

# Практический опыт промышленной реализации системы управления гетерогенной транспортной сетью

М. В. Наганов  
nmv@tercom.ru

Создание систем управления гетерогенными телекоммуникационными сетями является трудоемкой и наукоемкой задачей. В данной статье описывается практический опыт реализации системы управления гетерогенной транспортной сетью, проведенный сотрудниками кафедры Системного программирования СПбГУ и ГУП «Терком» в период с 2002 по 2004 год. Приведен анализ проделанной работы, описаны возникшие проблемы и некоторые технологические решения.

## Введение

В настоящее время российские операторы телекоммуникационных сетей вынуждены мириться с тем, что у них отсутствует возможность использовать централизованное управление для всей принадлежащей им сети. В лучшем положении находятся операторы, строго придерживающиеся правила вести закупки оборудования только одного производителя. При условии, что производитель сетевого оборудования также занимается и разработкой системы управления для него, оператор автоматически получает централизованную систему управления всей своей сетью.

Однако рассмотренная выше ситуация может быть справедливой только для относительно небольших операторов. Более-менее

крупный оператор или организация, имеющая собственную распределенную телекоммуникационную сеть (а к таким организациям, кстати, относятся и наши государственные, в том числе силовые ведомства), неизбежно сталкивается с тем, что на его сети мирно сосуществует оборудование различных производителей, как иностранных, так и отечественных. Такие сети принято называть *гетерогенными*.

Гетерогенность телекоммуникационных сетей проявляется не только в использовании оборудования различных производителей, но и в одновременном сосуществовании в сети различных технологий передачи данных, а также голосового трафика и трафика данных.

Типичный «центр управления» гетерогенной сетью представляет собой несколько персональных компьютеров, работающих под разными операционными системами, на которых «крутятся» системы управления различных производителей. В некоторых случаях даже для разных версий оборудования одного и того же производителя используются различные несовместимые друг с другом версии систем управления.

Тот факт, что обслуживающему персоналу сети приходится изучать множество систем управления, является, наверное, самой простой проблемой. Наиболее неприятные следствия такой ситуации — это то, что оператор не может «посмотреть» на свою сеть в целом нигде, кроме как на бумаге (или большой карте), сталкивается с трудностями при попытке внедрить системы биллинга или ввести дополнительные услуги для пользователей на своей сети. А это в конечном итоге означает потерю потенциальных прибылей.

Естественно, что проблема управления сетями связи не остается без внимания исследователей и коммерческих фирм, и, как результат их деятельности, существуют готовые системы управления гетерогенными сетями. Что мешает операторам использовать их? В первую очередь, конечно, стоимость таких систем. Например, стоимость решений на базе системы управления сетью HP OpenView OEMF фирмы Hewlett-Packard начинается от \$200 000. Кроме того, внедрение системы управления связано с большими косвенными расходами — на обучение персонала, интеграцию систем управления оборудованием в систему управления сетью, возможно даже появится необходимость в смене стратегии управления сетью.

Для представителей же силовых ведомств еще одной проблемой является сильное недоверие к программному и аппаратному обеспе-

чению иностранных производителей, как потенциально содержащему вредоносные «закладки». В то же время немногие отечественные производители могут похвастать тем, что их системы управления оборудованием готовы к интеграции в системы управления сетью. Да и отечественных систем управления сетями пока не наблюдается в большом количестве на рынке программного обеспечения.

В данной статье описывается практический опыт реализации системы управления гетерогенной транспортной сетью, проведенный сотрудниками кафедры Системного программирования СПбГУ и ГУП «Терком» в период с 2002 по 2004 год. Приведен анализ проделанной работы, описаны возникшие проблемы и некоторые технологические решения. Кроме собственно создания системы управления, нам пришлось совместно с производителем оборудования заниматься вопросами интеграции его системы управления в единую систему управления. Из соображений соблюдения коммерческой тайны мы не приводим в данной статье названия организаций заказчика и соисполнителя.

## **1. Технологии управления телекоммуникационными сетями**

Проблема управления телекоммуникационными сетями уже долгое время находится под пристальным вниманием представителей науки, организаций по стандартизации и коммерческих фирм — производителей сетевого оборудования. Основные тенденции в развитии систем управления таковы:

- стандартизация решений;
- использование открытых стандартов;
- интеграция систем управления;
- децентрализация управления.

Технологической основой для построения систем управления являются стандарты сети управления электросвязью (Telecommunications Management Network, TMN), разработанные Телекоммуникационным сектором Международного Союза Электросвязи (МСЭ-Т, ITU-T). В 2005 году исполняется 17 лет с момента публикации первой версии набора стандартов по TMN. Более того, сама концепция

TMN основана на еще более известной семиуровневой модели взаимодействия открытых систем, стандартизированной Международной Организацией по Стандартизации (МСО, ISO). Достаточно подробное изложение концепций TMN можно найти в [1].

Для дальнейшего обзора нам понадобится список функциональных областей управления (известный под аббревиатурой FCAPS), определенный в Рек. МСЭ-Т X.700 [9]:

- управление неисправностями (fault management);
- управление конфигурацией (configuration management);
- управление расчетами за услуги (accounting management);
- управление производительностью (performance management);
- управление безопасностью (security management).

Каждая функциональная область указывает ресурсы, которые могут изменяться в процессе функционирования системы для достижения цели управления.

Также нам понадобится список уровней логической многоуровневой архитектуры управления (Logical Layered Architecture), определенный в Рек. МСЭ-Т М.3010 [7]:

- уровень сетевых элементов (network elements layer);
- уровень управления элементами (elements management layer);
- уровень управления сетью (network management layer);
- уровень управления услугами (services management layer);
- уровень управления бизнесом (business management layer).

На уровне сетевых элементов в архитектуре управления находится собственно управляемое оборудование. К уровню управления элементами относится управление рабочими характеристиками каждого из элементов оборудования. Набор параметров управления и способ воздействия на сетевые элементы различаются для каждого производителя сетевого оборудования. К уровню управления сетью относится выполнение задач, требующих согласованного изменения настроек нескольких элементов сети (например, таблиц

маршрутизации) посредством функциональности уровня управления элементами.

Большинство систем управления фирм-производителей оборудования работает на уровнях управления сетью и элементами, предоставляя возможности по изменению параметров каждого сетевого элемента в отдельности и всей сети в целом. Однако в случае гетерогенной сети ни одна из них системой управления сетью не является. Требуется система, работающая «поверх» систем управления фирм-производителей. Напрямую с оборудованием она взаимодействовать не будет. Каждая из систем управления фирм-производителей должна предоставить *стандартизированный* интерфейс управления для управления своим участком сети.

Такой интерфейс взаимодействия в концепции TMN называется Q-интерфейсом. Первоначальные версии рекомендаций TMN предписывали использовать для его реализации протокол CMIP [10], а для описания моделей управления — языки GDMO/ASN.1 [11, 8]. Однако сложности реализации систем управления, основанных на этих средствах, вынудили производителей искать альтернативные решения [2]. Одним из таких решений стал протокол SNMP [5], успешно применяемый для управления IP-сетями.

Решение под названием SMART TMN [15], выдвинутое форумом TeleManagement Forum, предполагало использование более широкого круга технологий. В частности, для реализации транспортно-го уровня было предложено использовать объектную технологию CORBA [12]. Поскольку к тому моменту уже существовали системы управления, использующие протоколы CMIP и SNMP, для решения задач интеграции были выработаны спецификации организации взаимодействия между ними [13].

Кроме протокола удаленного взаимодействия, для Q-интерфейса должна быть определена информационная модель управления. Стандартные информационные модели управления приведены в рекомендациях МСЭ-Т, МСО, Европейского института телекоммуникационных стандартов (ETSI), форумов TeleManagement Forum, ATM Forum и др. В частности, в нашем проекте применялся набор стандартов TeleManagement Forum MTNM v2.1 [16,17, 18]. Правильность выбора подтверждается тем, что этот стандарт поддерживают системы управления таких производителей, как Cisco, Siemens, Hewlett-Packard и др.

## 2. Технологии разработки

Следующей задачей после выбора технологий сетевого управления стал выбор технологий разработки системы. Необходимо было выбрать платформу разработки, поддерживающую CORBA. Наиболее распространенные платформы, поддерживающие создание CORBA-приложений, — это C++ и Java. Выбор между ними является скорее бизнес-решением, чем техническим. Нами была выбрана Java, поскольку для нее имелось технологическое решение, обещающее сделать разработку проще (об этом ниже).

Далее предстояло выбрать подходящую реализацию служб CORBA. Мы остановили свой выбор на пакете JacORB<sup>1</sup>, разработанном сотрудниками Freie Universität Berlin совместно с Xtradyne Technologies AG. Фактически нам нужны были реализации служб имен и уведомлений. Последняя необходима для получения сообщений о возникающих авариях от системы управления элементами.

Для ускорения процесса разработки было решено воспользоваться CASE-пакетом REAL [4] и технологическим решением REAL-IT [3]. REAL-IT включает в себя методику использования моделей REAL для описания информационной системы, набор генераторов для создания баз данных Microsoft Access / SQL Server, а также общую архитектуру построения приложений и набор библиотек динамической поддержки.

В качестве СУБД было выбрано самое простое решение — Microsoft Access. От создаваемой системы не требовалась исключительная надежность, а для резервирования данных планировалось использовать зеркалирование жесткого диска средствами ОС Microsoft Windows 2000 Server.

## 3. Анализ требований заказчика

На этапе анализа требований мы старались, во-первых, определить, какая функциональность необходима заказчику в дополнение к «стандартной» (буквально: определенной стандартами), а во-вторых, какую «стандартную» функциональность нам, напротив, реализовывать не нужно.

Главным «дополнительным» требованием заказчика было наличие возможности как можно проще управлять соединениями между пользователями сети. Требовалось создать вариант рабочего места,

---

<sup>1</sup><http://www.jacorb.org/>

предназначенного исключительно для соединения абонентов сети друг с другом. Для того чтобы пользоваться таким рабочим местом, от оператора не требуется никаких специальных знаний.

Кроме упрощенных рабочих мест, заказчик требовал наличия одного полнофункционального, с помощью которого должно было осуществляться управление коммутацией в сети и мониторинг аварий. Кроме того, на нем подготавливались данные для использования на упрощенном рабочем месте: названия абонентов сети, их группировка и назначение прав доступа.

Таким образом, из всей функциональности, определенной в стандарте ТМФ 513 [16], нам потребовались следующие наборы функций:

- подключение к системе управления элементами;
- запрос состава и структуры сети;
- управление коммутацией (трассы «точка-точка»);
- мониторинг аварий.

Сужение набора реализуемой функциональности позволило снизить затраты на разработку.

#### 4. Проектирование системы

За основу были взяты архитектура «клиент-сервер» и трехслойная архитектура построения информационных систем. Сервер системы управления отвечает за предоставление доступа к БД, взаимодействие с системами управления элементами по CORBA-интерфейсу и реализует бизнес-логику. Клиент реализует графический пользовательский интерфейс. Для взаимодействия клиента и сервера применяется протокол RMI. Архитектура системы представлена на рис. 1.

Структура базы данных сервера содержит:

- объектно-реляционное отображение структуры объектов стандарта ТМФ 814 [18];
- таблицы для хранения параметров пользователей сети;
- таблицы журналов;
- данные операторов системы управления.

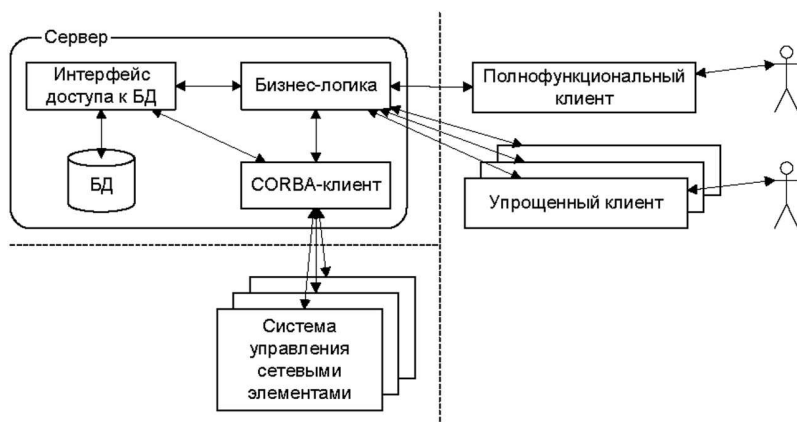


Рис. 1. Архитектура системы управления сетью

Схема базы данных была разработана в пакете REAL, после чего сама база данных генерировалась автоматически. Кроме того, по той же схеме данных пакетом REAL-IT был сгенерирован код большинства графических форм системы.

## 5. Проектирование пользовательского интерфейса

При проектировании графического пользовательского интерфейса (GUI) традиционно рассматриваются два независимых друг от друга аспекта: функциональность и его внешний вид. В соответствии с терминологией, применяемой в Рек. МСЭ-Т М.3010 [7], первый аспект относится к так называемому F-интерфейсу, а второй — к G-интерфейсу.

### 5.1. Требования к функциональности и внешнему виду GUI

Требования и рекомендации, применявшиеся нами при проектировании GUI, условно могут быть разбиты на три категории:

- общие рекомендации по проектированию GUI — универсальные рекомендации, применяемые без учета специфики предметной области, в основном относящиеся к внешнему виду GUI;



- рекомендации, учитывающие специфику телекоммуникационной области, — здесь присутствуют как функциональные требования, так и требования к внешнему виду;
- требования заказчика — в основном функциональные требования.

Общие рекомендации по проектированию касаются удобства работы пользователя и следования интерфейса стилю, диктуемому применяемой операционной системой, что в конечном итоге также следует из требования обеспечить комфортную рабочую среду. Для разработчиков графических приложений с использованием платформы Sun Java основополагающим руководством является книга [14]. Она описывает стандартные, рекомендуемые для всех Java-приложений, внешний вид и поведение.

Функциональные требования, относящиеся к специфике телекоммуникационного программного обеспечения, разделяются по областям FCAPS. Ниже мы перечислим требования, относящиеся к каждой из областей.

- Управление неисправностями:
  - выдача сообщений о регистрируемых авариях;
  - ведение журнала аварий;
  - подтверждение оператором ознакомления с аварийным сообщением;
  - управление тестированием нами не рассматривалось, так как данная функциональность возложена на систему управления элементами.
- Управление конфигурацией:
  - запрос конфигурации сети;
  - получение уведомлений об изменениях в конфигурации сети;
  - управление представлением элементов сети;
  - управление конфигурацией элементов сети не рассматривалось, так как эта функциональность также возложена на систему управления элементами.

- Управление расчетами за услуги не было необходимо рассмотреть, так как заказчик не предъявлял такие требования.
- Управление рабочими характеристиками и производительностью:
  - ведение журнала статистики.
- Управление безопасностью:
  - авторизация доступа;
  - управление пользователями системы;
  - ведение журнала действий оператора.

Особенности, связанные с проектированием GUI систем сетевого управления, указаны в рекомендации EG 201 024 европейского института ETSI [6]. Эти особенности связаны с тем, что, во-первых, управление сетями связи происходит в режиме реального времени, а во-вторых, управляемая среда является гетерогенной. Таким образом, в системе возможно возникновение большого количества событий, о которых необходимо своевременно оповестить оператора. Как следствие, возникает необходимость во введении приоритетов для сообщений и возможностей по их фильтрации. Кроме того, необходимо свести к минимуму усилия оператора по поиску необходимой ему информации.

Гетерогенность управляемой среды выражается в том, что в ней могут присутствовать сетевые элементы различного назначения, обладающие различными возможностями, что вносит дополнительные трудности в процесс управления сетью.

Специфика требований к внешнему виду систем сетевого управления заключается в необходимости использовать собственное метафорическое представление управляемых элементов и задач оператора. Например, существуют рекомендуемые к использованию символы для обозначения элементов сети, а также соглашения об используемых цветах. Кроме того, в [6] рекомендуется группировать пункты меню в соответствии с функциональными областями, перечисленными выше.

Еще одна особенность интерфейса сетевого управления — наличие у управляемых объектов такого признака, как географическое

местоположение. Таким образом, возникает необходимость в привязке графических элементов, соответствующих управляемым объектам, к изображению географической карты.

Использование в процессе разработки технологии REAL-IT [3] также наложило некоторые требования на проектируемый интерфейс. Технология REAL-IT ориентирована на автоматизированную генерацию диалоговых форм нескольких видов:

- список записей — отображение свойств нескольких объектов;
- карточка — редактирование свойств одного объекта;
- форма-отношение — представление связей типа «многие-ко-многим».

При работе со списком записей оператору доступны операции фильтрации, сортировки и поиска записей, а также операции добавления, удаления и вызова редактора свойств записи. И в списке, и в карточке можно определить дополнительные операции над объектами, соответствующими записям.

Таким образом, операции в проектируемой системе управления нужно было выразить в терминах работы с записями. Однако это несложное условие дало нам преимущество — большое количество диалогов системы стало возможным полностью или частично сгенерировать автоматически, что ускорило процесс разработки системы.

## 5.2. Представление GUI

Анализируя доступные нам системы управления сетями, мы выделили несколько основных «инструментов» оператора:

- карта сети;
- иерархия сети;
- просмотр оборудования;
- журналы;
- справочники.

Схематично пользовательский интерфейс системы управления изображен на рис. 2.

В качестве фонового изображения карты сети обычно используется географическая карта области, в которой развернута сеть

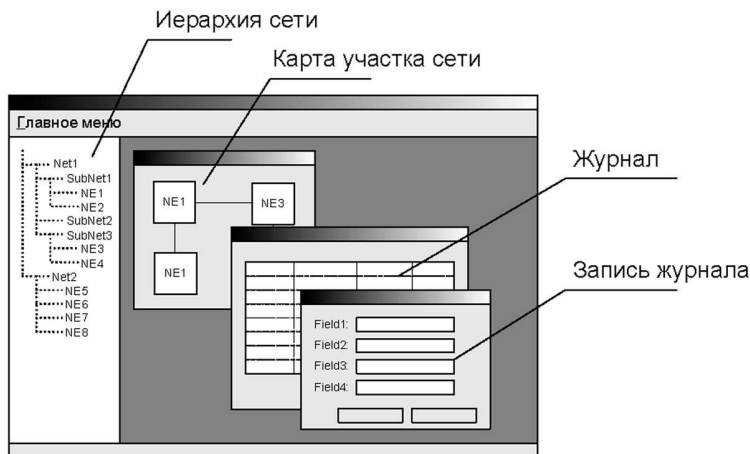


Рис. 2. Схема GUI системы управления

управления. Поверх фона отображается граф, узлами которого являются графические объекты, соответствующие элементам сети, а дугами — линии связи между ними. Оператору предоставляются возможности по изменению масштаба и уровня подробности отображения, а также по созданию новых окон для просмотра отдельных участков сети. Состояние сетевых элементов динамически отображается рядом или поверх узлов графа, что позволяет оператору отслеживать его в режиме реального времени.

Поскольку в информационной модели ТМФ 608 [17] у сетевых элементов нет атрибутов, соответствующих их географическому положению, система управления сетью, импортирующая конфигурацию сети из систем управления нижнего уровня, должна дать оператору возможность указать положение элементов. Таким образом, карта сети должна предусматривать два режима работы с сетевыми элементами. В первом режиме оператор должен иметь возможность свободно перемещать по экрану сетевые элементы или группы сетевых элементов, во втором режиме положения сетевых элементов должны быть фиксированы, чтобы предотвратить их случайное смещение.

Эти два режима могут быть выделены в системе явно, либо каждый из элементов должен обладать признаком того, что он закреплён, и его перемещать нельзя. Мы выбрали первый вариант

как более простой — в системе предусмотрены режимы редактирования карты и мониторинга состояния сети.

Иерархия сети представляет сеть в виде дерева, отражающего подчинение по управлению сущностей, составляющих сеть. Вот пример такой иерархии:

система управления → подсеть → сетевой элемент → оборудование

На дереве также может отображаться информация об авариях, возникающих на сетевых элементах. Вместо дерева можно использовать представление с использованием графа, как это рекомендуется в [6], однако этот способ более сложен с точки зрения реализации.

Режим просмотра оборудования отображает на экране стойки и этажи сетевого оборудования, позволяя оператору запрашивать состояние каждой платы. В нашей системе такой режим реализован не был — совместно с заказчиком было принято решение, что информацию о состоянии оборудования оператор может получить, обратившись к пользовательскому интерфейсу системы управления элементами сети.

Журналы используются для отображения информации об авариях, статистической информации и действиях оператора. Поскольку за время работы системы накапливается большое количество записей, журналы предоставляют средства для поиска, фильтрации, сортировки и архивации информации.

Справочники представляют собой электронную версию документации на систему. По сравнению с бумажной версией электронная обладает возможностями мгновенного вывода на экран нужного раздела справки по ссылке (контекстная подсказка) и поиска по ключевым словам.

## 6. Реализация

Как уже стало ясно из предыдущего изложения, часть кода системы была сгенерирована автоматически, а именно:

- скелетный код CORBA-клиента (приложением `idl2java` по IDL-файлам TMF 814);
- прикладной интерфейс доступа к БД (пакетом REAL-IT по схеме БД);

- код диалогов (им же).

Код форм схемы и дерева сети, а также главное окно приложения были написаны вручную, поскольку такие типы форм не поддерживаются пакетом REAL-IT. Для некоторых из автоматически сгенерированных форм пришлось провести ручную «доводку».

Поскольку одна из организаций-соисполнителей в процессе работы над проектом должна была реализовать код поддержки интерфейса MTNM для своей системы управления, возникла потребность в совместной отладке. Для обеспечения этого процесса пришлось организовать виртуальную частную сеть (VPN) между офисами ГУП «Терком» и организацией-соисполнителем. Как показала практика, такое удаленное взаимодействие хоть и приносит результаты, но является гораздо менее эффективным по сравнению с непосредственным взаимодействием. После того как был организован испытательный стенд, на котором присутствовали представители обеих организаций, процесс совместной отладки стал более продуктивным.

В процессе отладки совместной работы нашей системы управления с системой управления элементами обнаружилась проблема в стыковке нашего CORBA-клиента, написанного на Java, с CORBA-сервером системы управления элементами, написанным на Borland C++, ORB VisiBroker. Java при работе с потоками ориентируется на big-endian представление данных, в то время как VisiBroker, как более близкая к платформе PC система, использует little-endian представление. По спецификации CORBA способ представления данных указывается в заголовке пакета GIOP, и функции маршallingа должны осуществлять соответствующее перекодирование данных. Однако при обработке рекурсивных определений TypeCode перекодирование не производилось, в результате не работал обмен значениями типа any. К счастью, в силу доступности исходных кодов библиотеки Java и JacORB, указанную проблему удалось разрешить.

## Заключение

Из проделанного анализа опыта создания системы управления можно заключить, что разработка подобных систем является достаточно трудоемкой задачей, что и обуславливает высокую стоимость готовых систем управления телекоммуникационными сетями. В то

же время не всем заказчикам необходима обширная функциональность, предоставляемая системами управления от ведущих производителей. Разработка систем управления «под заказ» позволяет, во-первых, исключить ненужную заказчику функциональность для упрощения системы, а во-вторых, реализовать необходимую только ему функциональность.

Кроме того, при разработке даже достаточно простой системы управления сетью рекомендуется применять средства автоматической генерации кода, поскольку это избавляет от решения множества рутинных задач.

Подтверждением успешности нашего опыта по разработке может служить то, что созданная система успешно прошла государственные испытания и установлена на действующую сеть.

К недостаткам реализованной системы можно отнести излишнюю простоту используемой СУБД, что приводит к ухудшению надежности и быстродействия системы управления. Использование промышленной СУБД и реализация бизнес-логики системы на базе сервера приложений позволит удовлетворить более жесткие требования к рабочим характеристикам системы управления.

## Список литературы

- [1] *Дымарский Я. С., Крутякова Н. П., Яновский Г. Г.* Управление сетями связи: принципы, протоколы, прикладные задачи. — М.: Связь и бизнес. — 2003. — 382 с.
- [2] *Ерохин А. В., Корнев Н. А.* TMN: надежда и реальность альтернативных подходов. // Вестник связи. — № 4. — 2001. — С. 93–99.
- [3] *Иванов А. Н.* Технологическое решение REAL-IT: создание информационных систем на основе визуального моделирования // Системное программирование. — СПб. — 2004.
- [4] *Терехов А. Н., Романовский К. Ю., Кознов Д. В., Долгов П. С., Иванов А. Н.* Real: методология и CASE-средство для разработки информационных систем. // Программирование. — № 5. — 1999.
- [5] *Case J., Fedor M., Schoffstall M., Davin J.* Simple Network Management Protocol (SNMP). — RFC 1157. — 1990.
- [6] ETSI EG 201 024. Human Factors (HF); User Interface Design Principles for the Telecommunications Management Network (TMN) Applicable to the «G» Interface. — European Telecommunications Standards Institute. — 1997. — 31 p.
- [7] ITU-T Recommendation M.3010. Principles for a Telecommunications Management Network. — 2000. — 40 p.

- [8] ITU-T Recommendations X.680–X.693. Information Technology — Abstract Syntax Notation One (ASN.1) & ASN.1 Encoding Rules. — 2002.
- [9] ITU-T Recommendation X.700. Management Framework for Open System Interconnection (OSI) for CCITT Applications. — 1992. — 11 p.
- [10] ITU-T Recommendation X.711. Information Technology — Open Systems Interconnection — Common Management Information Protocol: Specification. — 1997. — 40 p.
- [11] ITU-T Recommendation X.722. Information Technology — Open Systems Interconnection — Structure of Management Information: Guidelines for the Definition of Managed Objects (GDMO). — 1992. — 50 p.
- [12] Object Management Group. Common Object Request Broker Architecture: Core Specification, Version 3.0.3. — OMG Document Formal/04-03-12. — 2004. — 1152 p.
- [13] Object Management Group. Interworking Between CORBA and TMN Systems Specification, Version 1.0. — OMG Document Formal/00-08-01. — 2000. — 298 p.
- [14] Sun Microsystems, Inc. Java Look and Feel Design Guidelines, Second Edition. — Pearson Education. — 2001. — 398 p.
- [15] TeleManagementForum GB 909. SMART TMN. Technology Integration Map. — Issue 1.1. — 1998. — 59 p.
- [16] TeleManagementForum TMF 513. Multi-Technology Network Management Business Agreement. Version 2.1. — 2002. — 359 p.
- [17] TeleManagementForum TMF 608. Multi-Technology Network Management Information Agreement. Version 2.1. — 2002. — 219 p.
- [18] TeleManagementForum TMF 814. Multi-Technology Network Management Solution Set. Version 2.1. — 2002. — 15 p.