

Формирование графических схем участков электрических сетей на основе единой информационной модели

А.В.Кочеткова¹, Т.Н.Кульман², А.О.Макаров²

¹ООО «Систел», Москва, Россия

²Филиал «Протвино» государственного университета «Дубна», г. Протвино, Московская обл., Россия

Аннотация

Статья посвящена практическому применению внедряемой в настоящее время в российскую электроэнергетику единой информационной модели (ЕИМ). Использование этой модели позволяет автоматизировать ряд задач интеграции существующих в этой сфере информационных систем. В этом материале рассматривается, как единая информационная модель помогает в создании конфигурационной базы данных ОИК «Систел» и формировании графических схем участков электрической сети. Кратко описывается разработанная для этих целей программа.

Ключевые слова: ЕИМ, электрические сети, электроэнергетика, граф, дерево, алгоритм, C++.

Formation of graphic diagrams of sections of electrical networks based on a unified information model

A.V.Kochetkova¹, T.N.Kulman², A.O.Makarov²

¹Systel LLC, Moscow, Russia

²Branch «Protvino» of the State University «Dubna», Protvino, Moscow region, Russia

Abstract

The article is devoted to the practical application of the Unified Information Model (UIM), which is currently being introduced into the Russian electric power industry. The use of this model allows you to automate a number of tasks of integrating existing information systems in this area. This material examines how a unified information model helps in creating the Systel OIC configuration database and generating graphical diagrams of sections of the electrical network. The program developed for these purposes is briefly described.

Keywords: EIM, electrical networks, electrical power engineering, graph, tree, algorithm, C++.

1. Введение

В России в 2019-2022 гг. была создана серия стандартов ГОСТ 58561 [1, 2, 3, 4], определяющая единую информационную модель (ЕИМ) российской энергосистемы, совместимую с международной моделью CIM (от англ. Common Information Model).

Эта модель была разработана Международной Электротехнической комиссией в 2005 году. Она описана в стандарте МЭК-61970 [5]. На его основе в настоящее время ЕИМ активно внедряется в электроэнергетику: в рамках неё уже описано более 6,5 миллионов базовых объектов [6], таких, как подстанции, трансформаторы, линии электропередачи и др.

Данная работа была выполнена на предприятии ООО «Систел» [7], организованном в 1996 году. Предприятие специализируется на выполнении работ в области автоматизации подстанций и диспетчеризации процессов передачи, распределения и учета электроэнергии в электрических сетях. Компанией разработаны и внедрены десятки программно-технических комплексов на подстанциях 6 – 110 кВ и диспетчерских пунктах.

В ПАО «Россети» была разработана программы РС-20, с помощью которой предприятия электроэнергетики создают собственные ЕИМ. Эти модели в виде XML-файлов предоставляются всем сторонним организациям, в том числе в ООО «Систел». Эти файлы могут быть очень большого размера, до нескольких миллионов

строк.

В ООО «Систел» был создан целый комплекс программ, решающий различные задачи, связанные с использованием моделей ЕИМ на предприятиях электроэнергетики. В частности, в ООО «Систел» есть программа, читающая данные ЕИМ, представленные в виде XML-файлов, и формирующая собственную БД ЕИМ.

В ООО «Систел» имеются также другие БД, используемые для самых разных задач. В данной работе будет использоваться конфигурационная БД. Таким образом, исходными данными для работы являются БД ЕИМ и конфигурационная БД.

Целью данной работы являлась разработка программы по формированию на основе этих БД графических схем участков электрических сетей.

Для достижения поставленной цели требуется решить следующие **задачи**:

1. Изучить состав общей информационной модели – основных сущностей, связей между ними, их семантику;
2. Разработать алгоритм заполнения некоторых таблиц в конфигурационной БД ОИК «Систел» данными из БД ЕИМ;
3. Разработать алгоритм для генерирования схем подстанций на основе их топологического графа, представленного в модели;
4. Создать приложение с графическим интерфейсом, реализующее упомянутые выше алгоритмы, с возможностью конфигурации его работы.

2. Обзор основных пакетов СИМ-модели

СИМ-модель, или общая информационная модель, – это абстрактная модель, которая описывает наиболее важные для предприятия электроэнергетики объекты, участвующие в его работе.

СИМ-модель описана с использованием средств объектно-ориентированного моделирования. В качестве языка моделирования применяется язык UML, который задаёт модель в виде набора пакетов. Каждый такой пакет содержит одну или более диаграмм классов, на которых графически представлены классы, входящие в пакет. Каждый класс в свою очередь определяется своими свойствами (атрибутами) и отношениями с другими классами. Разделение модели на пакеты сделано в связи с её большим размером.

В стандарте МЭК-61970 описано около десятка пакетов [8]. Основными среди этих пакетов являются следующие:

1. Domain;
2. Core;
3. Topology.

2.1. Пакет Domain

Этот пакет содержит определения примитивных типов данных, включающих единицы измерения и допустимые значения.

2.2. Пакет Core

Данный пакет включает основные сущности – ресурс энергосистемы (PowerSystemResource) и проводящее оборудование (ConductingEquipment).

На рисунке 1 представлена главная диаграмма классов этого пакета.

Рассмотрим данную диаграмму подробнее.

Контейнеры оборудования применяются для группирования оборудования,

которое находится на подстанции. Классы Substation (подстанция), VoltageLevel (распределительное устройство) и Bay (группа коммутационного оборудования) являются контейнерами оборудования. При этом группа коммутационного оборудования может быть частью как распределительного устройства, так и самой подстанции. В свою очередь распределительное устройство является частью подстанции. Подстанции могут входить в некоторую область (SubGeographicalRegion), а области – в один регион (GeographicalRegion).

Любой контейнер оборудования может иметь в своём составе некоторое оборудование (Equipment). Часть оборудования является электропроводящим (ConductingEquipment), то есть способным проводить ток. Такое оборудование может иметь несколько полюсов, с помощью которых оно подключается к другому оборудованию.

Распределительные устройства и проводящее оборудование работают под определённым напряжением, поэтому соответствующие классы, то есть VoltageLevel и ConductingEquipment, связаны с классом BaseVoltage, который содержит информацию об уровне рабочего напряжения.

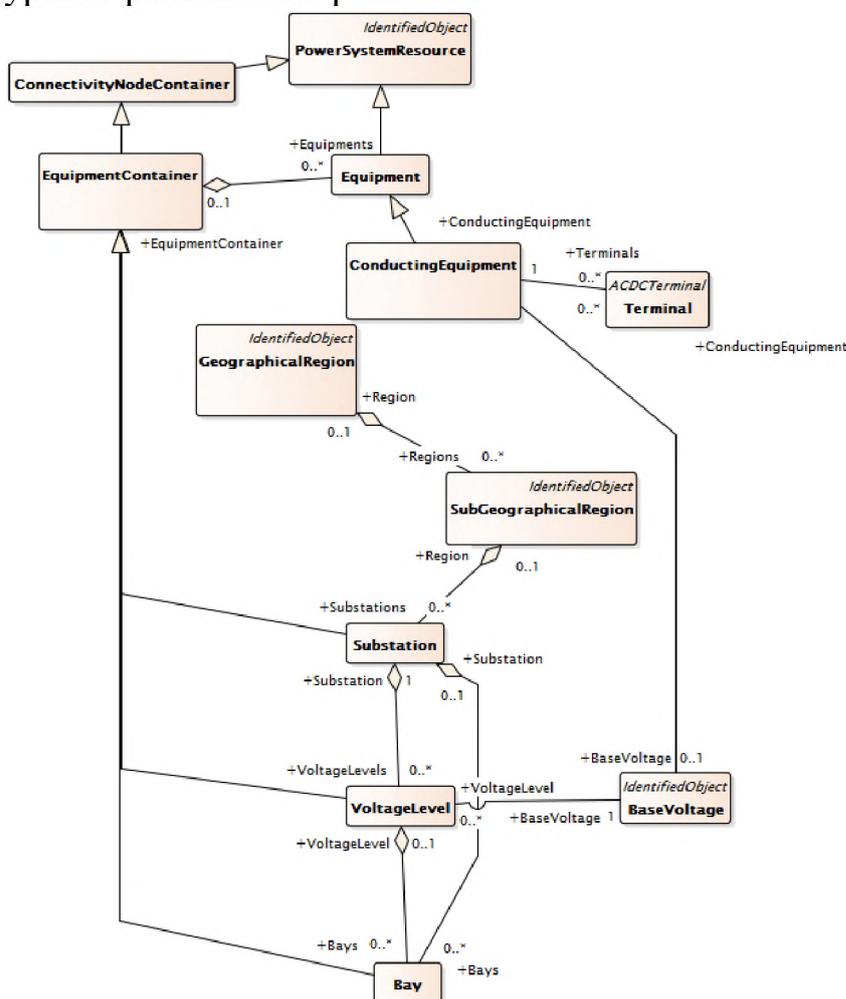


Рисунок 1. Диаграмма Core.Main

2.3. Пакет Topology

Пакет является расширением пакета Core и в связке с классом Terminal моделирует связность (Connectivity), то есть физическое определение того, как оборудование соединено друг с другом. Дополнительно этот пакет моделирует топологию – логическое определение соединений между оборудованием через замкнутые переключатели.

3. Алгоритм построения и импорта дерева объектов

В программе определяется список сущностей CIM-модели, которые требуется импортировать. В этот список входят подстанции (Substation), распределительные устройства (VoltageLevel), линии электропередачи (Line), ячейки (Bay) и оборудование (Equipment). Каждой сущности ставится в соответствие список сущностей-родителей.

Первым этапом построения дерева объектов является получение экземпляров сущностей, указанных в списке, и запись их в общий массив узлов. На их основе создаются узлы дерева объектов. Затем эти узлы сортируются в порядке возрастания по своему уникальному идентификатору. Сортировка узлов в дальнейшем позволит использовать двоичный поиск, чтобы находить узлы по идентификаторам.

Следующим шагом является обход всего массива узлов. Для каждого узла с применением алгоритма двоичного поиска ищется его родитель. Если родитель был найден, то в родительский узел заносится ссылка на узел-потомок, а в узел-потомок – ссылка на своего родителя. У одного узла может быть сколько угодно потомков, в том числе и ноль, однако родитель может быть только один.

Чтобы импортировать дерево объектов в базу данных ОИК, производится его рекурсивный обход начиная с тех узлов, у которых нет родителей. Сначала в базу данных заносится сам узел, после чего – его потомки.

4. Алгоритм генерации схем

Алгоритм генерации схем является более сложным процессом, поэтому он разбит на несколько этапов:

1. Формирование графа оборудования на основе узла подстанции из дерева объектов;
2. Размещение оборудования подстанции на схеме с помощью графа оборудования;
3. Соединение оборудования коннекторами и создание топологического графа схемы;
4. Добавление подписей;
5. Привязка телемеханических сигналов к соответствующему оборудованию.

Рассмотрим более детально первые три этапа, поскольку именно они являются наиболее важными.

4.1. Формирование графа оборудования

Под графом оборудования следует понимать такой граф, вершинами которого является оборудование, а рёбрами – физические соединения между этим оборудованием. Этот граф является промежуточным представлением схемы, которое упрощает её генерацию в программе.

Граф оборудования формируется на основе графа связности из CIM-модели, вершины которого представляют собой узлы соединения (ConnectivityNode) и само оборудование. Оборудование связывается с другим оборудованием посредством узлов соединения, к которым оно подключается с помощью своих полюсов (Terminal). На рисунке 2 представлен пример графа связности.

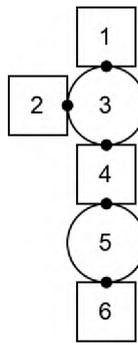


Рисунок 2. Пример графа связности

Здесь узлы соединения изображены в виде окружностей, а оборудование – в виде квадратов. Чёрные точки представляют собой полюсы, которыми оборудование входит в узлы соединения. Внутри узлов соединения и оборудования находятся их идентификаторы.

Иначе этот граф можно представить так, как это изображено на рисунке 3.

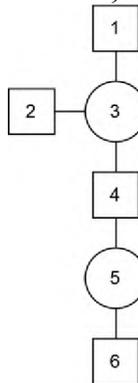


Рисунок 3. Пример видоизменённого графа связности

Полюсы оборудования на этом рисунке для наглядности заменены на линии.

Кратко рассмотрев структуры графа связности, вернёмся к процессу формирования графа оборудования на его основе.

Формирование графа оборудования начинается с получения списка всего оборудования подстанции, для которой генерируется схема. Затем для каждой единицы оборудования из базы данных запрашивается список полюсов, которые принадлежат данному оборудованию. Уже на этом шаге можно проверить корректность данных модели, если какое-либо оборудование не имеет ни одного полюса, когда предполагается обратное.

Для оставшегося оборудования производится поиск узлов соединения, в которые оно входит. Чтобы этого достичь, для каждого полюса оборудования в базе данных запрашивается индекс узла соединения, к которому подключён данный полюс. Здесь также может оказаться так, что оборудование, хотя и имеет полюсы, не входит ни в один узел соединения, что может являться ошибкой заполнения модели.

Следующим этапом является добавление в список оборудования сегментов линий, по которым энергия подаётся на подстанцию. Эти сегменты линий будут играть роль входных узлов в графе оборудования. При работе с этим графом его обход следует начинать именно с сегментов линий.

После того, как были определены входные узлы, запускается обход графа «в ширину». Во время обхода неориентированный граф связности преобразуется в ориентированный. Именно на этом шаге начинает формироваться граф оборудования.

Следующий этап состоит в том, чтобы заменить узлы соединений, в которые входят шины, на сами шины для более удобной обработки графа в будущем. Для этого в ориентированном графе связности ищутся узлы, которые непосредственно

приводят к шине. Затем все связи этих узлов передаются шинам, а сами узлы удаляются из графа.

Аналогично из ориентированного графа удаляются все узлы соединения.

4.2. Размещение оборудования подстанции на схеме с помощью графа оборудования

Данный этап генерирования схемы заключается в присваивании оборудованию координат. В качестве основы для размещения схема разбивается на равные по размеру клетки. Количество этих клеток зависит от размеров графа оборудования. В общем виде способ размещения оборудования определяется его типом:

- Шины размещаются от центра одной клетки до центра другой строго в горизонтальном направлении;
- Коммутирующее оборудование и трансформаторы помещаются в центр клеток;
- Сегменты кабелей и сегменты линий размещаются на вертикальной оси клеток.

Конечное положение оборудования определяется:

- Уровнем, на котором расположено оборудование;
- Количеством клеток, которое занимает оборудование;
- Смещение оборудования относительно начала уровня.

Уровень играет роль ординаты оборудования, а количество занимаемых клеток и смещение – роль абсциссы.

Уровень, на котором размещается оборудование, определяется при обходе графа оборудования в ширину. Максимальное значения уровня интерпретируется как глубина графа оборудования.

Количество занимаемых клеток и смещение задаются в ходе следующего алгоритма:

1. Граф оборудования обходится по секциям;
2. Обход графа начинается с входных сегментов линий;
3. Начальное смещение оборудования для первой секции равно нулю;
4. В зависимости от того, есть ли у какого-либо оборудования потомки, количество занимаемых ими клеток равно или минимально возможному значению для данного типа оборудования (всё оборудование занимает по крайней мере одну клетку, кроме шин, занимающих как минимум две), или сумме количества занимаемых клеток у потомков;
5. Ширина обрабатываемой секции равна максимальной сумме занимаемых клеток на одном уровне;
6. Смещение оборудования внутри секции вычисляется так: для каждого потомка его смещение равно смещению его родителя плюс сумма занимаемых клеток уже обработанных потомков родителя. Если у оборудования нет потомков, оно лишь принимает вычисленное родителем смещение;
7. Смещение для следующей секции определяется как сумма начального смещения оборудования текущей секции и её ширины;
8. Если текущая секция является последней, то алгоритм заканчивает работу, иначе перейти к пункту 4 для обработки следующей секции.

Следующим этапом размещения оборудования является присвоение ему координат. Для этого выполняется следующее:

1. Вычисляется верхний левый угол клетки.
2. Для коммутирующего оборудования и трансформаторов сначала определяется позиция внутри уровня.
3. После этого вычисляется центральная точка оборудования.
4. Для шин, сегментов линий и кабелей требуется определить их начальную и конечную точку.

После присвоения координат оборудованию требуется их преобразовать, поскольку на данном этапе:

1. Точка с координатами (0;0) (т. е. нулевая точка) находится в левом верхнем углу схемы, тогда как должна располагаться в её центре;
2. Направление оси ординат инвертировано;

Чтобы привести координаты оборудования к требуемому виду:

1. Вычисляется точка $(X_0'; Y_0')$, в которую следует переместить нулевую точку;
2. Из координат $(X_i; Y_i)$ i -го экземпляра оборудования вычитаются координаты $(X_0'; Y_0')$;
3. Ордината полученных координат i -го оборудования $(X_i'; Y_i')$ умножается на «-1».

После этого оборудование на схеме имеет корректно вычисленные координаты.

4.3. Соединение оборудования коннекторами и создание топологического графа схемы

Для визуализации физических связей между оборудованием оно соединяется на схеме с помощью коннекторов. Вместе с этим создаётся топологический граф схемы. Этот граф нужен для того, чтобы находить на схеме обесточенные участки и окрашивать их в соответствующий цвет.

Топологический граф похож по своей структуре на модель связности оборудования в СИМ-модели. Он состоит из вершин, соединяющих оборудование, которое представляет собой рёбра топологического графа.

Для его формирования граф обходится «в ширину». На каждом шаге алгоритма создаются вершины, связывающие оборудование. Оборудование соединяется между собой коннекторами, которые выглядят на схеме как прямые линии. Между оборудованием и коннекторам формируются соединения и заносятся в соответствующую вершину. После этого вершины соединяются между собой рёбрами.

Таким образом формируется топологический граф схемы.

5. Приложение, реализующее описанные алгоритмы

Приложение создавалось с помощью языка программирования C++17 и нескольких библиотек для реализации отдельных функций:

1. **wxWidgets** [9] – библиотека для разработки кросс-платформенных программ с графическим пользовательским интерфейсом;
2. **Boost** [10] – собрание более 150 библиотек, работающих на многих платформах и покрывающих широкий спектр задач. Ниже перечислены библиотеки из Boost, которые применялись в ходе разработки программы:
 - **PropertyTree**, предоставляющая древовидную структуру данных, подходящую для работы с XML-файлами;
 - **Format** для поддержки типобезопасных форматируемых строк;
 - **Algorithm**, содержащая большой набор универсальных алгоритмов;

3. **pqxx** [11] – библиотека для создания кросс-платформенных клиентских приложений, упрощающая взаимодействие с сервером PostgreSQL [12]. На нём хранятся базы данных ОИК и модели.

Применение указанных выше технологий, которые могут быть использованы в проприетарных проектах, делает возможным перенос программы с одной операционной системы (Windows) на другую (Linux).

5.1. Интерфейс пользователя

Пользовательский интерфейс показан на рисунке 4. Для демонстрации была взята ЕИМ Николаевского РЭС.

Окно программы разделено на две части:

1. Дерево объектов, которое можно раскрывать и просматривать;
2. Список подстанций напряжением не более 10 кВ.

Импорт дерева объектов в БД ОИК и генерация схем подстанций осуществляется с помощью меню *Инструменты* в верхней части экрана. В меню *Внешний вид схем* можно настроить соответствие напряжения оборудования цвету, изменить ширину линий элементов на схеме, а также сконфигурировать цветовую окраску дискретных сигналов. В меню *Настройки* можно указать путь в файловой системе, по которому будут записываться файлы генерируемых схем.

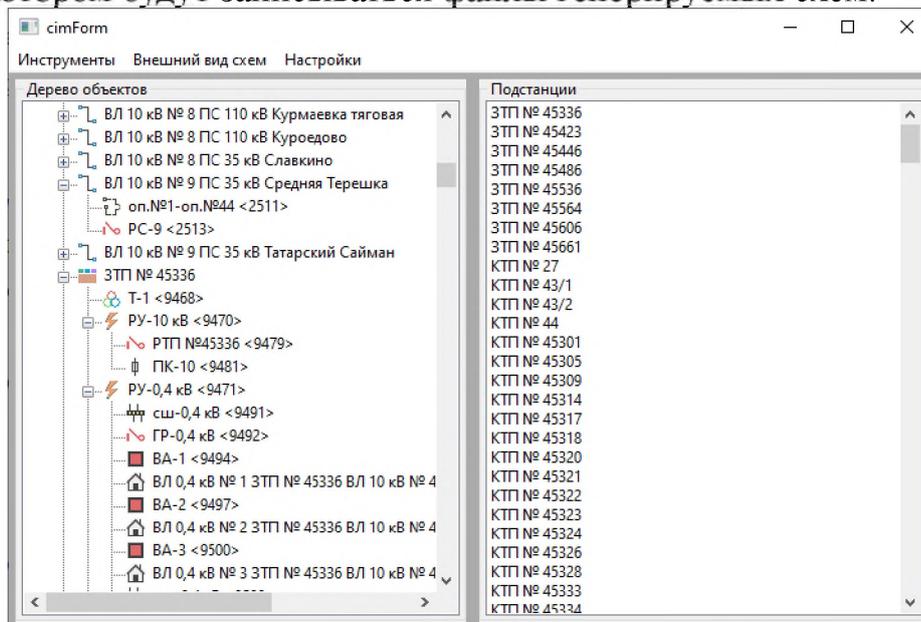


Рисунок 4. Интерфейс разрабатываемой программы

5.2. Импорт дерева объектов

В правильности импорта дерева можно убедиться с помощью программы для конфигурации БД ОИК. На рисунке 5 изображено как выглядит дерево объектов в этой программе после импорта.

5.3. Генерация схем

Генерация схем происходит на основе выбранных пользователем подстанций из их списка. По завершении генерации программа формирует текстовый отчет по сгенерированным схемам.

Пример сгенерированной схемы изображён на рисунке 6.

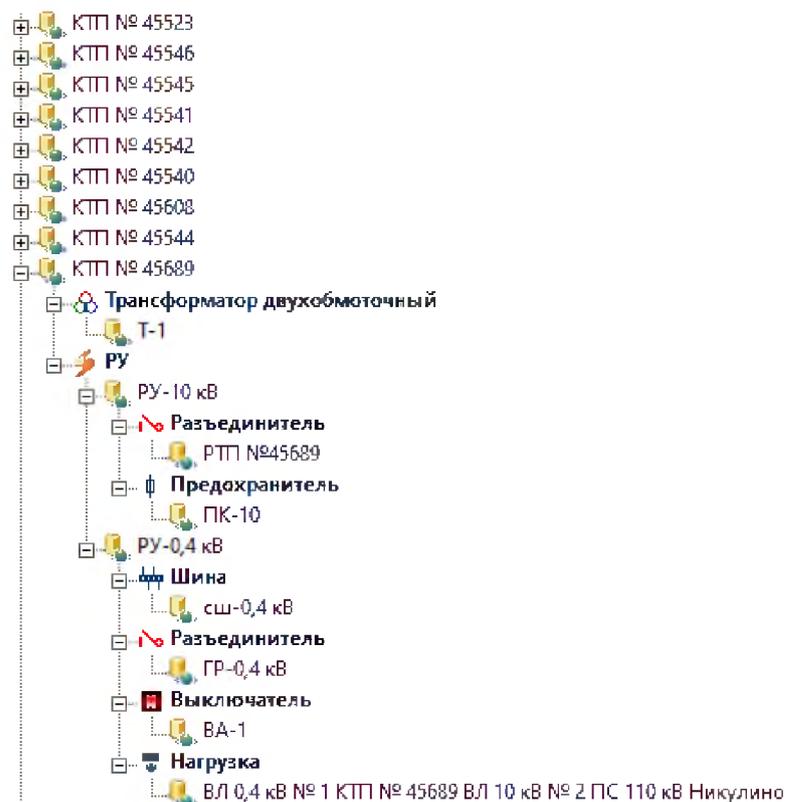


Рисунок 5. Дерево объектов в конфигурационной программе

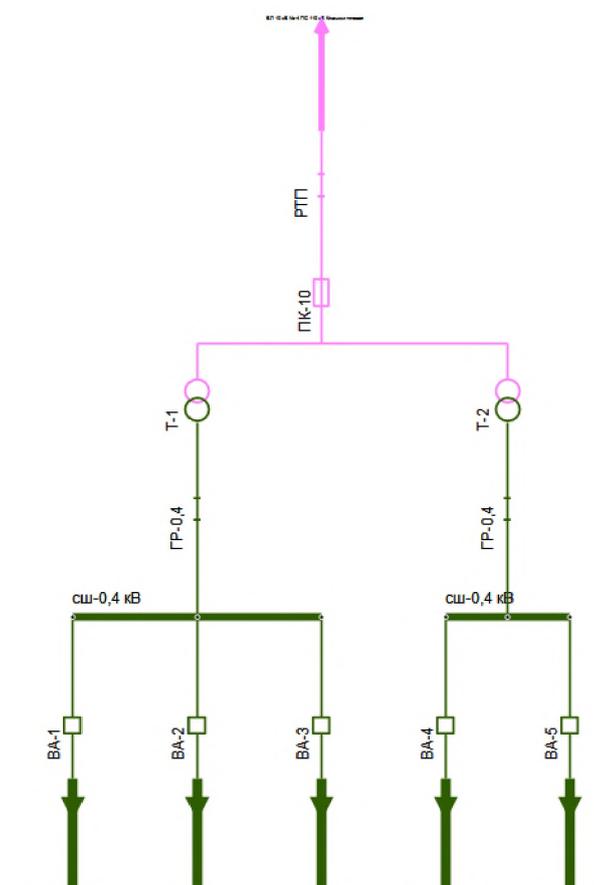


Рисунок 6. Пример сгенерированной схемы

Заключение

В результате выполнения данной работы была разработана программа по формированию на основе баз данных ОИК и ЕИМ графических схем участков электрических сетей.

Были решены следующие задачи:

1. Изучен состав общей информационной модели;
2. Разработан алгоритм заполнения БД ОИК «Систел» данными из БД ЕИМ;
3. Разработан алгоритм для генерирования схем подстанций;
4. Создано приложение с графическим интерфейсом, реализующее эти алгоритмы.

Библиографический список

- [1] ГОСТ Р 58651.1-2019. Единая энергетическая система и изолированно работающие энергосистемы. Информационная модель электроэнергетики. Основные положения. – Введ. 01.01.2020. – М. : Стандартинформ, 2019 – 28 с.
- [2] ГОСТ Р 58651.2-2019. Единая энергетическая система и изолированно работающие энергосистемы. Информационная модель электроэнергетики. Базисный профиль информационной модели. – Введ. 01.01.2020. – М. : Стандартинформ, 2019 – 32 с.
- [3] ГОСТ Р 58651.2-2019. Единая энергетическая система и изолированно работающие энергосистемы. Информационная модель электроэнергетики. Профиль информационной модели линий электропередачи и электросетевого оборудования напряжением 110-750 кВ. – Введ. 01.01.2021. – М. : Стандартинформ, 2020 – 69 с.
- [4] ГОСТ Р 58651.2-2019. Единая энергетическая система и изолированно работающие энергосистемы. Информационная модель электроэнергетики. Профиль информационной модели генерирующего оборудования. – Введ. 01.01.2021. – М. : Стандартинформ, 2020 – 30 с.
- [5] Energy management system application program interface (EMS-API) - Part 301: Common information model (CIM) base (IEC 61970-301:2020).
- [6] Богомолов, Р. А. Создание CIM-модели в АО «СО ЕЭС» [Текст] / Р. А. Богомолов // Электроэнергия. Передача и распределение. - 2021. № 2. - С. 26-31.
- [7] Рыкованов С.Н, Кульман Н.Ю., Ухов В.И. Развитие программного обеспечения верхнего уровня ПТК «Систел». Современные системы сбора, передачи, обработки и отображения информации на объектах электроэнергетики. Сборник докладов. Москва НЦ ЭНАС, 2006г.
- [8] Current CIM Model Drafts - All Documents. - Текст : электронный // CIM Users Group : [сайт]. - URL: <https://cimug.ucaiug.org/CIM%20Model%20Releases/Forms/AllItems.aspx> (дата обращения: 14.04.2024).
- [9] Overview - wxWidgets. - Текст : электронный // wxWidgets: Cross-Platform GUI Library : [сайт]. - URL: <https://wxwidgets.org/about/> (дата обращения: 09.04.2024).
- [10] Welcome to Boost.org!. - Текст : электронный // Boost C++ Libraries : [сайт]. - URL: <https://www.boost.org/> (дата обращения: 09.04.2024).
- [11] The C++ connector for PostgreSQL. - Текст : электронный // libpqxx: the official C++ language binding for PostgreSQL : [сайт]. - URL: <https://pqxx.org/libpqxx/> (дата обращения: 09.04.2024).
- [12] PostgreSQL: About. - Текст : электронный // PostgreSQL: The World's Most Advanced Open Source Relational Database : [сайт]. - URL: <https://www.postgresql.org/about/> (дата обращения: 09.04.2024).