;
    end;
    case UserData of
        MyFirstButtonID:
            begin
                Action.Checked:=not Action.Checked;

                // Если кнопка была активирована, включить режим,
                // связанный с нею. В противном случае выключить.
                if Action.Checked then
                    FApplication.CurrentDocument.SetCommand(
                        0, TMyModeHandler.Create(CurrentRoute))
                else
                    FApplication.CurrentDocument.SetCommand(0, nil);
                end;
            end;
end;

constructor TResponderObject.Create(const Appy: IApplication);
begin
    FApplication := Appy;
end;

```

4. Подсистема проектирования инженерных коммуникаций

Подсистема предназначена для решения задач, связанных с отображением и проектированием инженерных коммуникаций. Она позволяет:

- отображать коммуникации на плане, в продольных и поперечных профилях проектируемых объектов;
- отображать коммуникации в 3D-виде местности;
- формировать выходные формы отчетности в требуемом ГОСТами виде.

Подсистема выполнена в рамках данной дипломной работы. Изначально она была реализована в виде модуля расширения для САПР IndorCAD/Road, однако на настоящий момент она встроена в ядро САПР и используется наравне с другими подсистемами.

Перед разработкой подсистемы автор с головой погрузился в предметную область: принимал участие в проектно-изыскательских работах, лично участвовал в процессе проектирования объектов. После этого объекты предметной области были расклассифицированы, и была разработана первая версия объектной модели. Затем эта модель была усовершенствована, а также постепенно были добавлены новые алгоритмы и возможности в соответствии с пожеланиями заказчиков.

В данной главе рассматривается структура подсистемы, эволюция объектной модели, в соответствии с которой она разрабатывалась, а также анализируются некоторые алгоритмы, нашедшие применение в подсистеме. Также рассматриваются приёмы обобщённого программирования, благодаря которым стало возможно реализовать части подсистемы, отвечающие за отрисовку коммуникаций в различных проекциях и за экспорт коммуникаций в профили, элегантно, оставив возможности дальнейшего расширения в случае необходимости.

4.1. Структура подсистемы

Подсистема выполнена в среде Borland Delphi, представляет собой набор модулей, в которых содержатся классы объектов коммуникационной сети, а также модуль с описаниями интерфейсов на языке IDL. Вся иерархия объектов коммуникационной сети отражена в иерархии интерфейсов, а классы, содержащиеся в модулях подсистемы, реализуют интерфейсы и предоставляют возможность создавать в подсистеме объекты, соответствующие объектам коммуникационной сети.

Рассмотрим иерархию интерфейсов и классов. Их можно условно разделить на несколько групп:

Первая группа содержит интерфейсы объектов предметной области: узлов коммуникаций и самих коммуникаций. В группе присутствуют:

- интерфейсы, описывающие свойства и методы, общие для всех коммуникаций (например, интерфейс `ICommunication` содержит функциональность для получения ссылок на все узлы коммуникации, для изменения параметров отображения (цвета, коэффициента масштабного преобразования) и прочих свойств);
- интерфейсы, соответствующие коммуникациям определённых общих типов (`IPipeline` – трубопровод, `ICable` – кабель);
- интерфейсы, соответствующие конкретным типам коммуникаций (`IRainPipe` – трубопровод ливневой канализации, `ITelecommunicationCable` – кабель связи, и множество других).

В настоящее время иерархия этих интерфейсов сильно разветвлена, так как зачастую функциональность, характерную для нескольких типов коммуникаций, удобно вынести в базовые интерфейсы и унаследовать от них интерфейсы объектов предметной области. Поэтому диаграмму интерфейсов удобно разбить на части и рассмотреть каждую отдельно. С верхним уровнем иерархии можно ознакомиться на рисунке 4:

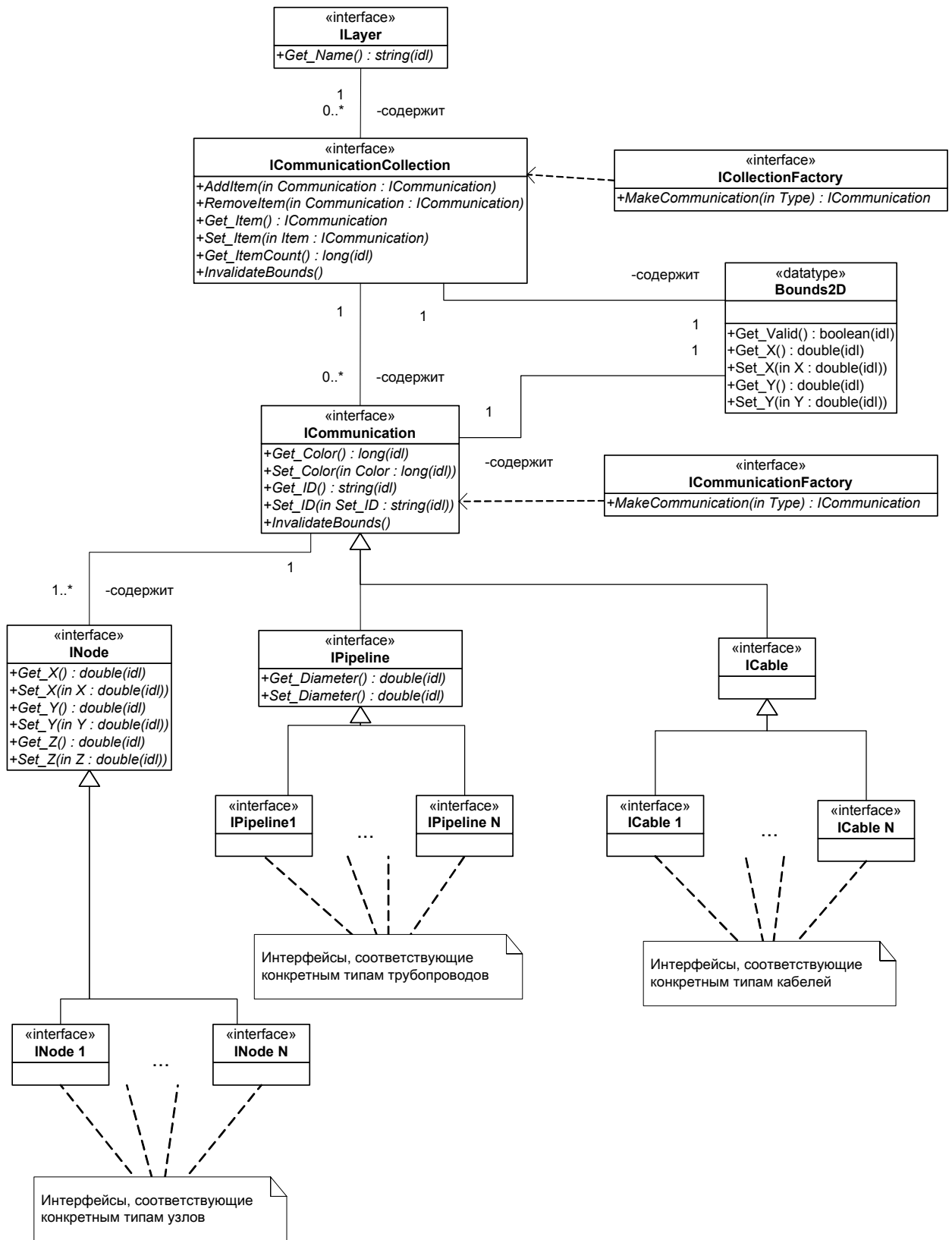


Рис. 4. Верхний уровень иерархии интерфейсов объектов коммуникационной сети

Здесь ILayer – это интерфейс САПР IndorCAD, дающий доступ к отдельному слою проекта. В каждом слое хранятся объекты-коллекции коммуникаций

(*ICollection*). Они, в свою очередь, хранят сами коммуникации (*Communication*).

На рисунке 5 представлена иерархия интерфейсов трубопроводов, кроме того, описаны вспомогательные типы данных, используемые интерфейсами:

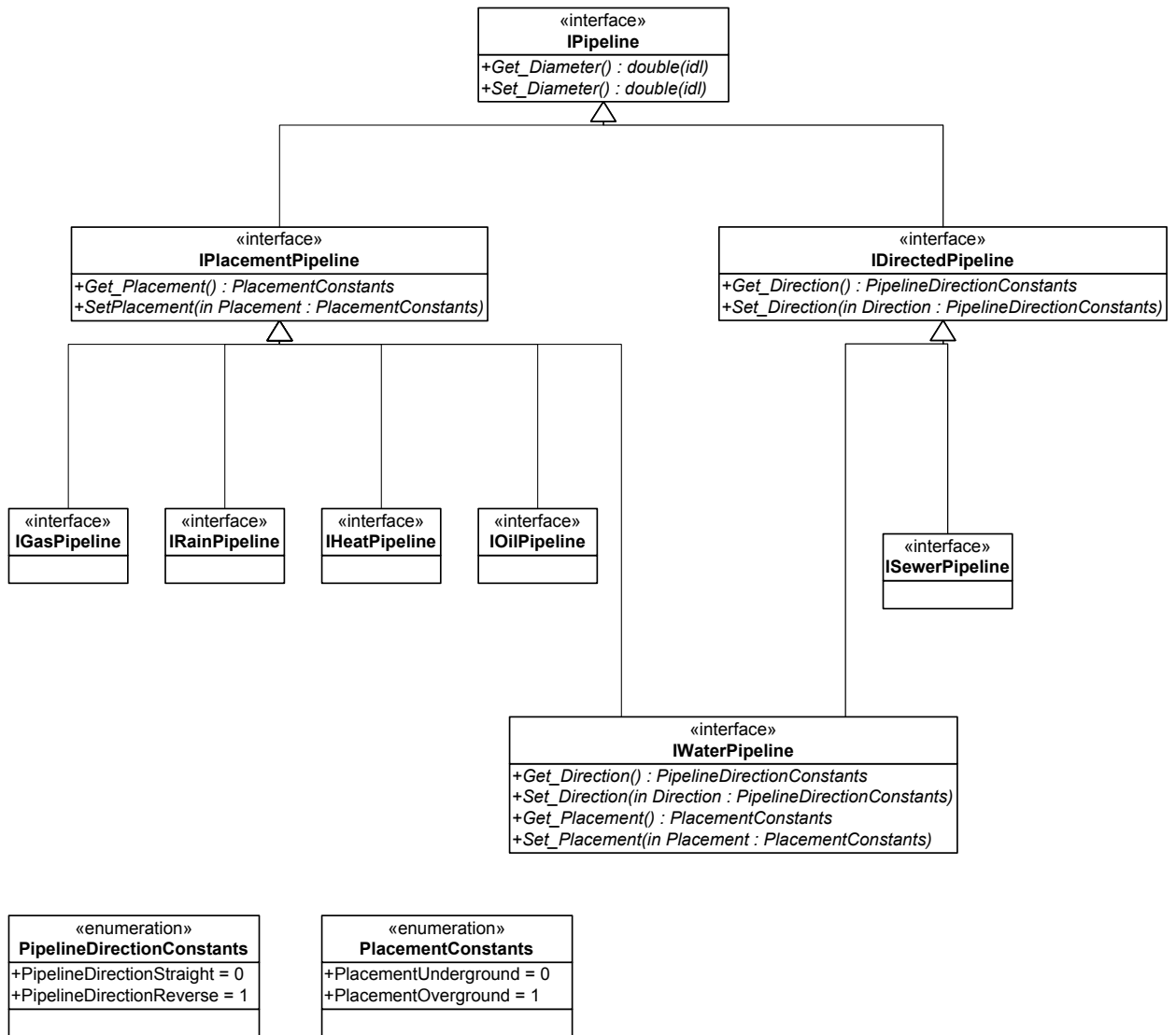


Рис. 5. Иерархия интерфейсов трубопроводов

На рисунке 6 представлена иерархия интерфейсов кабелей:

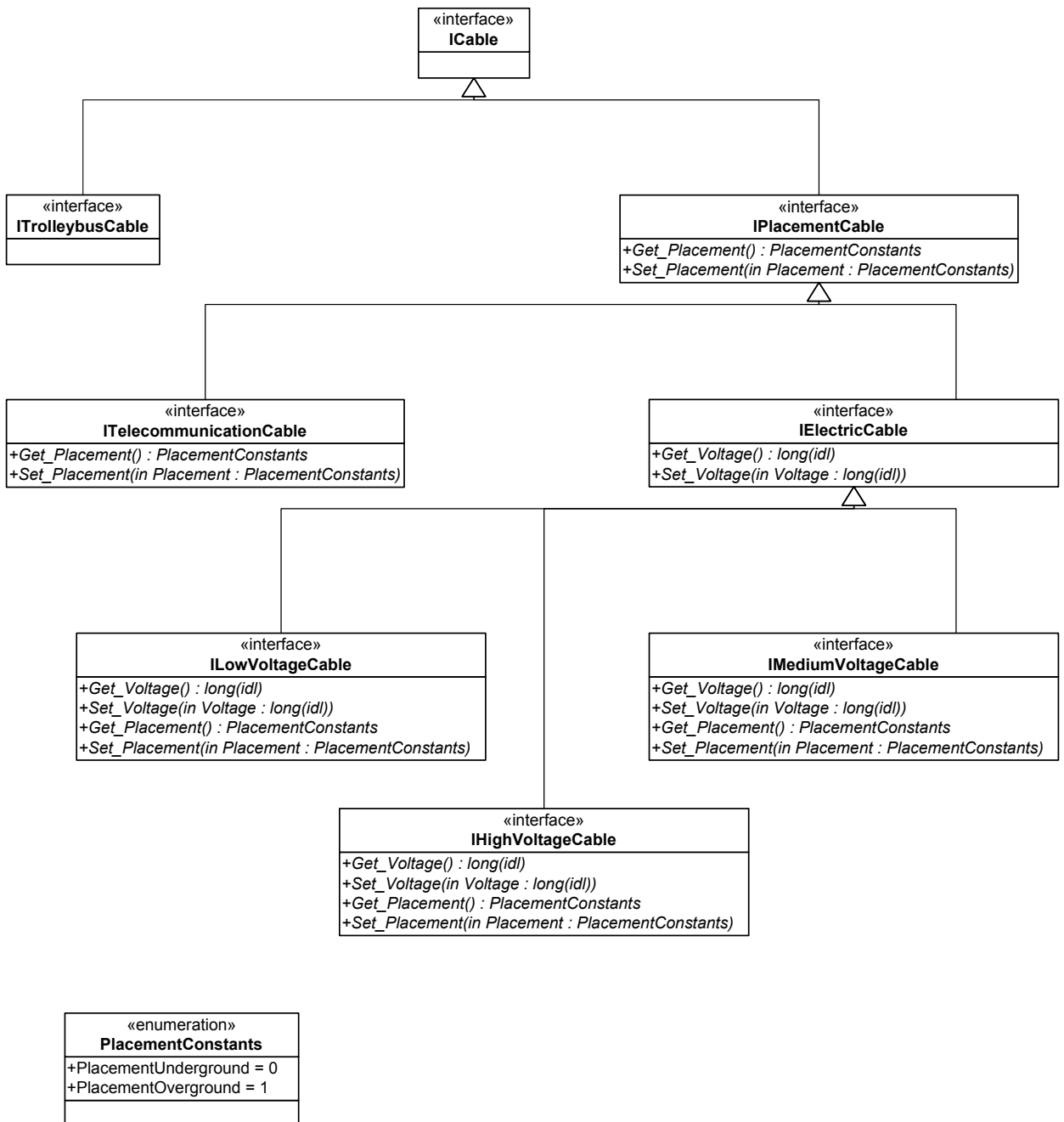


Рис. 6. Иерархия интерфейсов кабелей

И, наконец, на рисунке 7 представлена иерархия интерфейсов узлов коммуникационной сети:

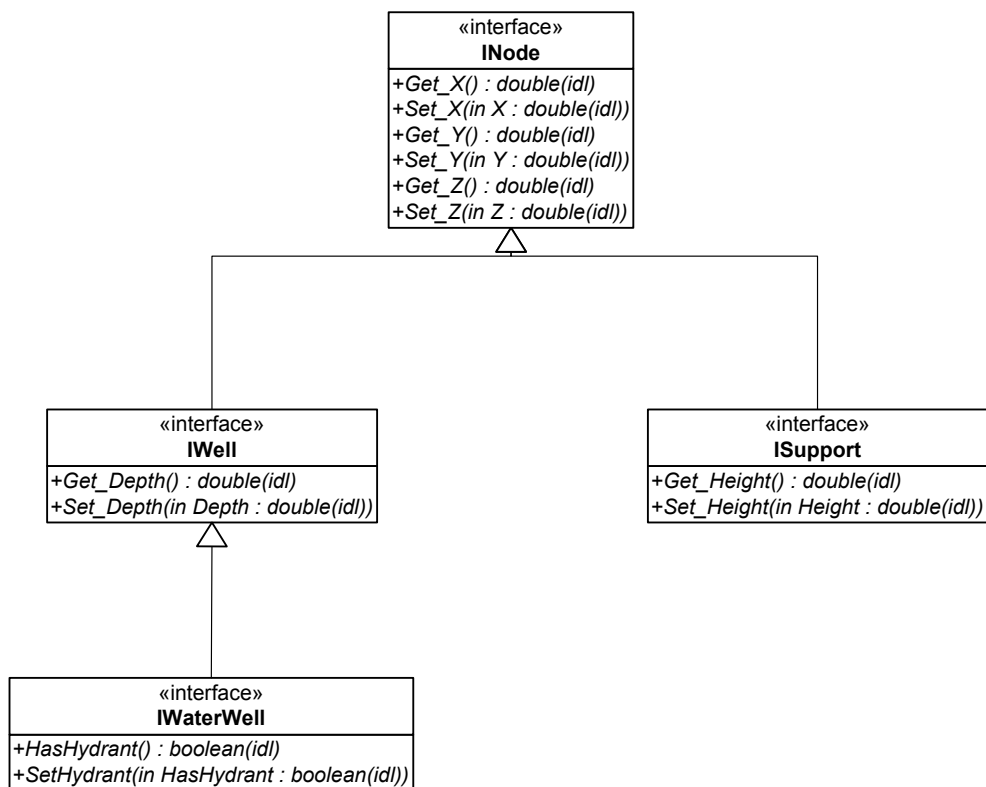


Рис. 7. Иерархия интерфейсов узлов коммуникационной сети

Вторая группа содержит классы-коллекции, объекты которых содержат ссылки на объекты, реализующие интерфейсы предыдущей группы. Интерфейсы этой группы предназначены для обеспечения связи между интерфейсами предметной области и интерфейсами САПР IndorCAD. Классы, реализующие интерфейсы коллекций, принимают запросы от САПР IndorCAD и перенаправляют их классам, реализующим интерфейсы объектов предметной области. Иерархия интерфейсов данной группы не так сильно разветвлена и в большей степени соответствует естественной классификации предметной области: Коммуникации → Трубопроводы, Кабели; Трубопроводы → Газопроводы, Нефтепроводы, Водопроводы, Теплотрассы и другие; Кабели → Тrolleyбусные кабели, Кабели связи и Кабели электропередач.

Иерархия интерфейсов представлена на рисунке 8:

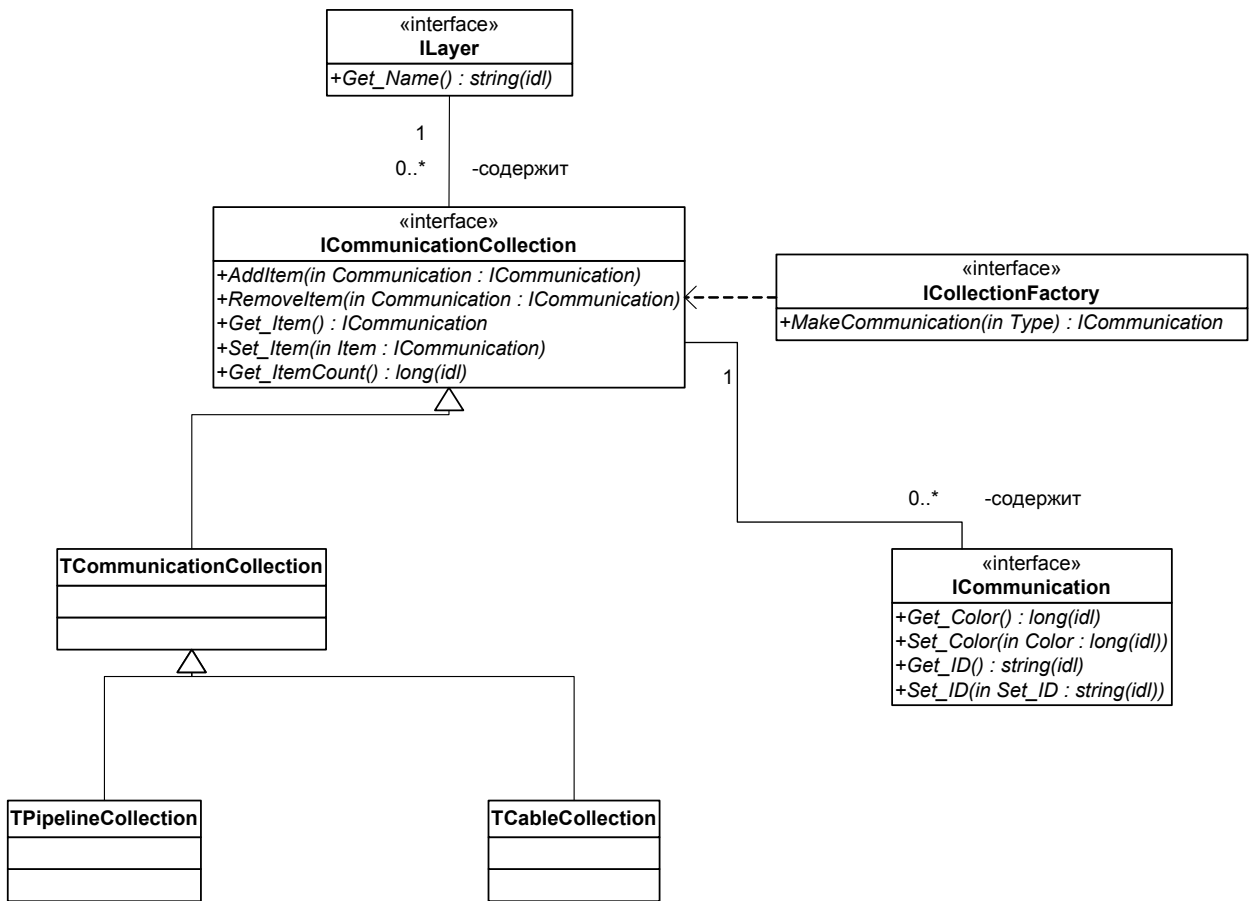


Рис 8. Иерархия коллекций объектов дорожной разметки

На рисунках 9-10 представлен нижний уровень иерархии классов-коллекций:

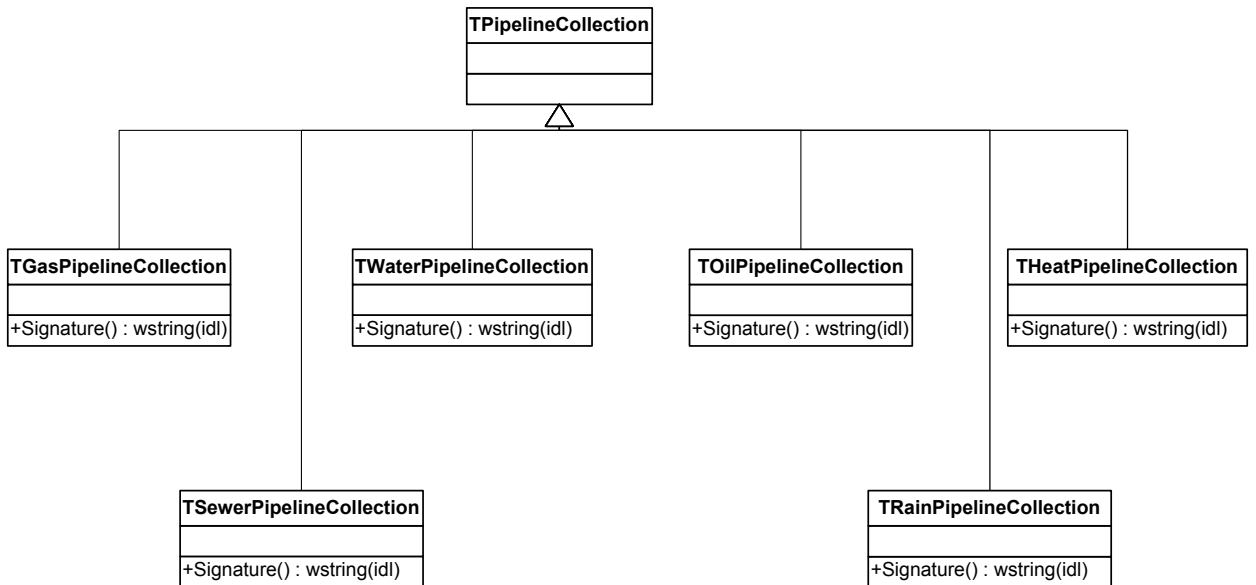


Рис. 9. Иерархия классов-коллекций трубопроводов

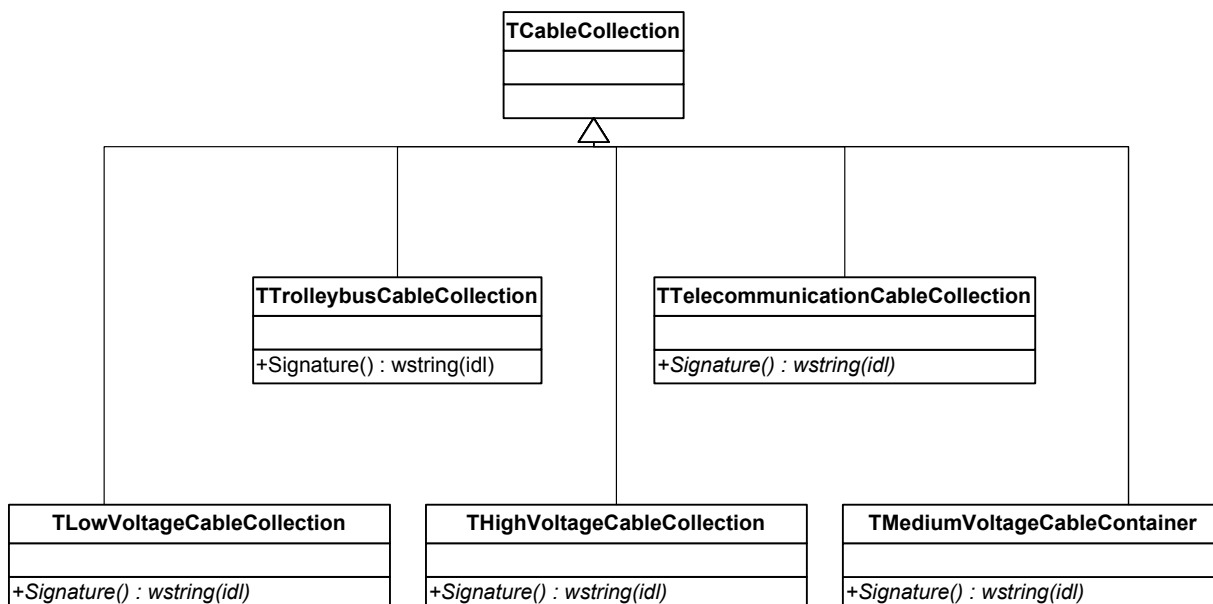


Рис. 10. Иерархия коллекций кабелей

К третьей группе относятся классы, предназначенные для создания удобного пользовательского интерфейса, а также классы форм. В эту группу входят классы-обработчики режимов, интерфейс класса-списка выделенных объектов, классы-обработчики событий от базовых элементов управления (в данной подсистеме используются только обработчики кнопок на панели инструментов), а также классы форм. Классы-обработчики режимов реализуют режимы создания и редактирования инженерных коммуникаций. Когда пользователь САПР включает режим, его обработчик начинает уведомляться обо всех происходящих событиях пользовательского ввода: нажатиях клавиш на клавиатуре, перемещениях мыши и прочих, и может предпринимать соответствующие действия. В списке выделенных объектов хранятся ссылки на выделенные в текущий момент коммуникации и/или их узлы.

Четвёртая группа – disp-интерфейсы к объектам коммуникационной сети. Эти интерфейсы предназначены для обеспечения управления по технологии ActiveX объектами подсистемы извне: из других программ, поддерживающих ActiveX-технологии, из сценариев. Классы данной группы соответствуют естественной иерархии классов предметной области и являются прослойкой между, с одной стороны, сценариями и высокоуровневыми программами, и, с другой стороны, классами первой и второй групп.

Все рассмотренные выше классы и интерфейсы образуют целостную расширяемую структуру. При желании обеспечить дополнительную функциональность, поддержка которой имеется в ядре САПР, но отсутствует в подсистеме, достаточно лишь описать

реализацию необходимого интерфейса в классе-коллекции и классах объектов коммуникационной сети.

4.2 Реализованная функциональность

Рассмотрим, как могут использоваться объекты, хранимые в подсистеме. Различные варианты их использования вводятся путём реализации COM-интерфейсов САПР IndorCAD объектами:

- Хранение коммуникаций в цифровой модели местности – интерфейс ICADObject;
- Отображение классификации коммуникаций в дереве объектов – интерфейс ICADTreeObject;
- Загрузка/сохранение коммуникаций из потока/в поток – интерфейс IPropertyStreamed;
- Рисовка в плане проектируемых коммуникаций – интерфейс ICADObjectDraw;
- Произвольная поточечная трансформация коммуникаций – интерфейс ITransformableObject;
- Автоматический поиск режима при щелчке по объекту – интерфейс ISearchableObject;
- Прилипание курсора мыши к коммуникации или узлу, когда они находятся на близком расстоянии – интерфейс ISnappingObject;
- Хранение минимального объемлющего прямоугольника для автоматического отсечения невидимых объектов ядром – интерфейс IBoundedObject;
- Импорт коммуникаций в один слой из другого (в том числе между разными проектами) – интерфейс ICADAssignableWithLayerObject;
- Рисование коммуникаций в произвольных разрезах (профилях) – интерфейс IProfileDrawObject;
- Экспорт плана расположения коммуникаций (в контексте общего плана местности) в чертёж для последующей подготовки к печати – интерфейс IExportableToPlan;

- Экспорт коммуникаций в чертежи разрезов, построенных по произвольным ломаным – интерфейс IExportableWithProfile;
- Управляемость из сценариев и высокоуровневых программ, поддерживающих технологию ActiveX – интерфейс ICADCOMObject;
- Отображение в трёхмерном виде местности – интерфейс IObject3D;

4.3 Применение обобщённых алгоритмов

В современных императивных языках создание алгоритмов, работающих с произвольными типами входных данных, в известной степени затруднено. Это связано с тем, что в основной массе они строго типизированы, и при создании алгоритмов требуется явно указывать типы входных данных. Даже особо развитые языки (C++ стандарта 1998 года, Ada), обладающие средствами облегчения этой проблемы, (в C++ - шаблоны (templates), в Ada – обобщённые компоненты (generics)), накладывают серьёзные ограничения на использование этих средств. Так, в большинстве реализаций C++ (от компаний Visual C от Microsoft, C++ Compiler от Intel, CodeWarrior от Metrowerks, GCC от Free Software Foundation) невозможно экспортировать шаблоны для использования между скомпилированными приложениями. Это крайне затрудняет использование системой подключаемых модулей.

В то же время, результаты, достигаемые при использовании языков с автоматическим вычислением типов аргументов на основе применяемых функций (Haskell [4], ML [6]), а также при разработке приложений на C++ в рамках ограничений, налагаемых системой шаблонов, крайне интересны. Алгоритмы на таких языках можно записывать без указания типов аргументов. Типы вычисляются автоматически либо исходя из того, какие операции используются над объектами, либо на основе того, какие данные поступают на вход.

В результате получается, что алгоритмы не нужно переписывать по много раз для разных типов аргументов, достаточно описать один раз, и система программирования сама вычислит необходимые типы. Естественно, этого результата хотелось бы добиться и в языках, не обладающих столь мощными возможностями.

Автором реализована такая возможность на примере алгоритма вычисления координат пересечения коммуникаций с различными разрезами, профилями. Он выглядит следующим образом:

- Перечисляются все коммуникации, производится отсечение по минимальным объемлющим прямоугольникам (MBR) каждой коммуникации и отрезка профиля, который будет отображён в чертеже
- У коммуникаций, которые не были отсечены, ищутся точки пересечения с профилем. Это происходит так. Известны точки X_0 , Y_0 на оси трассы – текущее положение, на котором строится поперечный профиль. Также известен угол α , под которым трасса идёт в этой точке. Кроме того, известны расстояния от оси до кромок проектируемой трассы в нашей точке: до левой кромки – $MinOffs$, до правой кромки - $MaxOffs$. По формулам преобразования координат при повороте системы координат на угол α рассчитываются границы поперечного профиля – (X_1, Y_1) и (X_2, Y_2) . Эти две точки дают нам отрезок, с которым надо искать пересечение коммуникаций. Далее идёт цикл по всем отрезкам коммуникации (коммуникация – линейно-протяжённый объект, и состоит из отрезков), в котором проверяется, пересекли ли проекции на плоскость xOy отрезка коммуникации и отрезка профиля друг друга. Если пересекли, то рассчитывается координата Z точки пересечения. Этот расчёт нетривиален в связи с тем, что информация о расположении коммуникации почти всегда неполна: глубины колодцев/высоты опор обычно известны только в ограниченном множестве точек, и для восстановления утраченных значений глубины приходится применять интерполяцию.
- Когда о точке пересечения известна вся пространственная информация (координаты X, Y, Z), необходимо выполнить требуемое действие.

Сложность заключается в том, что алгоритм поиска пересечений – один, а действий (которые выполняются в последнем пункте) требуется выполнять несколько. Кроме того, для каждого из этих действий необходимы разные исходные данные. Представить всю сложность обстановки несложно на паре примеров. Требуется, допустим, узнать диапазон изменения значений координат на каком-либо поперечном профиле. Для этого в САПР IndorCAD/Road есть стандартный интерфейс `IExportableWithProfile`, поддерживая который, контейнер объектов модуля обязан реализовать метод `GetTransBounds([in] int PkIdx, [out] double MinOffs, [out] double MinZ, [out] double MaxOffs, [out] double MaxZ)`, на вход которого поступает пикет интересующей точки на трассе (такова дорожная терминология: пикет – это расстояние по трассе от её начала до точки). Реализация этого

метода в контейнере должна вызвать его у всех объектов (коммуникаций), которые он содержит, после чего рассчитать по результатам этих вызовов минимальный объемлющий прямоугольник – и вернуть его через выходные параметры. Наш обобщённый алгоритм должен вызываться уже из реализации метода `GetTransBounds` в самом объекте. Но что же он должен сделать, найдя точку пересечения? Здесь – заполнить выходные параметры и вернуть управление. Допустим. Теперь нам, предположим, необходимо экспортироваться в чертёж. Метод уже совсем другой – `ExportTransverseProfilePage`, которому на вход передаётся страница. И есть ещё много других задач, которые необходимо решать, применяя этот алгоритм. Можно было бы отвести место под все возможные исходные данные, но этот подход неочевиден. В таких случаях говорят: «Программа – write-only». И это высказывание было бы совершенно справедливо, потому что читать программу действительно было бы невозможно. А это сильно затрудняет сопровождение и развитие.

Выход есть. Это функциональные объекты, или функторы [2] – объекты, имеющие семантику вызова (для облегчения понимания можно представлять эти объекты как аналог функций обратного вызова (callback) в объектно-ориентированном проектировании). Для обеспечения этой семантики необходим метод, который можно было бы универсальным образом вызывать. А классы, реализующие этот метод, для каждого действия должны создаваться свои. Назовём этот метод, к примеру, `Exec`. В нашем случае параметрами его будут координаты точки пересечения (удобно сразу задавать координаты профиля – смещение и высоту), и сама коммуникация, над которой необходимо выполнить действие. А все внешние данные, необходимые для выполнения действия можно передавать: входные – через конструктор функтора, выходные – через другие методы.

5. Подсистема проектирования дорожной разметки

Подсистема предназначена для решения задач, связанных с отображением и проектированием дислокации дорожной разметки. Она позволяет:

- отображать дорожную разметку на плане местности проектируемых объектов;
- формировать выходные формы отчётности в требуемом ГОСТами виде.

Подсистема выполнена в рамках данной дипломной работы и представляет собой модуль расширения для САПР IndorCAD/Road.

Перед разработкой автором была исследована предметная область, после чего объекты, требующие обработки, были расклассифицированы. Объектная модель строилась аналогично модели предыдущей подсистемы и повторно показала свою применимость. В настоящее время первая версия модуля готова к внедрению в САПР IndorCAD/Road.

5.1. Структура подсистемы

Подсистема выполнена в среде Borland Delphi и представляет собой набор модулей, в которых содержатся классы объектов дорожной разметки, а также модуль с описаниями интерфейсов на языке IDL. Вся иерархия объектов дорожной разметки отражена в иерархии интерфейсов, а классы, содержащиеся в модулях подсистемы, реализуют эти интерфейсы и предоставляют возможность создавать в подсистеме объекты, соответствующие объектам дорожной разметки.

Рассмотрим иерархию интерфейсов и классов. Их можно условно разделить на несколько групп:

Первая группа содержит интерфейсы объектов точечной, линейной, поперечной и площадной разметки. В группе присутствуют:

- интерфейсы, описывающие свойства и методы, общие для всех объектов разметки (например, интерфейс IMarkupElement содержит функциональность для изменения параметров отображения (кода разметки по ГОСТ'ам, цвета, коэффициента масштабного преобразования), и прочих свойств);
- интерфейсы, соответствующие разметке определённых общих типов (IPointMarkup – точечная разметка, ILinearMarkup – линейная разметка,

ITransversalMarkup – поперечная разметка, IArealMarkup – площадная разметка);

- интерфейсы, соответствующие конкретным типам разметки (IAxialLinearMarkup – осевая линейная разметка, IArbitraryLinearMarkup – линейная разметка, располагающаяся вдоль произвольной ломаной и другие).

Диаграмму интерфейсов удобно разбить на части и рассмотреть каждую отдельно. С верхним уровнем иерархии можно ознакомиться на рисунке 8:

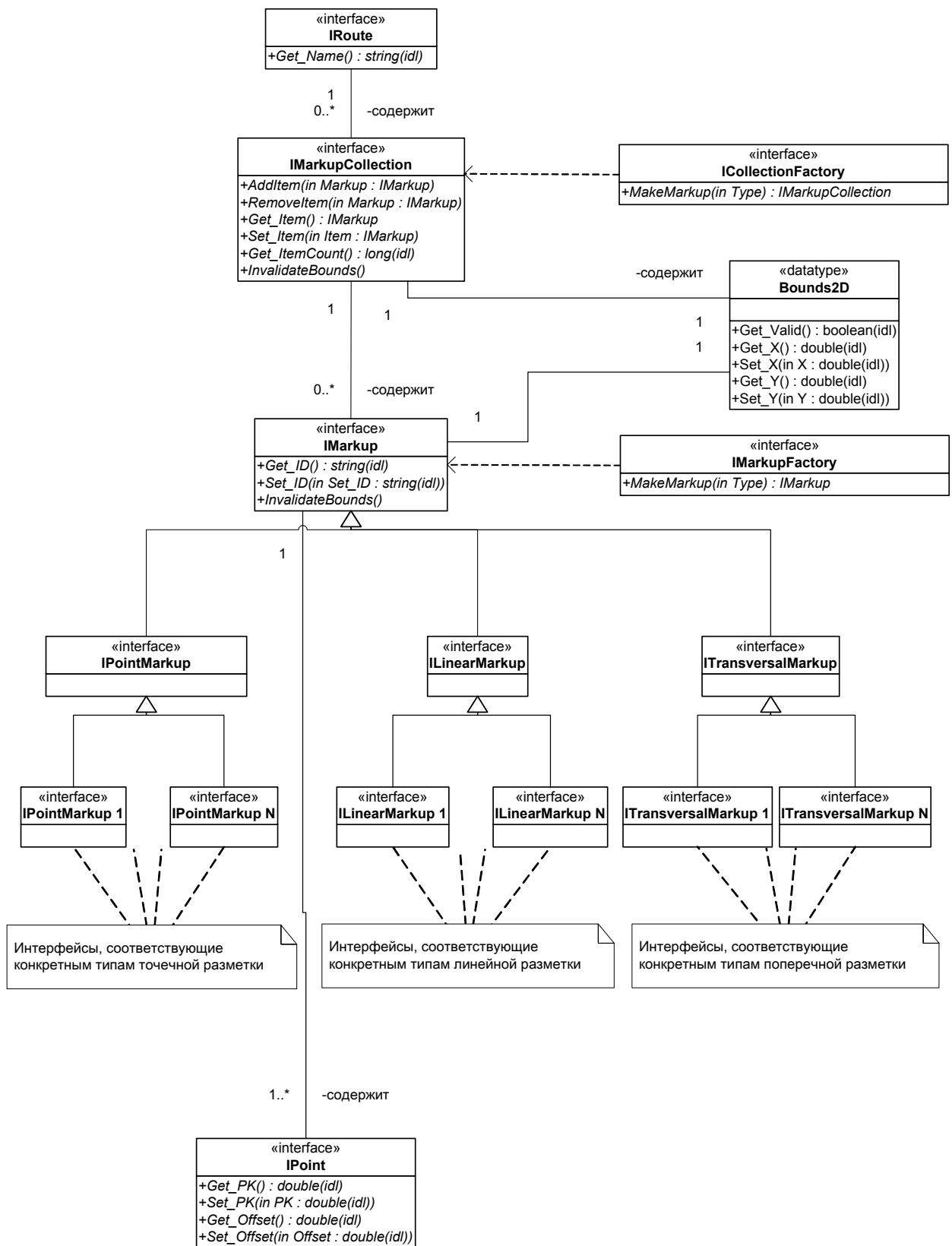


Рис. 11. Верхний уровень иерархии интерфейсов объектов дорожной разметки

Здесь IRoute – это интерфейс САПР IndorCAD, дающий доступ к трассе проектируемого объекта. С каждой трассой связана располагаемая на ней дорожная разметка. Физически – объекты-коллекции (IMarkupCollection) объектов дорожной

разметки хранятся в трассе как дочерние элементы, и, в свою очередь, хранят сами объекты дорожной разметки (IMarkupElement).

Рассмотрим иерархию объектов разметки глубже. Рисунок 9 предоставляет возможность ознакомиться с иерархией объектов точечной разметки, рисунок 10 – с иерархией объектов линейной разметки, а рисунок 11 – с иерархией объектов поперечной разметки.

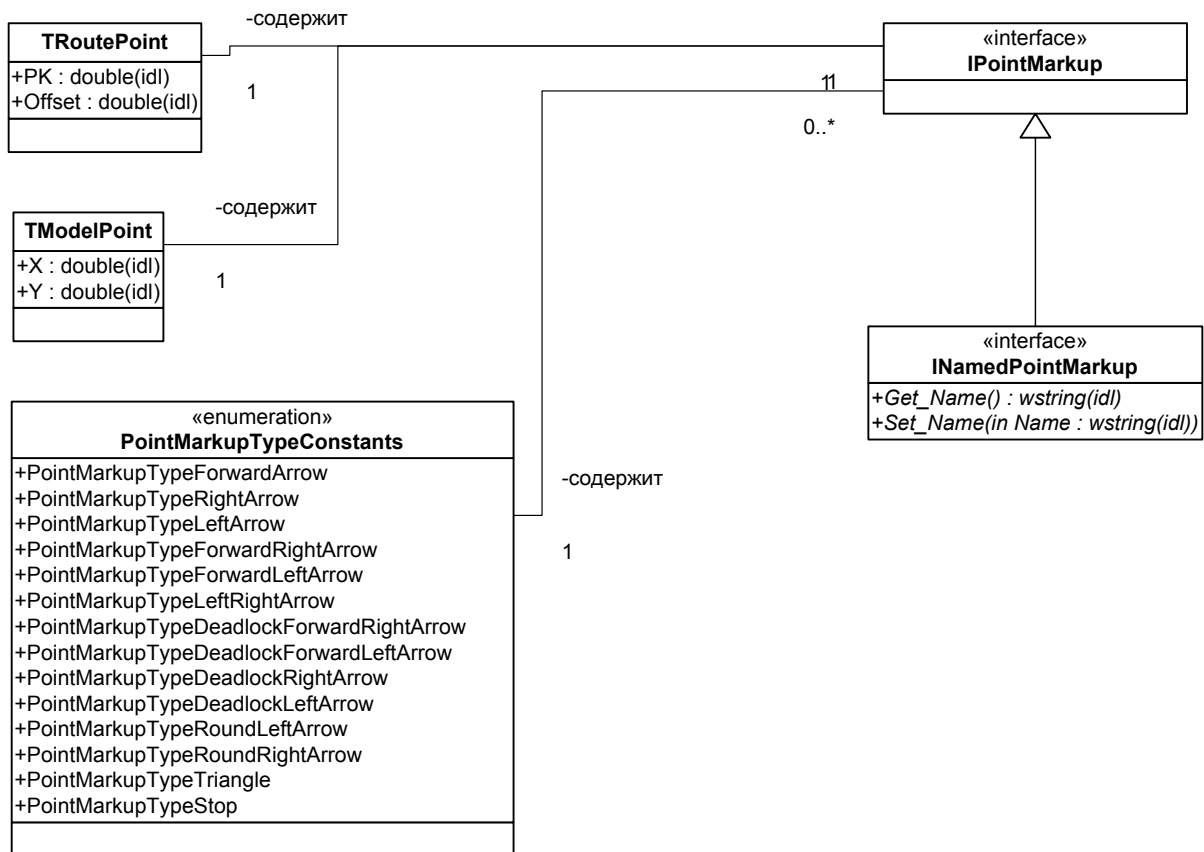


Рис. 12. Иерархия объектов точечной разметки

К точечной разметке (интерфейс по умолчанию - IPointMarkup) относятся стрелки, разметка, дублирующая дорожные знаки, обозначения автобусных остановок, а также различные подписи на дороге (которым соответствует интерфейс INamedPointMarkup).

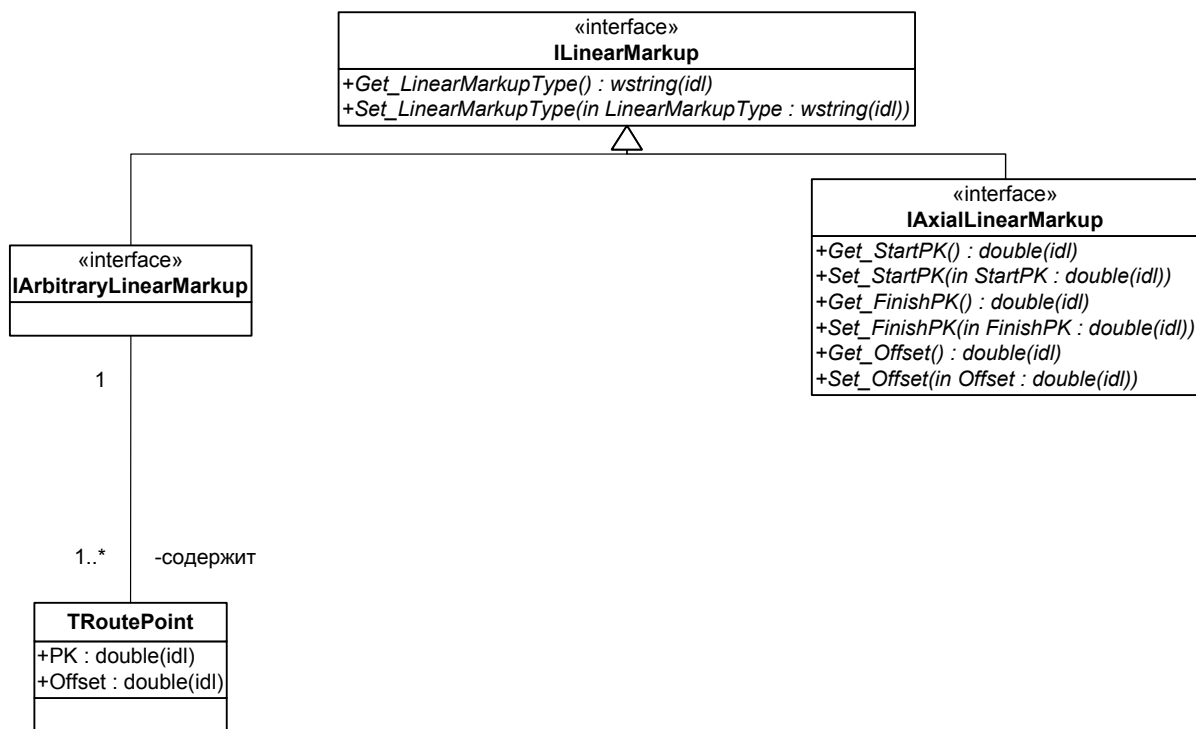


Рис. 13. Иерархия интерфейсов линейной разметки

К линейной разметке относится осевая разметка и разметка, разделяющая полосы движения. Линейная разметка в подсистеме может задаваться либо произвольно – набором точек (пикет, смещение), либо быть жёстко привязанной к элементу трассы (ось, левая или правая кромка) – в таком случае задаётся пикет начала разметки, пикет её окончания и смещение от элемента трассы.

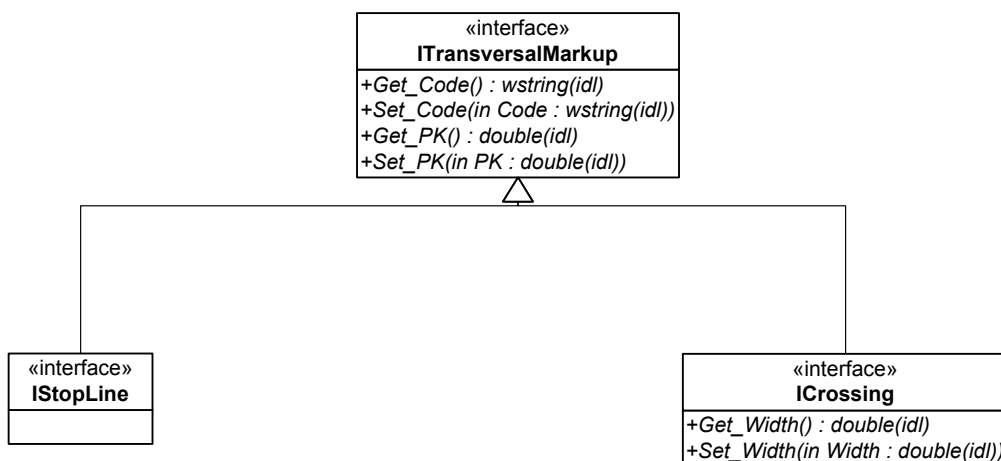


Рис. 14. Иерархия интерфейсов поперечной разметки

Вторая группа содержит классы-коллекции, объекты которых содержат ссылки на объекты, реализующие интерфейсы предыдущей группы. Интерфейсы этой группы предназначены для обеспечения связи между интерфейсами объектов дорожной разметки и интерфейсами САПР IndorCAD. Классы, реализующие интерфейсы коллекций, принимают запросы от САПР IndorCAD и перенаправляют их классам, реализующим интерфейсы объектов дорожной разметки.

К третьей группе относятся классы, предназначенные для создания удобного пользовательского интерфейса, а также классы форм. В эту группу входят классы-обработчики режимов, интерфейс класса-списка выделенных объектов, классы-обработчики событий от базовых элементов управления (в данной подсистеме используются только обработчики кнопок на панели инструментов), а также классы форм. Классы-обработчики режимов реализуют режимы создания и редактирования инженерных коммуникаций. Когда пользователь САПР включает режим, его обработчик начинает уведомляться обо всех происходящих событиях пользовательского ввода: нажатиях клавиш на клавиатуре, перемещениях мыши и прочих, и может предпринимать соответствующие действия. В списке выделенных объектов хранятся ссылки на выделенные в текущий момент коммуникации и/или их узлы.

Четвёртая группа – disp-интерфейсы к объектам коммуникационной сети. Эти интерфейсы предназначены для обеспечения управления по технологии ActiveX объектами подсистемы извне: из других программ, поддерживающих ActiveX-технологии, из сценариев. Классы данной группы соответствуют естественной иерархии классов предметной области и являются прослойкой между, с одной стороны, сценариями и высокоуровневыми программами, и, с другой стороны, классами первой и второй групп.

Все рассмотренные выше классы и интерфейсы образуют целостную расширяемую структуру. При желании обеспечить дополнительную функциональность, поддержка которой имеется в ядре САПР, но отсутствует в подсистеме, достаточно лишь описать реализацию необходимого интерфейса в классе-коллекции и классах объектов коммуникационной сети.

5.2 Реализованная функциональность

Рассмотрим, как могут использоваться объекты, хранимые в подсистеме. Различные варианты их использования вводятся путём реализации COM-интерфейсов САПР IndorCAD объектами:

- Хранение дорожной разметки в цифровой модели местности – интерфейс ICADObject;
- Отображение классификации дорожной разметки в дереве объектов – интерфейс ICADTreeObject;
- Загрузка/сохранение дорожной разметки из потока/в поток – интерфейс IPropertyStreamed;
- Рисовка в плане проектируемых дорожной разметки – интерфейс ICADObjectDraw;
- Произвольная поточечная трансформация дорожной разметки – интерфейс ITransformableObject;
- Автоматический поиск режима при щелчке по объекту – интерфейс ISearchableObject;
- Прилипание курсора мыши к коммуникации или узлу, когда они находятся на близком расстоянии – интерфейс ISnappingObject;
- Хранение минимального объемлющего прямоугольника для автоматического отсечения невидимых объектов ядром – интерфейс IBoundedObject;
- Импорт дорожной разметки в один слой из другого (в том числе между разными проектами) – интерфейс ICADAssignableWithLayerObject;
- Экспорт плана расположения дорожной разметки (в контексте общего плана местности) в чертёж для последующей подготовки к печати – интерфейс IExportableToPlan;

Заключение

По результатам данной дипломной работы можно сделать следующие основные выводы:

1. Автором исследовано состояние проблемы, проведён анализ существующих решений, а также указаны недостатки существующих систем автоматизированного проектирования инженерных коммуникаций и дислокации инженерного обустройства – как универсальных, так и узкоспециализированных;
2. Описана методика разработки подключаемых модулей для САПР IndorCAD;
3. Разработана технология проектирования объектов окружения автомобильных дорог, позволяющая значительно ускорить процесс проектирования различных объектов;
4. На основе разработанной технологии проектирования созданы две коммерческие подсистемы для САПР IndorCAD, разрабатываемой в ООО «Индорсофт. Инженерные сети и дороги»: подсистема проектирования инженерных коммуникаций и подсистема проектирования дислокации дорожной разметки. Подсистемы выполнены в среде Borland Delphi с использованием современных методов объектно-ориентированного проектирования, обобщённых алгоритмов, технологий COM и ActiveX;
5. Разработанные подсистемы внедрены в промышленную эксплуатацию в ведущие проектные организации Томска, Сибирского региона, России и СНГ: ООО «Индорсофт. Инженерные сети и дороги», ООО ИДЦ «Индор», ОАО «Томгипротранс», ГУП «АлтайИндорПроект», ОАО «КрасИндорПроект», ООО НППФ «Краснодаравтотдорсервис», ОАО «Каздорпроект»;
6. На основании полученных результатов опубликованы две печатные работы [12], [14].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Abelson H., Sussman G.J. Structure and Interpretation of Computer Programs. 2nd edition. Cambridge, MA: MIT Press, 1996.
2. Gamma E., Helm R., Johnson R. and Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. Reading, MA: Addison-Wesley, 1995.
3. Graham P. OnLisp, Prentice Hall, 1993.- 432 p.
4. Hudak P., Peterson J., Fasel J. A Gentle Introduction to Haskell., 1999. (<http://www.haskell.org/tutorial/>)
5. Keller R.M. Computer Science: Abstraction to Implementation, Harvey Mudd College, 2001.
6. Leroy X. The Objective Caml system release 3.07. Institut National de Recherche en Informatique et en Automatique, 2003. (<http://caml.inria.fr/ocaml/htmlman/index.html>)
7. Petzold C. Programming Windows. Microsoft Press, 1998.
8. Rogerson D. Inside COM: Microsoft's Component Object Model. Microsoft Press, 1997.
9. Richter J. Programming Applications for Microsoft Windows. 4th edition. Microsoft Press, 1999.
10. Utility GIS – More than Just AM/FM. – N.Y.: ESRI Press, 2003. – 13 p.
11. Гончаренко С.В., Гуральник М.Л. САПР-подход к инженерным коммуникациям // Инженерные коммуникации и геоинформационные системы: Материалы первого учебно-практического семинара, «ГИС–Ассоциация», 14–17 октября 1997 г. – М.: 1997, с. 3–9.
12. Лубкин А.М., Полякова Н.А., Снежко В.В. Разработка геоинформационной системы дорожной отрасли. // Материалы XLII Международной студенческой конференции "Студент и научно-технический прогресс". - Информационные технологии. - 2004. - С. 16-17
13. Слюсаренко С.Г., Рожков В.П., Субботин С.А. и др. Современные информационные технологии в эксплуатации инженерных сетей // Труды международной научно-практической конференции «Геоинформатика-2000» 15-18 сентября 2000 г. – Томск: Изд-во Томского государственного университета, 2000, с. 219–224
14. Снежко В.В., Петренко Д.А. Разработка модуля инженерных коммуникаций в САПР. // Материалы XLII Международной студенческой конференции "Студент и научно-технический прогресс". - Информационные технологии. - 2004. - С. 8-9.
15. Чаусова Е.В. - IndorCAD/Road. Руководство пользователя. (<http://www.indorsoft.ru/Articles/Supporting/Help/IndorCAD>)

ПРИЛОЖЕНИЕ 1. Подсистема проектирования инженерных коммуникаций. Руководство пользователя.



Подсистема проектирования инженерных коммуникаций предназначена для решения задач, связанных с отображением и проектированием инженерных коммуникаций. Она позволяет:

- отображать коммуникации на плане, в продольных и поперечных профилях проектируемых объектов;
- формировать выходные формы отчетности в требуемом ГОСТами виде.

Подсистема не требует отдельной установки, так как встроена в ядро САПР IndorCAD/Road и является частью основного исполняемого модуля в том случае, если она приобретена. В противном случае функции отображения и проектирования коммуникаций недоступны.

П 1.1. Начало работы

Если Вы приобрели САПР IndorCAD/Road со встроенной подсистемой проектирования коммуникаций, то после создания или открытия проекта в САПР на дополнительной панели инструментов главного окна появятся кнопки для работы с коммуникациями:

-  **Создать новую коммуникацию.** Кнопка включает режим создания коммуникаций.
-  **Выделить коммуникацию, задать свойства.** Включает режим выделения и редактирования коммуникаций.

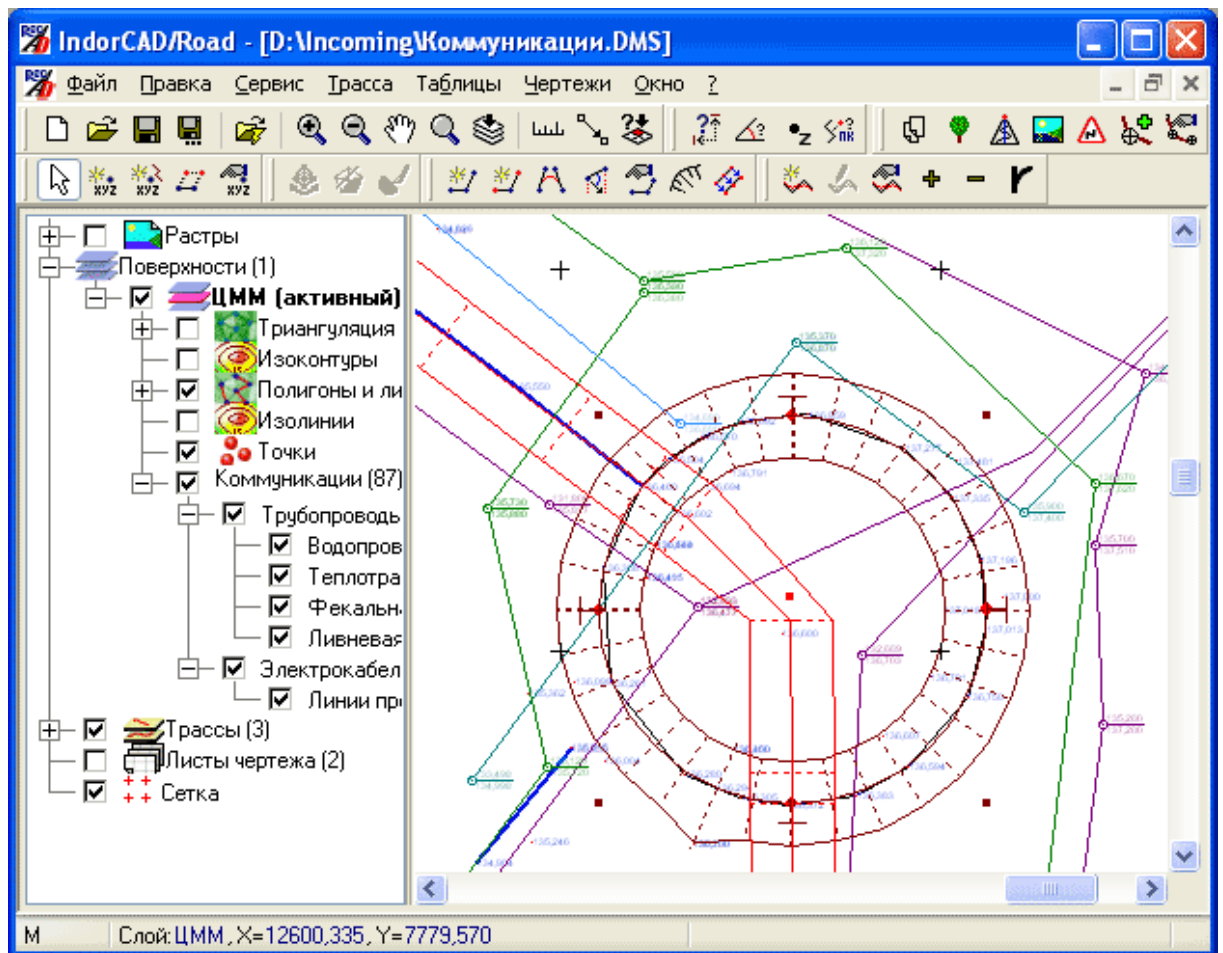


Рис. 15. Главное окно системы IndorCAD/Road.

II 1.2. Создание и удаление коммуникаций

Коммуникации создаются в виде самостоятельных объектов. При этом в дереве объектов проекта появляется новый объект «Коммуникации», видимость которого можно отключить в случае необходимости.

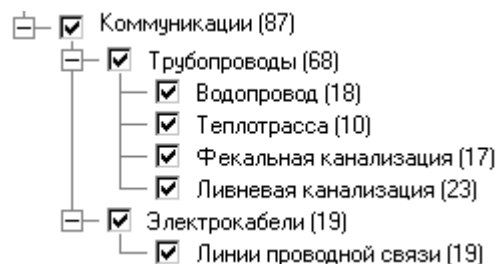



Рис. 16. Фрагмент дерева объектов проекта с коммуникациями

Чтобы создать линию коммуникации, включите режим создания коммуникаций. Для этого щелкните на кнопке **Создать новую коммуникацию**, расположенной на дополнительной панели инструментов. Последовательными щелчками мыши обозначьте узлы, по которым будет построена линия коммуникации. Чтобы исключить последний узел из линии, воспользуйтесь клавишей **ВкSp**. Чтобы завершить построение линии,

поместите курсор на последний узел линии (курсor примет вид ) и щелкните левой кнопкой мыши. Для отмены построения нажмите клавишу Esc.

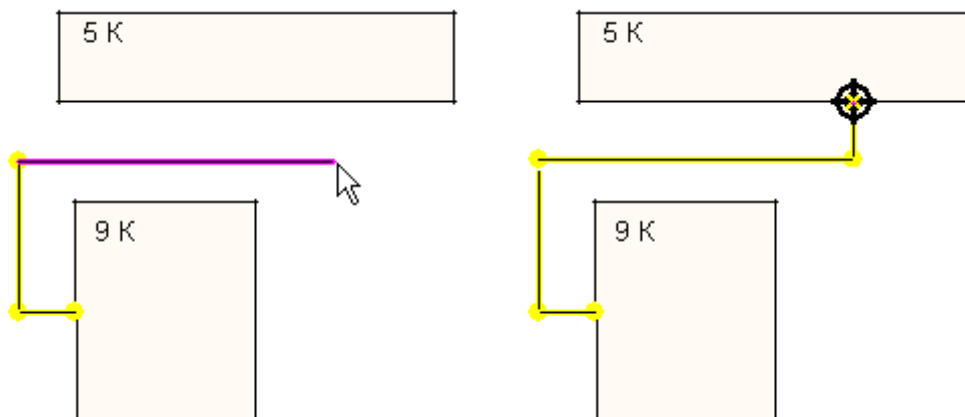


Рис. 17. Создание коммуникации

При создании линии коммуникации удобно использовать контекстное меню, которое открывается щелчком правой кнопки мыши.

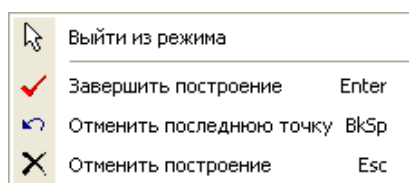






Рис. 18. Контекстное меню

Контекстное меню содержит следующие команды:

-  **Выйти из режима.** Отключает режим создания линий коммуникаций.
-  **Завершить построение.** Завершает построение линии.
-  **Отменить последнюю точку.** Исключает последний узел из линии.
-  **Отменить построение.** Отменяет построение линии.

После завершения построения линии откроется диалоговое окно для выбора типа коммуникации.

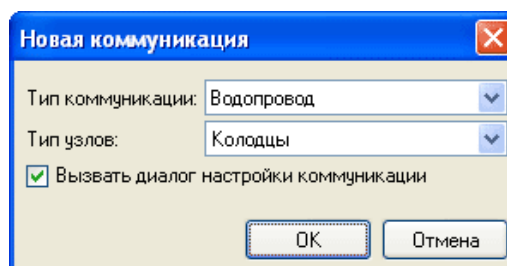


Рис. 19. Выбор типа коммуникации

- **Тип коммуникации.** Выбирается из раскрывающегося списка. Список содержит две группы: **Трубопроводы** и **Электрокабели** и ЛЭП, которые

объединяют соответствующие типы коммуникаций. Выбор подходящего типа осуществляется щелчком левой кнопкой мыши по его названию в списке.

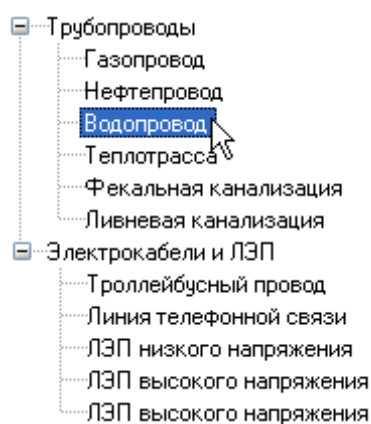


Рис. 20. Типы коммуникаций

- **Тип узлов.** В узлах линии коммуникации можно разместить колодцы или опоры. Для этого достаточно выбрать соответствующий тип узла: **Колодцы**, **Опоры** или значение **Нет**, чтобы отказаться от специальных обозначений узлов.

После нажатия на кнопку **ОК** на активной поверхности появится новая линия коммуникации.

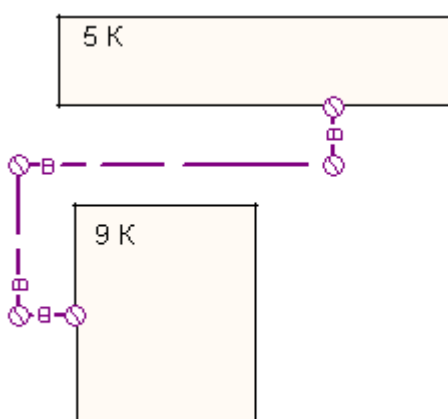


Рис. 21. Нанесение водопровода с колодцами в узлах

Стиль и цвет линии коммуникации определяются из общих настроек, установленных для каждого типа коммуникации (см. следующий раздел). Чтобы задать индивидуальные настройки линии, в окне диалога **Новая коммуникация** выберите опцию **Вызвать диалог настройки коммуникации**. Тогда после нажатия на кнопку **ОК** откроется диалоговое окно для настройки параметров коммуникации (см. раздел «Параметры коммуникаций»). Установленные параметры будут применяться по умолчанию для всех вновь созданных коммуникаций этого типа.

Удаление коммуникаций

Для удаления коммуникации включите режим выделения и редактирования коммуникаций, щелкните на линии правой кнопкой мыши, выполните команду **Удалить** из контекстного меню и дайте положительный ответ на запрос системы. Еще один способ удаления коммуникации: выделить линию коммуникации и выполнить команду **Удалить** в меню **Правка** или нажать на клавишу **Delete**.

П 1.3. Настройка отображения коммуникаций

Дважды щелкните левой кнопкой мыши объекте Коммуникации в дереве объектов или выделите этот объект и выберите пункт **Свойства** из контекстного меню. Откроется диалоговое окно **Настройка отображения**, в котором можно установить общие параметры отображения коммуникаций в плане и поперечном профиле.

Цвета линий коммуникаций, установленные на вкладках **Цвет трубопроводов**, **Цвет электрокабелей**, используются при создании новых коммуникаций и применяются ко всем существующим коммуникациям, у которых не задан индивидуальный цвет.

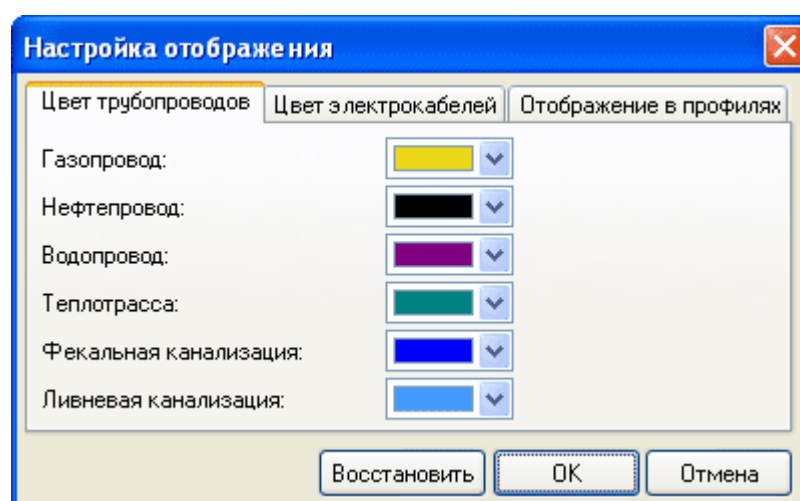


Рис. 22. Настройка цветов трубопроводов

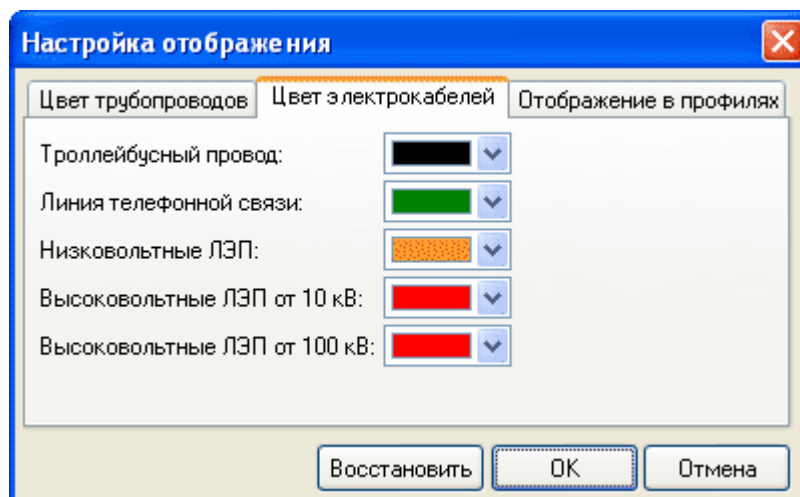


Рис. 23. Настройка цветов электрокабелей

На вкладке **Отображение в профилях** можно выбрать способ отображение коммуникаций в поперечном профиле: упрощенный или согласно соответствующим ГОСТам.

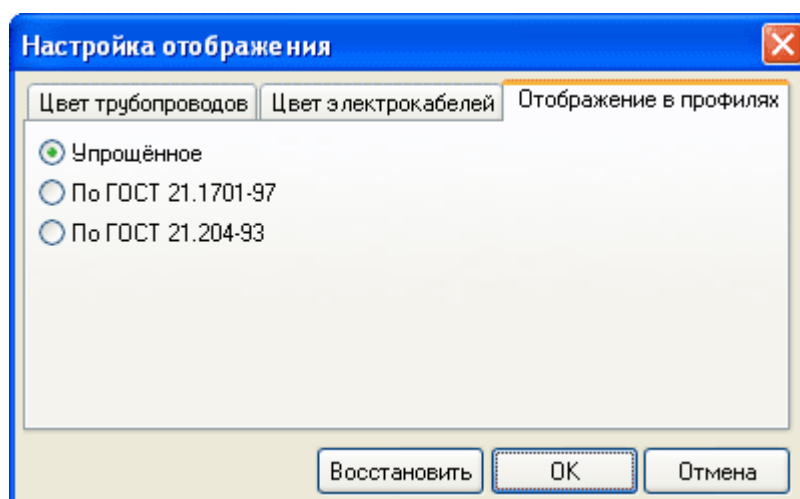


Рис. 24. Настройка отображения коммуникаций в поперечном профиле


Чтобы закрыть диалоговое окно с применением или без применения внесенных изменений, используйте кнопки **ОК** и **Отмена**. Кнопка **Восстановить** позволяет вернуть параметры, заданные в системе по умолчанию для разных типов коммуникаций.

II 1.4. Выделение коммуникаций

Чтобы включить режим выделения и редактирования коммуникаций, щелкните на кнопке **Выделить коммуникацию, задать свойства**, расположенной на дополнительной панели инструментов. Режим становится доступным при наличии коммуникаций на

активной поверхности. Выделенные линии и узлы окрашиваются в желтый цвет и доступны для редактирования. Количество выделенных объектов отображается в строке состояния.

Последовательное выделение коммуникаций

В режиме выделения коммуникаций стрелка мыши вблизи линий коммуникаций активной поверхности окрашивается в черный цвет, а вблизи узлов коммуникаций активной поверхности принимает вид прицела . Чтобы выделить объект коммуникации (линию или узел), щелкните на нем левой кнопкой мыши. Удерживая нажатой клавишу **Shift**, можно выделить несколько объектов (линий и/или узлов), последовательно щелкая на них мышью. Чтобы выделить все узлы определенной линии коммуникации, выделите эту линию или хотя бы один ее узел и выполните команду **Выделить образующие узлы** из контекстного меню. Команда **Выделить все** в меню **Правка** по умолчанию выделяет все линии коммуникаций. Чтобы выделить все узлы коммуникаций, выделите хотя бы один узел и выполните команду **Выделить все** в меню **Правка**.

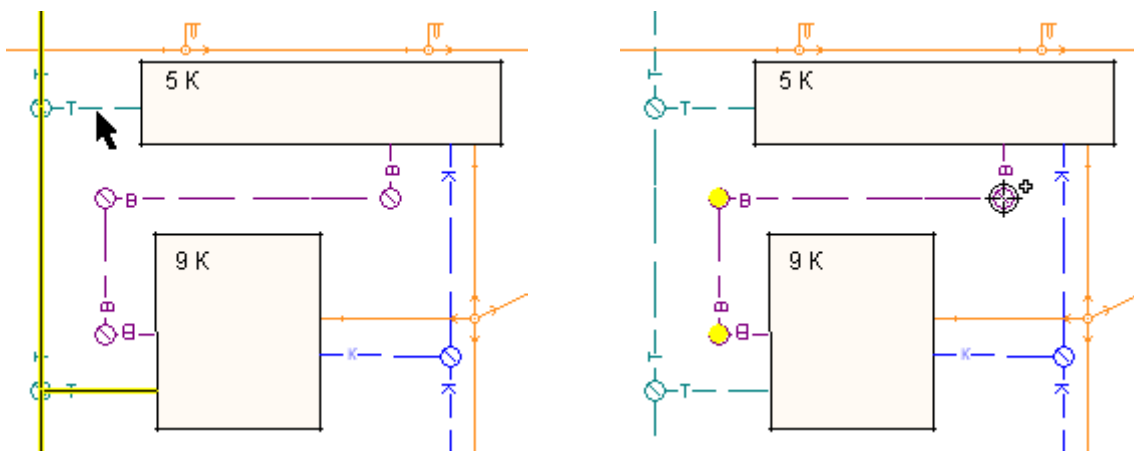


Рис. 25. Последовательное выделение линий и узлов коммуникаций

Выделение коммуникаций прямоугольной рамкой

Узлы коммуникаций удобнее выделять прямоугольной рамкой. Для этого нажмите левую кнопку мыши и растяните вокруг них рамку. Все узлы, попавшие внутрь этой рамки, становятся выделенными. Однако если внутрь рамки попадет линия коммуникации целиком, то будет выделена линия. Рамкой удобно выделять узлы, расположенные друг под другом, и узлы без условных обозначений.

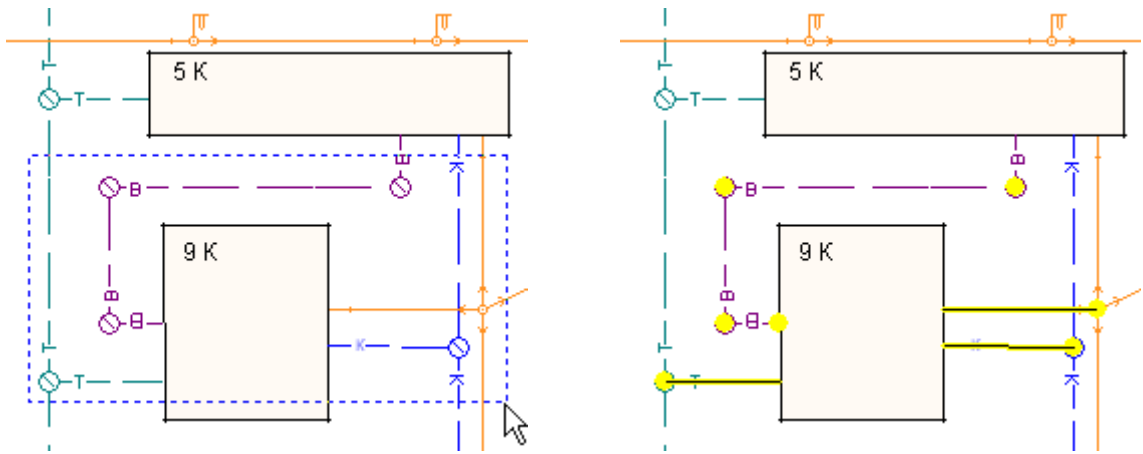


Рис. 26. Выделение линий и узлов коммуникаций прямоугольной рамкой

Особенности выделения узлов коммуникаций

При выделении узла одновременно выделяются узлы других линий, попавшие в некоторую малую окрестность этого узла (например, когда узлы находятся друг под другом или расположены довольно близко друг от друга). Эта возможность облегчает редактирование узлов-коммутаторов при разветвлении линий коммуникаций (см. раздел «Построение ответвлений»). Если требуется выделить один из узлов, удерживайте клавишу **Ctrl**. Чтобы выделить узел конкретной линии, попробуйте увеличить масштаб изображения или выделите образующие узлы этой линии, а затем снимите выделение с ненужных узлов.



Рис. 27. Выделение узлов: количество выделенных объектов отображается в строке состояния

Отмена выделения

Чтобы снять выделение с линии или узла, нажмите клавишу **Shift** и щелкните на объекте мышью или обведите его рамкой. Чтобы снять выделение со всех объектов, щелкните в любом месте плана или выполните команду **Правка|Снять выделение**.

П 1.5. Параметры коммуникаций

Параметры линий коммуникации

Для редактирования параметров линий коммуникации включите режим выделения и редактирования коммуникаций и дважды щелкните на линии. Или, выделите эту линию и выберите пункт **Свойства...** в меню **Правка** или контекстном меню. Откроется диалоговое окно, в котором можно задать параметры отображения линии коммуникации в плане и ее основные характеристики.

Параметры отображения

Для отображения разных типов коммуникаций используются специальные обозначения— Ех-шрифты, у которых можно настроить следующие параметры:

- **Масштаб.** Изменяя положение ползунка, можно установить масштаб Ех-шрифта, которым отображается линия коммуникации.
- **Цвет.** Цвета Ех-шрифтов, используемые при создании коммуникаций, задаются в общих настройках отображения (см. раздел «Настройка отображения коммуникаций»). Чтобы установить индивидуальный цвет линии, включите опцию **Цвет** и выберите подходящий из раскрывающейся палитры цветов. Чтобы вернуть цвет, заданный в общих настройках, отключите флажок этой опции.
- **Стиль в узлах.** Выберите эту опцию, чтобы применить Ех-шрифт к каждому сегменту линии отдельно.

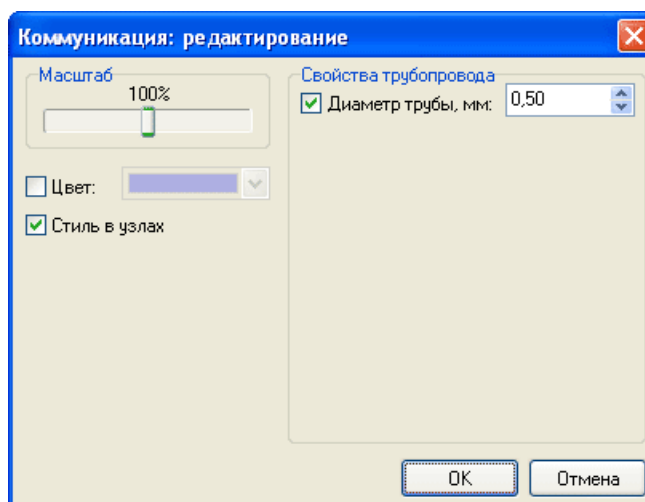


Рис. 28. Свойства линии коммуникации

Основные характеристики

В области **Свойства** отображаются основные характеристики коммуникации, количество которых зависит от ее типа.

- **Положение относительно поверхности.** Выбирается из раскрывающегося списка. Возможны варианты: Расположение **надземное**, **подземное**, **подземное в коробе**, **подземное в блоке**, **подземное в каналах**.
- **Диаметр или напряжение.** Если у трубопровода известен диаметр трубы, то выберите опцию Диаметр трубы и укажите его значение в числовом поле, иначе этот параметр будет считаться неизвестным. В 3D-виде трубопроводы с неизвестным диаметром отображаются с диаметром 150 мм. Для линий электропередач можно задать напряжение, выбрав опцию Напряжение.

Параметры узлов коммуникации

Для редактирования узлов коммуникации включите режим выделения и редактирования коммуникаций и дважды щелкните на узле. Или, выделите этот узел и выберите пункт **Свойства...** в меню **Правка** или контекстном меню. Откроется диалоговое окно, в котором можно задать параметры отображения узла коммуникации в плане, его координаты и основные характеристики.

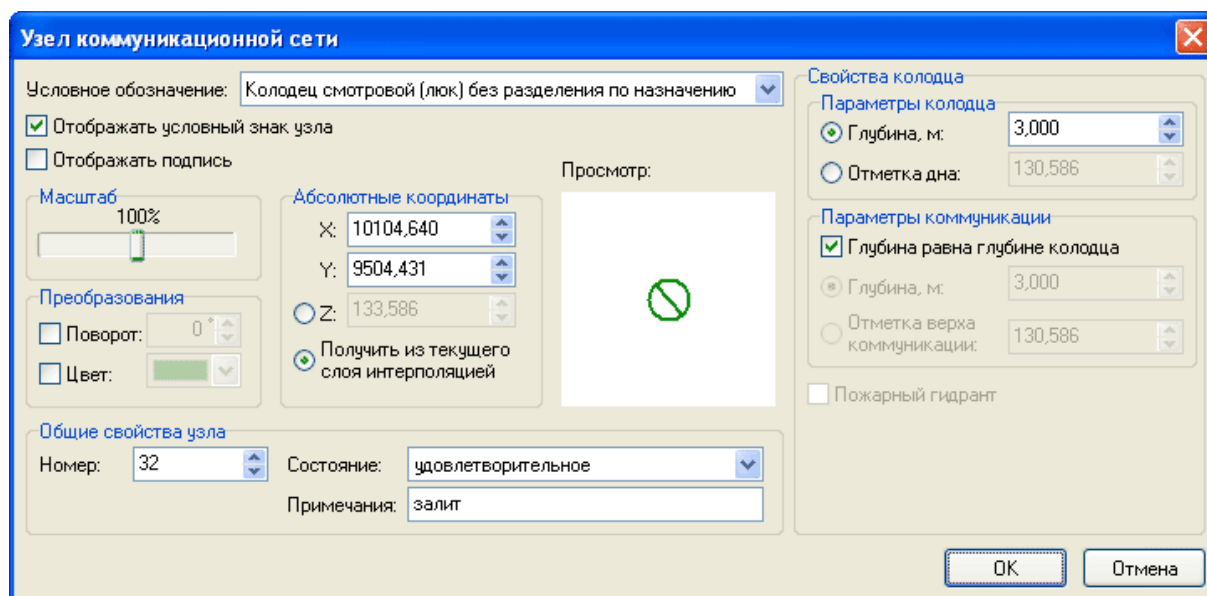


Рис. 29. Свойства узла коммуникации


Параметры отображения

- **Условное обозначение узла.** Для узла, в котором установлен колодец или опора, из раскрывающегося списка можно выбрать условный знак, которым он будет отмечаться на плане. Чтобы отменить отображение знака, отключите флажок в опции **Отображать условный знак узла**. Образец выбранного знака отображается в окне просмотра, которое обновляется при изменении масштаба и преобразовании знака. Под преобразованием знака понимается изменение его цвета и угла поворота. Для этого необходимо выбрать опции **Поворот** и **Цвет** и задать подходящий угол поворота и цвет знака. Чтобы восстановить установленные по умолчанию параметры знака, отключите флажки этих опций.

- **Подпись узла.** Для отображения подписи узла необходимо включить опцию **Отображать подпись**. В подписи отображается Z-отметка узла и отметка верха коммуникации (для колодцев).

Координаты узла

В группе элементов **Абсолютные координаты** отображаются координаты узла в плане (X,Y) и его Z-отметка, которая интерполируется по активной поверхности. Чтобы изменить высоту узла, установите переключатель рядом с числовым полем Z и введите необходимое значение.

 **Замечание.** Если координаты X, Y новой точки не принадлежат области определения активной поверхности, то координата Z этой точки по умолчанию принимает значение 100 или, если выбрана опция **Получить Z-отметку из слоя интерполяцией**, -1000.

Общие характеристики

Общие характеристики узла определяются в области **Общие свойства узла**. К ним относятся **Номер** узла и его **Состояние**: **хорошее**, **удовлетворительное** или **неудовлетворительное**. В поле **Примечания** можно ввести дополнительную информацию об узле.

Дополнительные характеристики

Дополнительные характеристики узла зависят от типа узла. Для колодцев можно задать **Глубину** или **Отметку дна** колодца и **Глубину залегания** или **Отметку верха** коммуникации. В колодцах водопроводов можно установить пожарные гидранты. Для этого достаточно выбрать опцию **Пожарный гидрант**. Для опор можно задать **Высоту** опоры.

II 1.6. Редактирование коммуникаций


Все операции над коммуникациями выполняются в режиме выделения и редактирования коммуникаций, который включается кнопкой **Выделить коммуникацию, задать свойства**, расположенной на дополнительной панели инструментов. Этот режим позволяет устанавливать свойства линий и узлов коммуникаций, изменять путь коммуникации, параметры отображения в плане и 3D-виде.

Добавление и удаление

Чтобы добавить узел к линии коммуникации, включите режим выделения и редактирования коммуникаций, укажите курсором положение узла на линии и выполните команду **Вставить узел** из контекстного меню.

Чтобы удалить узел коммуникации, щелкните на нем правой кнопкой мыши, выполните команду **Удалить узел** из контекстного меню и подтвердите удаление узла в запросе системы. Еще один способ удаления узла: выделить этот узел и выполнить команду **Удалить** в меню **Правка** или нажать на клавишу **Delete**.

Перемещение узлов и линий коммуникаций

Чтобы переместить узел, включите режим выделения и редактирования коммуникаций, выделите узел и поместите на него курсор. После того как курсор примет вид «прицела» , нажмите левую кнопку мыши и переместите узел. Эта техника перемещения называется «Drag'n'Drop». Если выделено несколько узлов, то при перемещении одного из них соответственно смещаются остальные. Координаты узлов можно уточнить при редактировании свойств узла.

Линии коммуникаций перемещаются также как и узлы. Для перемещения выделите линию и перетащите ее с помощью мыши.

Преобразование узлов

Чтобы изменить тип узла, заданный при создании коммуникации, включите режим выделения и редактирования коммуникаций, щелкните на узле правой кнопкой мыши и в контекстном меню выберите пункт **Преобразовать в колодцы, опоры** или обычные **узлы**.

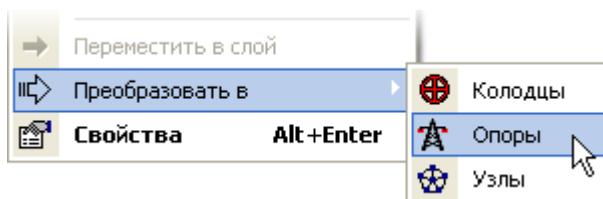


Рис. 30. Преобразование узлов

Откроется диалоговое окно для настройки параметров преобразования узлов. При выборе опций **Сохранять условный знак** и **Сохранять Z-отметку** условные знаки и Z-отметки преобразуемых узлов остаются неизменными, иначе условный знак узла определяется по его типу, а Z-отметка – по высоте линии коммуникации.

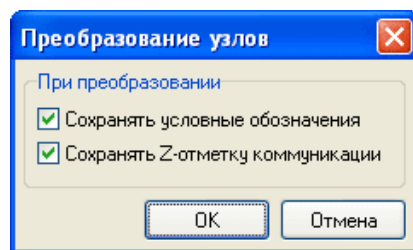


Рис. 31. Настройка преобразования узлов

Перемещение коммуникаций в другой слой

Выделите коммуникации, которые требуется переместить в другой слой. Откройте контекстное меню, выберите пункт **Переместить в слой** и укажите название слоя в появившемся подменю.

Построение ответвлений

Ответвления линий коммуникаций создаются как отдельные линии, имеющие общие узлы-коммутаторы с основной линией. Чтобы точнее совместить два узла, включите режим привязки к объектам. Затем включите режим выделения и редактирования коммуникаций, выделите эти узлы и задайте им одинаковые свойства. Напомним, что в этом случае (когда узлы находятся друг под другом или очень близко друг к другу) одним щелчком мыши выделяются сразу все узлы (см. для проверки число выделенных объектов в строке состояния).