

<тут будет титульный лист>

<тут будет техническое задание>

Аннотация

Русский

Работа рассматривает текущее положение дел в области цифровых библиотечных систем. Обозреваются форматы хранения и передачи библиографических данных. Проводится сравнение сетевых протоколов, используемых библиотечными системами. Даются рекомендации по доработке системы ГПНТБ «Веб-ИРБИС» с прицелом на формирование открытой библиотечной сети.

Английский

The paper deals with actual standards for digital libraries. Bibliographic data storage and transmission formats are considered. Comparison for library network protocols is done. Recommendations for upgrade of GPNTB «Web-IRBIS» library software are given to establish transparent interlibrary communication.

Оглавление

1	Постановка задачи.....	5
2	Обзор литературы.....	7
3	Стандарты.....	8
3.1	Международные стандарты идентификации объектов в библиотечном деле.....	8
3.2	Стандартный формат обмена библиографическими данными MARC.....	9
3.3	Форматы электронных сообщений, связанных с межбиблиотечным обменом.....	11
3.4	Форматы команд для поиска в библиографических базах данных ISO-SR.....	13
3.5	Определение прикладной услуги информационного поиска и спецификация протокола ANSI Z39.50.....	14
3.6	Стандарты представления электронных документов.....	16
3.7	Форматы электронного обмена полными текстами документов GEDl.....	16
4	Форматы описания данных.....	17
4.1	MARC.....	17
4.2	MARCXML.....	29
4.3	MODS.....	33
5	Сетевые протоколы.....	36
5.1	HTTP.....	36
5.2	Z39.50.....	40
6	Система Web-IRBIS.....	52
6.1	Общие сведения.....	52
6.1.1	Основные системные характеристики Web-ИРБИС.....	52
6.1.2	Базовые операции Web-ИРБИС.....	53
6.1.3	Основные программно-технологические компоненты.....	57
6.2	Особенности работы Web-ИРБИС.....	59
7	Результаты.....	60
7.1	Основные различия между Z39.50 и HTTP.....	60
7.2	Возможности построения информационных систем на базе Z39.50 и HTTP.....	61
8	Заключение.....	64
8.1	Выводы.....	64
8.2	Перспективы.....	64
9	Список литературы.....	66

1 Постановка задачи

Фраза «Кто владеет информацией — тот владеет миром» не потеряла своей актуальности и 2 века спустя после своего появления. Сегодня как никогда очевидна ценность информации, и роль ее будет все более увеличиваться по мере развития человеческого общества. Следует также вспомнить поговорку «предупрежден — значит вооружен». Не вызывает сомнений, что построение успешных процесса, организации, государства наконец, требует огромного количества достоверных, точных и своевременных данных из множества источников. В этой связи ключевую роль играют хранилища информации — библиотеки. Они содержат все, что человечество создало за века своего существования и посчитало полезным для будущих поколений.

Библиотеки прошли огромный путь от стопки глиняных табличек за тысячи лет до нашей эры до современных цифровых сред. Цифровой способ хранения информации обеспечивает ряд преимуществ по сравнению с аналоговым, таких как: компактность, удобство и скорость поиска, относительная дешевизна обслуживания, легкость копирования. Развитие электронных коммуникаций также добавило возможность одновременного доступа к одному материалу из множества точек на планете.

Тем не менее, современные электронные библиотеки не лишены и недостатков. В связи с неравномерной скоростью развития науки и технологий в мире на данный момент находятся в эксплуатации десятки различных несовместимых друг с другом библиотечных систем. При этом хранящиеся в них данные подчас уникальны. Это заставляет искать способы скопировать информацию либо обеспечить так называемые «шлюзы» - каналы для доступа из других систем.

Целью данной работы является разработка архитектуры интеграции

библиотечной системы СПбГЭТУ «ЛЭТИ» Web-IRBIS с другими библиотечными системами.

Необходимо, в частности:

- исследовать стандарты в области электронных библиотечных систем
- выполнить обзор форматов описания библиотечных данных
- выполнить обзор библиотечных сетевых протоколов
- провести анализ архитектуры системы Web-IRBIS
- разработать модернизированную архитектуры для Web-IRBIS

2 Обзор литературы

Среди русскоязычных источников наиболее заметными являются публикации Политехнического университета Санкт-Петербурга: работы [1] и [2] посвящены протоколам поиска и передачи библиографической информации. Заслуживают внимания публикации [3], рассматривающие опыт построения открытой библиотечной сети на базе ГПНТБ.

Среди англоязычных источников нужно выделить сайт Библиотеки Конгресса США [5], на котором можно получить исчерпывающее представление о господствующих стандартах в области цифровых библиотек, а также. С инициативами Европейского библиотечного консорциума можно ознакомиться на интернет-сайтах [11] и [12].

Детали работы системы «Веб-ИРБИС» подробно рассмотрены в [9] и [10].

3 Стандарты

3.1 Международные стандарты идентификации объектов в библиотечном деле.

Для идентификации книг, периодических изданий, аудиовизуальных материалов и электронных записей, а также издательств и библиотек, приняты международные стандарты, которые ведут технические комитеты ISO TC 46 / SC9 и ISO / IEC JTC1 / WG4 [4]

- ISO 2108:1992. Information and documentation. International Standard Book Numbering (ISBN);
- ISO 3297:1986. Documentation - International Standard Serial Numbering (ISSN);
- ISO 3901:1986. Documentation - International Standard Recording Code (ISRC);
- ISO 10444:1994. Information and documentation. International Standard Technical Report Number (ISRN);
- ISO 10957:1993. Information and documentation. International Standard Music Number (ISMN);
- ISO 15511 International Standards Library and organization Identifier (ISLOI);
- ISO/IEC 9070:1991. Information technology - SGML support facilities - Registration procedures for public text owner identifiers.

Для идентификации периодических изданий и отдельных публикаций (статей) в этих изданиях также применяется американский стандарт ANSI / NISO Z39.56-1991. Serial Issue and Contribution Identifier (SICI). Стандарт SICI в первоначальной версии широко используется на уровне идентификации изданий во многих библиотечных системах мира как важный элемент

сообщений электронного обмена данными.

Существенная ревизия этого стандарта, выполненная в последние годы, обеспечила возможность широкого его применения в электронном обмене данными с использованием среды Интернет (унифицированные имена ресурсов). В 1996-97 гг. ассоциация американских издателей (AAP-Association of American Publishers) представила проект, названный DOI (Digital Object Identifier), который обеспечивает управление защитой авторских прав при обмене электронной информацией. Идентификатор, установленный для некоторого информационного ресурса, позволяет пользователю направить запрос к центральной базе данных DOI. Запрос передается издателю, который возвращает пользователю полную запись ресурса, включающую как библиографическую информацию, так и сведения о статусе копирайта этого ресурса.

Вопрос полного или частичного применения указанных выше международных стандартов в российских библиотеках выходит за пределы настоящего документа. Отметим только, что решения по этому вопросу, принятые соответствующими федеральными органами РФ, должны быть приняты как исходные при формировании профилей ЭБ, касается правил идентификации объектов.

3.2 Стандартный формат обмена библиографическими данными MARC

Стандартный формат обмена библиографическими данными MARC (Machine Readable Cataloguing) является основой для формирования электронных каталогов библиотек и библиографических баз данных.

Стандарты на формат обмена MARC поддерживаются техническим комитетом ISO TC46 / SC4, международной ассоциацией библиотек IFLA, Библиотекой Конгресса США, Британской библиотекой, Национальной

библиотекой Канады, Постоянным комитетом UNIMARC. Формат MARC широко принят для обмена библиографическими данными в разных странах. В частности, его используют 69% библиотечных систем в Европе.

Исторически сложились разные версии формата MARC: американская USMARC, разработанная для Библиотеки конгресса США, и английская UKMARC, разработанная для Британской библиотеки. В настоящее время проводится работа, направленная на то, чтобы принять USMARC в расширенной форме и для Великобритании.

Разработан международный формат UNIMARC, который поддерживает Permanent UNIMARC Committee. Недавно UNIMARC был выбран многими европейскими библиотеками для обмена каталожными данными, а также Консорциумом Европейских исследовательских библиотек (CERL) для формирования базы данных Hand Printed Books, объединяющей данные из разных библиотек Европы.

Форматы MARC для обмена библиографическими данными определены следующими документами:

- The Anglo-American Cataloging Rules (1994). Ottawa: Canadian Library Association; London: Library Association Publishing; Chicago: American Library Association;
- ISBD(G) General International Standard Bibliographic Description (1992) prepared by IFLA Committee on cataloging;
- The UKMARC Manual: a Cataloguer's guide to the bibliographic format (1996);
- UNIMARC Manual (1994) IFLA Universal Bibliographic Control and International MARC Programme, British Library Bibliographic Services;
- USMARC Format for Bibliographic Data. Including Guidelines for Content

Destination (1994), ISBN 0-8444-0809-3.

База данных OCLC (Online Computer Library Center), в которой применен формат USMARC, используется более чем 20000 библиотеками в 61 стране. Она содержит в настоящее время более 30 млн. библиографических записей на 400 языках. Еженедельно в нее добавляются более 34 тысяч записей по монографиям, периодическим изданиям, аудио видео материалам, картам, архивам, рукописям, компьютерным файлам. Эти записи оформляются в соответствии с англо-американскими правилами каталогизации (AACR2), принятыми в 1978 г. и дополненными в 1993 г.

OCLC является результатом кооперированной деятельности по каталогизации Библиотеки Конгресса США, Британской библиотеки, Национальной библиотеки США по медицине, Национальной библиотеки Канады и Национальной библиотеки Австралии.

При формировании библиотечного сектора в регионах России следует учитывать обеспечение возможности доступа к OCLC, как к мировой базе библиографических данных.

3.3 Форматы электронных сообщений, связанных с межбиблиотечным обменом

Форматы электронных сообщений, связанных с межбиблиотечным обменом (по межбиблиотечным абонентам) установлены международными стандартами, поддерживаемыми Техническим комитетом ISO TC46 / SC4:

- ISO 10160:1997. Information and documentation - Open Systems Interconnection. Interlibrary Loan Application Service Definition;
- ISO 10161:1997. Information and documentation- Open Systems Interconnection. Interlibrary Loan Application Protocol Specification. Part1: Protocol Specification. Part2: Protocol implementation conformance Statement proforma.

Стандарт ISO 10160 определяет спецификации сообщений по межбиблиотечному обмену (ILL-Inter Library Loan), касающихся запросов на требуемые читателям издания, сведений об отсылке затребованных книг, ответов на запросы и т.д. Стандарт 10161 определяет в нотации ASN.1 спецификации протоколов телекоммуникационной среды, необходимых для обмена сообщениями в архитектуре OSI.

Для использования этих двух стандартов определены международные профили OSI-ISP ALD 2n Library and Documentation Interlibrary Loan Store-and-Forward Protocols. Часть 4 (ALD22) этого профиля специфицирует применение протокола электронной почты X-400 для передачи сообщений, связанных с межбиблиотечным обменом.

В качестве альтернативы возможно трактовать ILL-сообщения как EDIFACT-сообщения.

Следующие проекты в области электронного межбиблиотечного обмена, поддерживаются Европейской комиссией:

- DALI, обеспечивающий услуги для поставки мультимедийных документов в распределенной среде, использующей протоколы ISO-ILL;
- UNIVERSE, исследовательский проект реализации ILL;
- ION OSI, пилотный проект европейских библиотечных сетей, Великобритании, Нидерландов и Франции, базирующийся на подмножестве ILL - протоколов и протоколе поиска в библиографических базах данных ISO-SR (см. ниже), реализованных на платформах с ОС типа UNIX;
- AIDA (Alternatives for International Document Availability).

3.4 Форматы команд для поиска в библиографических базах данных ISO-SR

Международные стандарты, поддерживаемые Техническим комитетом ISO TC46 / SC4, определяют команды для поиска в библиографических базах данных с использованием услуг телекоммуникационной среды по протоколам OSI:

- ISO 8777:1993. Information and documentation - Commands for interactive text searching;
- ISO 10162:1993. Information and documentation- Open Systems Interconnection. Search and Retrieve Application Service Definition;
- ISO 10163. Information and documentation - Open Systems Interconnection. Search and Retrieve Application Protocol Specification. Part1 (1993): Protocol Specification.

Стандарт ISO 8777 определяет набор из 30 поисковых команд вместе с 8 специальными символами, которые могут использоваться, чтобы квалифицировать эти команды.

Стандарт ISO 10162 описывает, как услуги поиска, отвечающие заданным правилам, специфицируются на прикладном (седьмом) уровне OSI (по ISO 7498) и могут быть представлены протоколами ACSE нижележащих уровней OSI (по ISO 8649). Стандарт определяет услуги поиска и замещения (SR-search and replace) для библиотек, информационных служб и каталожных центров, используя модель "клиент-сервер".

Стандарт ISO 10163 определяет спецификации служб (протоколов среды), необходимых для поддержки услуг поиска, соответствующих ISO 10162.

Для использования стандартов ISO 10162, 10163 определены профили OSI-ISP ALD1n Library and Documentation Search and Retrieve Specifications.

ISO 10162 используется как подмножество протокола Z39.50 (см. ниже).

Форматы ISO-SR применены в проекте Europe Gate, выполняемом в рамках Европейской библиотечной программы. Программное обеспечение, реализующее клиентскую часть протокола ISO-SR (Z39.50 V2) разработано как часть проекта SOCKER, выполненного при поддержке Европейского сообщества. Серверная часть протокола реализует услуги, обеспечивающие доступ к онлайн-публичным каталогам. Они разрабатываются в европейском проекте ARCA (Access to Remote Catalogues).

Форматы ISO-SR применены также в проекте OPAC Network in Europe (ONE), который должен обеспечить клиентам библиотек лучший доступ к онлайн-публичным каталогам (OPAC-Online Public Access Catalogues) и другим национальным каталогам. Некоторые из участников проекта уже предлагают услуги, базирующиеся на применении протоколов Z39.50 и / или ISO-SR.

3.5 Определение прикладной услуги информационного поиска и спецификация протокола ANSI Z39.50

Стандарты, поддерживаемые ANSI и US National Information Standards Organization (NISO), распространяются на услуги поиска в библиографических базах данных, совместимые с протоколами OSI. К этим стандартам относятся:

- ANSI / NISO Z 39.50-1992. Information Retrieval Application Service Definition and Protocol Specification for Open Systems Interconnection (revision and redesignation of ANSI Z 39.50-1988);
- ANSI / NISO Z 39.50-1995. Information retrieval (Z 39.50): Application service definition and protocol Specification (revision of ANSI Z 39.50-1992). По содержанию идентичен ISO 23950;

- ISO 23950:1998. Information and documentation - Information retrieval (Z 39.50). Application service definition and protocol Specification.

Протокол Z 39.50 на прикладном (седьмом) уровне OSI специфицирует запросы поиска информации в онлайн-библиотечных каталогах и других структурированных наборах данных. ANSI Z 39.50 усиливает правила ISO, определенные стандартами ISO 10162, 10163, и представляет множество поисковых опций для использования в OSI-совместимых приложениях. После того, как ISO 10162 был реализован, для американских библиотек потребовались дополнительные функции просмотра и управления доступом. В связи с этим в ISO был представлен стандарт ISO 23950.

Для поиска информации в библиографических базах данных введен профиль ATS-1 стандарта Z 39.50, который специфицирует использование набора атрибутов BIB-1.

В США стандарт Z 39.50 составляет основу для Government Information Locator Service (GILS). Официальным органом, ведущим регистрацию реализаций Z 39.50 в мире, является Библиотека Конгресса США.

В Европе на Z 39.50 базируется проект OPAC Network in Europe (ONE), обеспечивающий клиентам библиотек доступ к онлайн-публичным каталогам. В рамках Европейской библиотечной программы протоколы Z 39.50 реализуются в проектах SOCKER и ARCA.

В связи с тем, что во многих случаях телекоммуникационную среду определяет стек протоколов TCP / IP, в Европе разрабатывается проект Paragon, финансируемый Европейским сообществом. Цель этого проекта - обеспечить клиентам библиотек простые шлюзы к электронным каталогам и архивам мультимедиа-информации через WWW. Одновременно Европейское

сообщество финансирует проект EUROPA Gate, в котором обеспечивается шлюз между Z 39.50 и WWW и пользователями электронной почты через Internet.

Профили для применения Z 39.50 в Европе разрабатываются Европейским Форумом Реализаторов библиотечных приложений (EFILA) совместно с IFOBS и международной группой реализаторов Z 39.50 (ZIG).

3.6 Стандарты представления электронных документов

Технический комитет ISO TC 46 / SC9 разработал стандарт ISO 12083. Electronic manuscript preparation and markup, в который сведены требования к структуре документов и к связанным с ними библиографическим данным, которые ранее действовали как разрозненный набор стандартов ISO, выпущенных в 70-80-ых годах. Стандарт ISO 12083 поддерживается Техническим Комитетом ISO TC 46 / SC4, ANSI и Ассоциацией американских издателей (AAP). Этот стандарт содержит стандартизованный набор тегов разметки языка SGML и определение типа документа (DTD -Document Type Definition) применительно к книгам и статьям, а также к периодическим изданиям.

3.7 Форматы электронного обмена полными текстами документов GEDI

Форматы и протоколы электронного обмена полными текстами документов поддерживаются Техническим Комитетом ISO TC 46 / SC4 WG4.

Они определены соглашениями GEDI (Group on Electronic Data Interchange Agreements), которые касаются двух аспектов:

- определение формата электронного документа;
- описание механизма обмена.

Формат GEDI состоит из информации дескриптивного заголовка, за

которым следует образ документа. В первой версии формат GEDI содержал ограниченный набор дескрипторов для описания образа документа в формате TIFF и специфицировал механизм FTAM для обмена.

Во второй версии набор дескрипторов был расширен так, чтобы связать элементы данных между GEDI и ISO ILL, наряду с FTAM был введен как механизм обмена протокол FTP.

Третья версия GEDI, разработанная в 1997 году, предусматривает следующие форматы документов:

- TIFF;
- PDF;
- JPEG;
- Post Script.

В качестве механизма обмена введен протокол электронной почты MIME, принятый для Интернет, и удален FTAM. Соглашения GEDI широко используются в Европе, США и Австралии для поддержки национальных и международных служб доставки документов в библиотечном мире.

В Европе возможности использования соглашений GEDI с учетом многонационального аспекта европейских стран демонстрирует проект EDIL, финансируемый Европейским сообществом.

4 Форматы описания данных

4.1 MARC

Стандартный формат обмена библиографическими данными MARC (Machine Readable Cataloguing) является основой для формирования электронных каталогов библиотек и библиографических баз данных. [6]

Стандарты на формат обмена MARC поддерживаются техническим комитетом ISO TC46 / SC4, международной ассоциацией библиотек IFLA, Библиотекой Конгресса США, Британской библиотекой, Национальной библиотекой Канады, Постоянным комитетом UNIMARC. Формат MARC широко принят для обмена библиографическими данными в разных странах. В частности, его используют 69% библиотечных систем в Европе.

Исторически сложились разные версии формата MARC: американская USMARC, разработанная для Библиотеки конгресса США, и английская UKMARC, разработанная для Британской библиотеки. В настоящее время проводится работа, направленная на то, чтобы принять USMARC в расширенной форме и для Великобритании.

Разработан международный формат UNIMARC, который поддерживает Permanent UNIMARC Committee. Недавно UNIMARC был выбран многими европейскими библиотеками для обмена каталожными данными, а также Консорциумом Европейских исследовательских библиотек (CERL) для формирования базы данных Hand Printed Books, объединяющей данные из разных библиотек Европы.

Форматы MARC для обмена библиографическими данными определены следующими документами:

- The Anglo-American Cataloging Rules (1994). Ottawa: Canadian Library Association; London: Library Association Publishing; Chicago: American Library Association;
- ISBD(G) General International Standard Bibliographic Description (1992) prepared by IFLA Committee on cataloging;
- The UKMARC Manual: a Cataloguer's guide to the bibliographic format (1996);
- UNIMARC Manual (1994) IFLA Universal Bibliographic Control and

International MARC Programme, British Library Bibliographic Services;

- USMARC Format for Bibliographic Data. Including Guidelines for Content Destination (1994), ISBN 0-8444-0809-3.

База данных OCLC (Online Computer Library Center), в которой применен формат USMARC, используется более чем 20000 библиотеками в 61 стране. Она содержит в настоящее время более 30 млн. библиографических записей на 400 языках. Ежедневно в нее добавляются более 34 тысяч записей по монографиям, периодическим изданиям, аудио видео материалам, картам, архивам, рукописям, компьютерным файлам. Эти записи оформляются в соответствии с англо-американскими правилами каталогизации (AACR2), принятыми в 1978 г. и дополненными в 1993 г.

OCLC является результатом кооперированной деятельности по каталогизации Библиотеки Конгресса США, Британской библиотеки, Национальной библиотеки США по медицине, Национальной библиотеки Канады и Национальной библиотеки Австралии.

В США стандартами для представления и обмена в машиночитаемой форме библиографическими, авторитетными, классификационными данными, данными о фондах, а также данными об общественной информации являются пять коммуникативных форматов USMARC: USMARC Format for Bibliographic Data (Формат USMARC для библиографических данных); USMARC Format for Authority Data (Формат USMARC для авторитетных данных); USMARC Format for Classification Data (Формат USMARC для классификационных данных); USMARC Format for Holdings Data (Формат USMARC для данных о фондах); USMARC Format for Community Information (Формат USMARC для общественной информации). Каждый отдельно опубликованный полный формат USMARC содержит подробное описание полей и инструкции по применению установленных

определителей содержания с примерами, а также идентифицирует правила, обеспечивающие последовательный ввод данных. Публикация “USMARC Concise Formats” предоставляет возможность быстрого наведения справок по определителям содержания каждого формата USMARC. В издании содержится сокращенное описание каждого поля, каждой позиции символов в полях с элементами данных фиксированной длины, а также определенных индикаторов в полях переменных данных. [В тексте используются термины “переменные поля”, “поля переменных данных”. Под этими понятиями подразумеваются поля данных переменной длины]. Описания кодов подполей и кодированных значений даются только тогда, когда их названия недостаточно информативны.

КОМПОНЕНТЫ ЗАПИСИ USMARC

В настоящем введении приводятся характеристики USMARC, общие для всех форматов. Специфическая информация о форматах дается в конкретных введениях к каждому формату.

Запись USMARC состоит из трех элементов: структуры записи, определения содержания и содержания данных записи. Структура записи является реализацией Американского национального стандарта для обмена информацией ANSI Z39.2 (American National Standard for Information Interchange). Определение содержания — коды и условные обозначения, установленные специально для идентификации и последующей характеристики элементов данных в записи, а также для манипулирования этими данными, — осуществляется в каждом формате USMARC. Содержание элементов данных, которые составляют запись USMARC, обычно устанавливается в соответствии со следующими стандартами: International Standard Bibliographic Description (ISBD) (Международный

стандарт на библиографическое описание), Anglo-American Cataloguing Rules, 2-nd ed.(AACR2) (Англо-американские правила каталогизации, 2-е изд.), Library of Congress Subject Headings (LCSH) (Предметные рубрики Библиотеки Конгресса), American National Standard for Serial Holdings Statements (ANSI Z39.44) (Американский национальный стандарт по сведениям о фондах сериальных изданий), Library of Congress Classification (LCC) (Классификация Библиотеки Конгресса) или другими регламентирующими документами, которые используются организацией, составляющей запись. В форматах USMARC определяется содержание конкретных кодированных элементов данных (например, Маркера, полей 007 и 008).

Запись USMARC состоит из трех основных компонентов: Маркера, Справочника и Переменных полей. Нижеприведенная информация обобщает структуру записи USMARC. Более детально она представлена в USMARC Specifications for Record Structure, Character Sets, Tapes (Спецификации USMARC для структуры записи, наборов символов, лент).

Маркер — Элементы данных, обеспечивающие информацию, необходимую для обработки записи. Элементы данных содержат цифры или кодированные значения и идентифицируются соответствующими позициями символов. Маркер имеет фиксированную длину 24 позиции символов и является первым полем записи USMARC.

Справочник — Ряд статей, содержащих метку, длину и начальную позицию каждого переменного поля внутри записи. Длина каждой статьи составляет 12 символов. Статьи справочника для переменных контрольных полей располагаются первыми в порядке возрастания цифровых меток. Следующие за ними статьи полей переменных данных располагаются в порядке

увеличения по отношению к первому символу метки. Расположение в записи полей переменных данных не обязательно должно быть идентичным порядку приведения соответствующих им статей справочника. Дублетные метки различаются только по месту расположения соответствующих полей в записи. Справочник заканчивается символом конца поля (ASCII 1E16).

Переменные поля — Данные в записи USMARC организованы в переменные поля. Каждое из них идентифицируется трехсимвольной цифровой меткой, которая хранится в статье Справочника для этого поля. Каждое поле заканчивается символом конца поля. Последнее переменное поле заканчивается символами как конца поля, так и конца записи (ASCII 1D16).

Имеется два типа переменных полей:

Переменные контрольные поля — Поля 00X. Переменные контрольные поля по своей структуре отличаются от полей данных переменной длины. Эти поля не содержат позиций индикаторов и кодов подполей. Они могут содержать как отдельный элемент данных, так и несколько элементов данных фиксированной длины, идентифицированных соответствующими позициями символов.

Поля переменных данных — Поля 0XX-8XX

В полях переменных данных используются следующие два вида определителей содержания:

Позиции индикаторов — Первые две позиции символов в начале каждого поля переменных данных, которые содержат значения, объясняющие или дополняющие данные в этом поле. Значения каждого индикатора рассматриваются самостоятельно,

то есть не приписываются двум индикаторам, взятым вместе. Значения индикаторов могут быть цифровыми или строчными буквенными символами. Пробел (ASCII 2016), представленный в этом документе как “b перечеркнутое “, используется в позиции неопределенного индикатора. В позиции определенного индикатора пробел может быть присвоенным значением или может обозначать отсутствие информации. [В данном переводе для обозначения пробела вместо “b перечеркнутого” используется знак # (хэш)].

Коды подполей — Два символа, предшествующие в поле каждому элементу данных, для которого требуется самостоятельная обработка. Код подполя состоит из разделителя (ASCII 1F16), представленного в этом документе как “перечеркнутый знак равенства = “, за которым следует идентификатор элемента данных. [В данном переводе для обозначения разделителя подполя вместо “перечеркнутого знака равенства =” используется знак \$]. Идентификаторы элементов данных обозначаются цифровыми или строчными буквенными символами. Коды подполей устанавливаются для каждого поля независимо от других полей, однако там, где это возможно, в разных полях предусматриваются параллельные значения. Коды подполей предназначены для идентификации данных, а не для установления порядка их расположения в записи. Порядок расположения подполей в целом определяется стандартами на содержание данных, такими как правила каталогизации.

ПОВТОРЯЕМОСТЬ ПОЛЯ И ПОДПОЛЯ

Теоретически все поля и подполя могут повторяться. Характер данных, однако, часто не допускает их повторения. Например, запись может

содержать только одно поле 1XX; библиографическое поле может содержать только одно подполе \$a (Имя лица), но может содержать более одного подполя \$c (Титулы (звания) и другие слова, ассоциируемые с именем). Повторяемость или неповторяемость поля или подполя указывается знаками (R) и (NR) [далее по тексту (П) и (НП)] после названия каждого поля или подполя в каждом формате USMARC.

Библиографические или авторитетные записи USMARC могут содержать данные как в латинской, так и в нелатинской графике. Специальные поля и подполя сигнализируют о наличии нелатинских символов в записи и содержат данные, представленные нелатинскими символами. Раздел “Наборы символов” в USMARC Specifications for Record Structure, Character Sets, Tapes содержит подробное описание методов, используемых для доступа к переключающимся наборам графических символов, и обозначение используемого в поле графического набора символов в записях USMARC.

Если в записи USMARC имеются данные в нелатинской графике, используются следующие определители содержания:

066 Используемые наборы символов — Подполе \$c (Идентификация альтернативного (переключающегося) набора графических символов) содержит коды, определяющие нелатинские наборы символов, используемые в записи.

01X-85X Поля переменных данных — Поля, содержащие только данные в латинской графике. Многие из этих полей могут ассоциироваться с полем 880, которое содержит те же данные в нелатинских символах. В каждом ассоциируемом поле подполе \$6 (Элемент связи) связывает это поле с соответствующим ему полем 880, которое также содержит подполе \$6. В альтернативной (переключающейся) графической форме могут быть

представлены только те ассоциированные поля, для которых предусмотрено подполе \$6.

880 Альтернативное (переключающееся) представление графических символов — Поле содержит данные в нелатинских символах, которые приводятся в ассоциируемом поле в латинских символах. Иногда в записи может приводиться поле с данными в нелатинской графике, не связанное с полем, содержащим данные в латинской графике.

СИМВОЛ-ЗАПОЛНИТЕЛЬ И ВЗАИМОСВЯЗАННЫЕ ЗНАЧЕНИЯ

Символ-заполнитель (ASCII 7C16), представленный в этом документе в виде вертикальной черты (|), может использоваться в записи, когда составитель записи принимает решение не применять код, определенный в спецификации формата. Символ-заполнитель не разрешается использовать в позициях символов Маркера или как компонент меток, позиций индикаторов или кодов подполей. Использование символа-заполнителя в записях, составляющих национальную базу данных, зависит также от требований национального уровня, установленных для каждого элемента данных в каждом формате USMARC.

Код u (Неизвестно), когда он определен, применяется для указания того, что составитель записи пытался присвоить код, но не мог подобрать адекватного значения кода.

Код n (Не применяется) предусмотрен во многих кодовых позициях. Он используется для указания того, что характеристика, определенная позицией, не применима для конкретной записи.

ДИСПЛЕЙНЫЕ КОНСТАНТЫ

Дисплейные константы могут быть термином, словосочетанием и/или интервалом или знаком пунктуации, которые при определенных условиях могут генерироваться системой, чтобы сделать визуальное представление данных в записи более удобным для пользователя. В формате для авторитетных данных, например, определенные метки полей (такие, как поля трассировок 4XX и 5XX), коды подполей (такие, как подполя тематических подразделов \$x, \$y и \$z в расширенном предметном заголовке), а также кодовые значения (такие, как контрольное подполе \$w/0 Специальная взаимосвязь) могут использоваться для генерирования специфических терминов, словосочетаний и/или интервалов или знаков пунктуации при выводе записи на дисплей. Предлагаемые константы и примеры имеются в каждом формате USMARC. Использование дисплейных констант определяется организацией или системой.

ПРАВИЛА РАСПРЕДЕЛЕНИЯ ОТВЕТСТВЕННОСТИ

Как правило, ответственность за содержание данных, определение данных, а также их транскрипцию в записи USMARC может устанавливаться на основе принципов, изложенных в разделе “Правила распределения ответственности”. Однако содержание определенных элементов данных ограничивается, если элемент данных составлен организацией-составителем записи или авторитетной организацией.

Правила распределения ответственности

В немодифицированных записях организация, идентифицированная в подполе \$a поля 040 (Источник каталогизации/Источник записи) как организация-составитель, является ответственной за содержание данных записи. Организация, идентифицированная в подполе \$c поля 040 как организация, осуществляющая перезапись, является ответственной за определение содержания и транскрипцию данных.

В модифицированных записях организации, идентифицированные в подполях \$a и \$d поля 040 (Служба модификации), несут коллективную ответственность за содержание данных записи. Организации, идентифицированные как организации, осуществляющие транскрипцию или модификацию данных в подполях \$c и \$d поля 040, несут коллективную ответственность за определение содержания и транскрипцию данных. (Составитель оригинальной записи может также быть идентифицирован в кодированной форме в поле 008/39 (Источник каталогизации) библиографических и авторитетных записей).

Элементы данных, установленные организацией-составителем записи

Элемент данных, определенный организацией-составителем, устанавливается только той организацией, которая несет за него ответственность, например, поле 010 (Контрольный номер Библиотеки Конгресса). Однако, когда определение содержания и перезапись данных, установленных организацией-составителем, выполняет другая организация, ответственность за эти функции определяется “Правилами распределения ответственности”.

Элементы данных авторитетной организации

Элемент данных авторитетной организации является элементом, содержание которого проставляется на основе перечня или набора согласованных правил, которые составляются и ведутся этой организацией.

Организация, применяющая элементы данных, установленные авторитетной организацией, должна пользоваться авторитетным перечнем или согласованными правилами, специфицированными для конкретного элемента данных, например, USMARC Code List for Geographic Areas в поле 043 (Код географического региона), схемами Library of Congress Classification

в поле 050 (Расстановочный шифр Библиотеки Конгресса). Применяемый перечень или согласованные правила специально идентифицируются в формате в описаниях элементов данных. Ответственность за содержание элементов данных авторитетной организации несут организации, использующие перечень или согласованные правила, что определено “Правилами распределения ответственности”.

ТРЕБОВАНИЯ НАЦИОНАЛЬНОГО УРОВНЯ (ТНУ)

Библиографические и авторитетные записи USMARC, создаваемые для национальной базы данных, должны соответствовать требованиям к содержанию, которые определяются в настоящем документе кодами в графе Формат/ТНУ перечня определителей содержания для каждого поля. Кодами требований национального уровня являются:

О — Обязательный

Код О — “обязательный” означает, что элемент данных должен быть представлен в полной записи национального уровня. Символ-заполнитель (|) не допускается применять для обязательных элементов данных.

У — Обязательный, если применяется (условно — обязательный)

Код У — “обязательный, если применяется” означает, что элемент данных должен быть представлен, если он необходим и о нем имеется информация. Символ-заполнитель не допускается в элементах, которые обозначены как “обязательные, если применяются”.

Ф — Факультативный

Код Ф — “факультативный” означает, что использование

элемента данных является необязательным. Символ-заполнитель разрешается применять для факультативных элементов данных полей 007 и 008, подполя \$7 библиографических полей 760-787 (Связь библиографических записей) и 533 (Примечание о репродукции), а также для позиций символов контрольного подполя \$w в полях 4XX и 5XX авторитетных данных.

Н — Не используется

Код Н — “не используется” означает, что элемент данных не определен или не используется, что зависит от характеристик материала, типа определителей содержания или практики авторитетной организации.

Коды требований национального уровня не определены в форматах USMARC Format for Holdings Data, USMARC Format for Classification Data и USMARC Format for Community Information.

4.2 MARCXML

Описание MARC XML

Программистами The Library of Congress' Network Development (Сети Библиотеки Конгресса США) и MARC Standards Office разработан каркас для работы с MARC-форматом в среде XML. Данный каркас создан гибким и расширяемым, чтобы обеспечить пользователям работу с данными MARC в зависимости от их нужд. Он содержит большое количество компонент, таких как схемы, стили и утилиты, разработанные и находящиеся под надзором Библиотеки Конгресса. [7]

Простая и гибкая схема MARC XML

Ядром каркаса MARC XML является простая XML схема, содержащая данные MARC. Этой схемой можно пользоваться в том случае, когда

необходимы полные MARC-записи, когда предполагаются дальнейшие преобразования или валидация. Схема MARC XML не изменяется, отражая мелкие изменения в MARC21. Она сохраняет семантику MARC.

Все контрольные поля, включая лидер, трактуются как строка данных. Поля трактуются как элементы с тегом, а индикаторы — как атрибуты. Подполя трактуются как вложенные элементы с наименованием подполя в атрибуте.

Преобразование без потерь из MARC в XML

Все необходимые данные из записи MARC преобразуются и выражаются в XML. Структурные элементы MARC, такие как длина поля и стартовая позиция данных поля в записях каталогов не требуются в XML записях. Позиции лидирующих символов, ненужные в среде XML сохраняются либо трактуются как пустые символы.

Возврат от XML обратно к MARC

Как следствие преобразования без потерь из MARC(2709), информация в MARC XML допускает полное восстановление в виде MARC(2709). Это преобразование также проводится без потерь данных.

Представление данных

Как только данные MARC преобразованы в XML, становится возможным их представление через написание списка стилей XML для отбора и отображения элементов в конкретной разметке.

Редактирование MARC

Некоторые одиночные или групповые изменения, такие как добавление, обновление или удаление полей в записи MARC могут быть достигнуты простыми преобразованиями XML.

Конвертация данных

Большая часть конвертаций может быть реализована как XML-преобразования. Для более сложных можно реализовать программные инструменты.

Валидация данных

Валидация по схеме выполняется поставляемой программой. Данная программа отделена от схемы и предоставляет 3 возможных уровня валидации:

- Базовая XML валидация в соответствии со схемой MARC XML
- Валидация тегов (полей и подполей) MARC21
- Валидация содержимого записи MARC, в т.ч. значений, дат и времени

Расширяемость

Использование XML в качестве структуры для MARC-записей пользователи MARC XML каркаса получают возможность более легкого написания собственных программ для манипуляций с данными MARC.

Архитектура MARC XML

Схематично архитектура каркаса представлена на рис. 1.

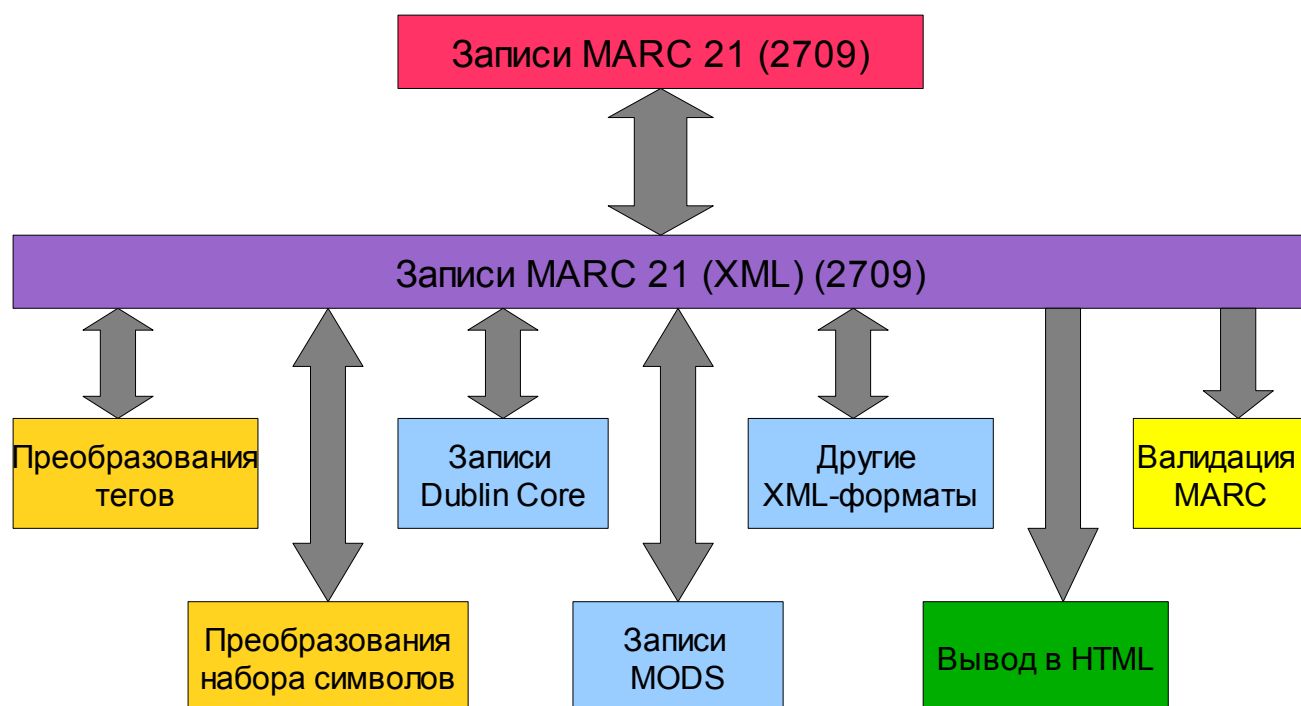


рис 1. Архитектура каркаса MARC XML

Преобразование MARC 21 (2709) в/из MARC 21 (XML)

- Эти инструменты преобразуют данные MARC между структурами MARC (2709) и MARC (XML). Преобразования происходят без потерь.

Шина MARC XML

- Центральная часть каркаса MARC XML — простая и гибкая XML схема, хранящая данные MARC и предоставляющая общий формат для записи дополнений MARC XML

Дополнения MARC XML

Записи MARC XML могут использоваться в «оригинальной» форме. В другом случае, дополнения MARC XML разделяются на 3 категории: преобразования, представление и анализ

- Преобразования — это конвертация между MARC XML и другими форматами метаданных
- Представление — вывод на экран данных MARC в некоторой читаемой

форме

- Анализ — обработка данных MARC для получения аналитических данных, например валидация

MARC XML: использование и возможности

1. Использование

MARC XML потенциально может использоваться для:

- представления полных записей MARC в XML
- как схема расширения для METS (Metadata Encoding and Transmission Standard)
- для отображения метаданных и анализа при помощи OAI
- для описания ресурсов в XML синтаксисе

2. Достоинства MARC XML

Некоторые достоинства MARC XML:

- Схема поддерживает все данные MARC вне зависимости от формата
- Каркас MARC XML имеет компонентно-ориентированную, расширяемую архитектуру, дающую пользователям возможность легко встраивать дополнительные программные расширения

3. Ограничения MARC XML

- Валидация MARC выполняется не по схеме, а сторонним ПО

4.3 MODS

Разработчики Библиотеки Конгресса США и Офис Стандартов MARC

разрабатывает схему для библиографических наборов элементов, которые можно использовать в том числе для библиотечных приложений. Как XML-схема, она предназначена для переноса выбранных данных их существующих записей MARC 21, а также для создания оригинальных записей о ресурсах. Она включает подмножество полей MARC и использует теги-слова вместо численных, что в некоторых случаях приводит к перегруппировке элементов MARC 21. Эта схема в данный момент имеет статус черновика и именуется как Metadata Object Description Schema (MODS). MODS представляет собой XML-схему в соответствии с правилами консорциума WWW. Стандартом управляют разработчики Библиотеки Конгресса США и Офис Стандартов MARC. [8]

1. Использование

MODS может быть использована для:

- формат SRU (Search/Retrieve via URL)
- как схема расширения для METS (Metadata Encoding and Transmission Standard)
- для представления метаданных
- для представления оригинальных ресурсов в XML синтаксисе
- для представления упрощенной записи MARC в XML

2. Достоинства MODS

MODS предназначена для совместного использования с другими форматами метаданных. В некоторых приложениях, особенно использующих записи MARC, она будет иметь преимущества, такие как:

- Набор элементов богаче, нежели Dublin Core
- Набор элементов более совместим с библиотечными данными, нежели ONIX
- Схема больше ориентирована на конечного пользователя, чем на полная схема MARC XML
- Более простой набор элементов, чем в полной схеме

3. Особенности MODS

- Элементы как правило наследуют семантику MARC
- В некоторых случаях несколько элементов MARC могут быть объединены
- MODS не предполагает использование конкретных кодов каталогов
- Некоторые элементы содержат необязательный атрибут ID для связей на уровне элементов

4. Ограничения MODS

MODS включает в себя подмножество данных MARC 21. Как набор элементов, позволяющих отображение данных в MARC-системах, он предназначен для преобразований главных полей с потерей второстепенных данных. Как набор элементов для описания новых ресурсов, он позволяет создавать упрощенные записи с меньшим количеством обязательных тегов, нежели полная MARC запись.

Однако схема не предназначена для многократных преобразований в MARC 21 и обратно. Другими словами, при преобразовании из MARC 21 в

MODS происходит потеря данных, и точное обратное преобразование становится невозможным. В некоторых случаях элементы могут смещаться в более общие в схеме MODS. Тем не менее, сами данные не будут потеряны, лишь детальный тип элемента. В других случаях элемент MARC может не иметь эквивалента в MODS и тогда данные будут потеряны при обработке.

MODS не включает бизнес-правила для управления элементами. Перечисленные в `xsd:documentation` поля предназначены для комментариев, чтобы указать пользователю на семантику для элементов.

5. Порядок элементов

Отметим, что порядок элементов MODS не соответствует порядку отображения. Для управления выводом на экран необходимо использовать стили.

5 Сетевые протоколы

5.1 HTTP

HTTP является одним из механизмов WWW, которая включает в себя еще несколько средств работы с документами: **HyperText Markup Language**, **Common Gateway Interface** (CGI) - общий шлюзовый интерфейс и **Uniform Resource Locator** (URL) - универсальный указатель ресурсов. Основное назначение HTTP - извлечение HTML-документов, адресуемых с помощью URL. WWW-клиент посылает запросы серверу и получает документы в качестве ответа. Основной способ взаимодействия между клиентом и сервером изображен на рис.2.

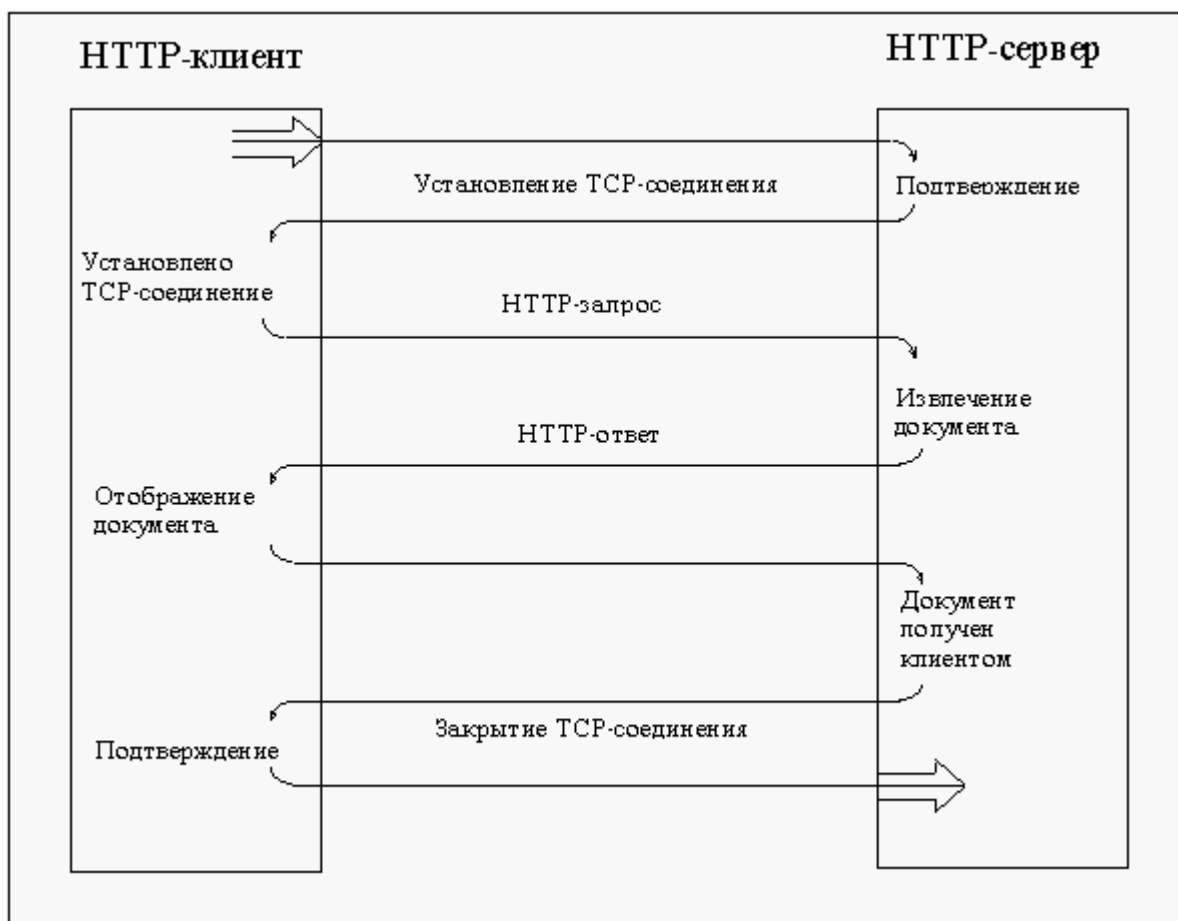


рис.2 Схема работы HTTP-протокола

Спецификация HTTP не оговаривает параметры соединения. Это означает, что HTTP может выполняться и над пакетом протоколов, отличным от TCP/IP, а также это означает возможность проведения более чем одной HTTP-транзакции в рамках одного соединения. Но в настоящий момент ни один из известных HTTP-серверов и клиентов не использует эти возможности. [1]

Формат HTTP-запроса прост по сравнению с Z39.50. В первой строке запроса указывается метод и объект, к которому этот метод применяется. Объект идентифицируется при помощи URI (например URL). Наиболее часто используется метод **"GET"**, который запрашивает сервер выслать копию объекта (HTML-документ) клиенту. Существует еще один важный метод - **"POST"**. С помощью этого метода клиент может посылать данные серверу.

Как сервер будет обрабатывать эти данные, определяется объектом, определенным URL в строке запроса. Обычно этот метод используется в сочетании с меткой FORM HTML-документа, предоставляющей пользователю разнообразные способы ввода информации.

Клиент может также посылать серверу последовательности необязательных заголовков. Наиболее употребительными являются заголовки "Асепт", которые сообщают серверу типы объектов, которые клиент может поддерживать, например:

Асепт: text/plain

Асепт: text/html.

Ответы HTTP-сервера начинаются с заголовков, после которых помещается извлекаемый объект, являющийся в большинстве случаев HTML-документом. Обязательным заголовком является строка статуса, в которой показывается версия HTTP, поддерживаемая данным сервером и результат обработки запроса. Существует 4 категории результатов или кодов статуса:

1. **успешная обработка запроса** - запрос был успешно принят, понят и обработан;
2. **перенаправление** - клиентом должны быть выполнены последующие действия, необходимые для выполнения запроса;
3. **ошибка клиента** - неверен синтаксис запроса или запрос не может быть выполнен;
4. **ошибка сервера** - сервер не может выполнить запрос.

Сервер может посылать клиенту необязательные заголовки. Наиболее важными из них являются: "**Content-Type**", который описывает тип возвращаемого объекта, и "**Content-Length**", отображающий размер объекта.

Рассматривая предыдущий Z39.50-пример в контексте WWW,

получим следующий сценарий обращения к базе данных:

Начало

открытие TCP/IP соединения
посылка GET **"выбор базы данных"**
прием **"выбор базы данных"**
закрытие TCP/IP соединения

Отображение **"выбор базы данных"**

открытие TCP/IP соединения
посылка GET/POST **"поиск в базе данных"**
прием **"поиск в базе данных"**
закрытие TCP/IP соединения

Отображение **"поиск в базе данных"**

открытие TCP/IP соединения
посылка POST **"список заголовков"**
прием **"список заголовков"**
закрытие TCP/IP соединения

Отображение **"список заголовков"**

открытие TCP/IP соединения
посылка GET **"запись"**
прием **"запись"**
закрытие TCP/IP соединения

Отображение **"запись"**

Каждый раз, когда клиент желает послать запрос серверу, он должен открыть соединение, которое закрывается сервером после отправки ответа. Таким образом сервер не имеет информации о результатах выполнения предыдущих запросов. Ранее уже было отмечено, что эта особенность

относится лишь к реализациям HTTP, но не к самому протоколу как таковому.

Первый запрашиваемый документ - "выбор базы данных". Выбор посылается серверу в форме запроса на документ "поиск в базе данных". Этот документ, в свою очередь, содержит HTML-форму, с помощью которой пользователь вводит свой запрос. Этот запрос посылается объекту "список заголовков", который представляет собой CGI-приложение. Данное приложение запускается HTTP-сервером и получает необходимые данные (например ключевое выражение, имя базы данных) через CGI. CGI-приложение способно производить поиск в базе данных и возвращать HTTP-серверу HTML-документ, содержащий информацию о количестве найденных записей и "список заголовков" этих записей. Для каждого заголовка найденной записи документ содержит URL, указывающий местоположение полной записи. Затем, с помощью метода GET, клиент извлекает "запись".

Следует отметить интересную возможность использования Z39.50-клиента в качестве CGI-приложения.

Итак, HTTP является удобным и весьма распространенным средством извлечения документов. однако их поиск может быть осуществлен лишь при помощи CGI-приложения, которое может быть реализовано как клиентская часть какой-либо пиковой службы (в частности Z39.50) или как некий самостоятельный механизм, работающий с локальными структурами данных.

5.2 Z39.50

Z39.50 - Американский Национальный Стандарт, который был принят в 1988 году Национальной Организацией по Информационным Стандартам (NISO), а также Американским Национальным Институтом Стандартов (ANSI). NISO является подразделением ANSI, специализирующимся на издательских, библиотечных и информационных стандартах. В частности NISO разработаны следующие стандарты:

- MARC (Z39.2) - стандарт на формат обмена библиографическими данными,
- Common Command Language - общий командный язык (Z39.58),
- Format for Circulation Transactions - формат транзакций обращения (Z39.70).

Z39.50 - протокол прикладного уровня в рамках семиуровневой эталонной модели взаимодействия открытых систем, разработанной Международной Организацией Стандартов (ISO) и поэтому может быть реализован в различных типах сетей (например, в сетях TCP/IP, IPX/SPX, OSI), независимо от реализации транспортного уровня. Его назначение - предоставить компьютеру, работающему в режиме "клиент", возможности поиска и извлечения информации из другого компьютера, работающего как информационный сервер. [2]

Стандарт определяет для компьютеров-клиентов единую процедуру запроса информационных ресурсов - серверов, поддерживающих библиотечные каталоги.

Первоначально многие приложения Z39.50 создавались исключительно для использования с библиографическими данными (например, общедоступные библиотечные каталоги). Однако, в настоящее время, протокол развит настолько, что позволяет обрабатывать различные данные - финансовую, химическую, техническую информацию, тексты и изображения. Использование специфических данных (например, изображений) требует расширений протокола. Группы разработчиков Z39.50-приложений в настоящее время исследуют такие возможности и определяют наборы атрибутов для разнообразных типов данных (в т.ч. и полнотекстовых).

Z39.50 не является международным стандартом. Однако, существует

стандарт ISO 10162/10163 "Поиск и извлечение" (SR), который был принят в качестве международного в 1991 году. В целом он является подмножеством Z39.50. В 1992 году Z39.50 был приведен в полное соответствие с SR и таким образом стал совместимым надмножеством международного стандарта. В 1995 году начался процесс принятия Z39.50 как международного стандарта.

Модель службы Z39.50 предусматривает обмен сообщениями типа "запрос - ответ" между соответствующими приложениями - клиентом и сервером. Формат таких сообщений и определяется протоколом Z39.50. Типовая последовательность обмена такими сообщениями представлена на рис. 3.

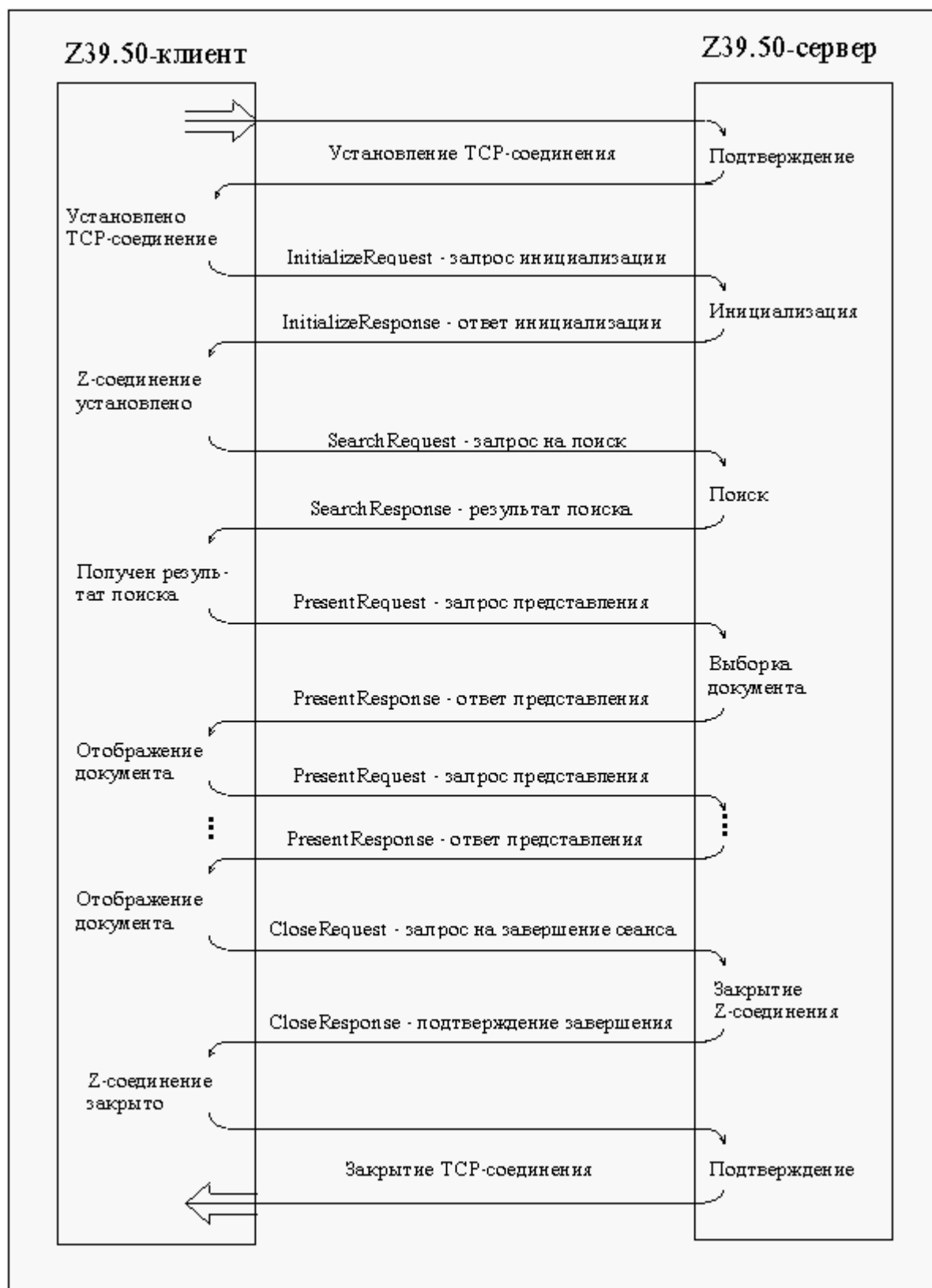


рис.3 Схема работы протокола Z39.50

После установления TCP-соединения (или любого другого,

зависящего от способа передачи данных) устанавливается Z39.50-соединение, посредством обмена протокольными блоками данных - Protocol Data Unit (PDU). Пример формата PDU инициализации, который клиент посылает серверу представлен в таблице 1. В первом столбце перечислены имена элементов PDU, во втором представлены данные, в действительности передающиеся по линии связи.

Таблица 1
Пример формата PDU запроса на инициализацию

Имя PDU / тега	Тег / значение параметра	Краткое описание
InitializeRequest	[20]	Запрос на инициализацию
protocolVersion	[3]	Сведения о поддерживаемой клиентом версии протокола
version-1	1	Этот бит всегда должен быть установлен в 1
version-2	1	Поддерживается версия 2
version-3	0	Не поддерживается версия 3
Options	[4]	Сведения о поддерживаемых клиентом услугах
search	1	Клиент поддерживает услугу поиска
present	1	Клиент поддерживает услугу представления
delSet	1	Клиент поддерживает услугу удаления результирующего множества
resourceReport	0	Клиент не поддерживает услугу отчета о ресурсах
triggerResourceCtrl	0	Клиент не поддерживает услугу переключения контроля ресурсов
resourceCtrl	0	Клиент не поддерживает услугу контроля ресурсов
accessCtrl	0	Клиент не поддерживает услугу контроля доступа
scan	0	Клиент поддерживает услугу просмотра списка поисковых термов
sort	0	Клиент не поддерживает услугу сортировки результирующего множества
--	0	Зарезервировано
extendedServices	0	Клиент не поддерживает услугу расширенных сервисов
level-1Segmentation	0	Клиент не поддерживает сегментацию 1-го уровня
level-2Segmentation	0	Клиент не поддерживает сегментацию 2-го уровня
concurrentOperations	0	Клиент не поддерживает возможность выполнения параллельных операций
namedResultSets	1	Клиент поддерживает услугу именования результирующих множеств

Имя PDU / тега	Тег / значение параметра	Краткое описание
preferredMessageSize	[5], 1024	Предпочтительный размер сообщения (в байтах)
exceptionalRecordSize	[6], 65536	Предельный размер записи (в байтах)

Механизм кодирования протокольных блоков данных предполагает использование нескольких способов для распознавания их элементов. Одним из таких методов является использование тегов (меток), представленных в данном случае числами в квадратных скобках. После каждого тега следует значение элемента. Существуют также теги для идентификации самих PDU. В этом примере тег [20] идентифицирует запрос как запрос инициализации (InitializeRequest). В данном случае было послано 4 элемента PDU. Клиент сообщил серверу, что поддерживает версию 2 протокола, но не версию 3. С помощью элемента options (метка [4]) клиент сообщил серверу о своих возможностях; в данном случае он может посылать запросы на поиск, получать найденные записи, просматривать список поисковых термов, представляемый сервером и т.д. Также клиент уведомил сервер о том, что ему предпочтительнее работать с сообщениями размером не более 1024 байт. В крайних случаях он может принять запись размером 65535 байт.

По получении этого PDU от клиента сервер отвечает следующим образом: (см. табл.2).

Таблица 2
Пример формата PDU ответа инициализации

Имя PDU / тега	Тег / значение параметра	Краткое описание
InitializeResponse	[21]	Ответ инициализации
protocolVersion	[3]	Сведения о поддерживаемой клиентом версии протокола
version-1	1	Этот бит всегда должен быть установлен в 1
version-2	1	Поддерживается версия 2
version-3	0	Не поддерживается версия 3
Options	[4]	Сведения о поддерживаемых сервером услугах
search	1	Сервер поддерживает услугу поиска

Имя PDU / тега	Тег / значение параметра	Краткое описание
present	1	Сервер поддерживает услугу представления
delSet	0	Сервер не поддерживает услугу удаления результирующего множества
resourceReport	0	Сервер не поддерживает услугу отчета о ресурсах
triggerResourceCtrl	0	Сервер не поддерживает услугу переключения контроля ресурсов
resourceCtrl	0	Сервер не поддерживает услугу контроля ресурсов
accessCtrl	0	Сервер не поддерживает услугу контроля доступа
scan	0	Сервер не поддерживает услугу просмотра списка поисковых термов
sort	0	Сервер не поддерживает услугу сортировки результирующего множества
--	0	Зарезервировано
extendedServices	0	Сервер не поддерживает услугу расширенных сервисов
level-1Segmentation	0	Сервер не поддерживает сегментацию 1-го уровня
level-2Segmentation	0	Сервер не поддерживает сегментацию 2-го уровня
concurrentOperations	0	Сервер не поддерживает возможность выполнения параллельных операций
namedResultSets	0	Сервер не поддерживает услугу именования результирующих множеств
preferredMessageSize	[5], 1024	Предпочтительный размер сообщения (в байтах)
exceptionalRecordSize	[6], 65536	Предельный размер записи (в байтах)
result	[12], TRUE	Z-соединение установлено

Сервер принял некоторые из параметров инициализации, установленные клиентом, но предложил не использовать в данном сеансе возможности удаления и именования результирующих множеств. По принятии этого сообщения клиентом Z39.50 соединение считается установленным. В этом случае клиент может либо продолжить работу с такими параметрами, либо закрыть соединение и попытаться затем установить новое - быть может с другими параметрами. После установки Z39.50 соединения пользователь может вводить запрос на поиск информации. Например 'Год издания=1942'. В этом случае клиент посылает PDU такого

вида:

Таблица 3
Пример формата PDU запроса на поиск

Имя PDU / тега	Тег / значение параметра	Краткое описание
SearchRequest	[22]	Запрос на поиск
smallSetUpperBound	[13],0	Верхняя граница "малого" множества
largeSetLowerBound	[14],1	Нижняя граница "большого" множества
mediumSetPresentNumber	[15],0	Количество возвращаемых записей из "среднего" множества
replaceIndicator	[16], TRUE	Если результирующее множество с именем resultSetName уже существует, то заменить его
resultSetNames	[17], "default"	Имя результирующего множества
databaseNames	"BOOKS"	Имена баз данных, в которых проводится поиск
query	[21]	Запрос
type-1	[1]	Типа 1
attributeSet	1.2.840.10003.3.1	Идентификатор набора атрибутов bib-1
op	[0]	Операнд
AttributesPlusTerm	[102]	Атрибуты + терм
AttributeList	[44]	Список атрибутов
attributeType	[120],1	Поиск
attributeValue	[121],31	По году издания
attributeType	[120],2	Отношение
attributeValue	[121],3	Равенство - искать записи о документах, год издания которых равен заданному
attributeType	[120],3	Местоположение терма
attributeValue	[121],3	Любое в поле
attributeType	[120],4	Структура терма
attributeValue	[121],4	Год
attributeType	[120],5	Признак усечения
attributeValue	[121],100	Не усекаать
term	[45],1942	Искомое выражение

Основной целью ответа о результатах поиска (SearchResponse) является предоставление клиенту данных о числе найденных записей (т.е. количестве документов, соответствующих ключевому выражению при поиске в базе данных). Однако в запросе на поиск (searchRequest) клиент может попросить сервер указать в ответе (SearchResponse) не только число найденных записей, но и сами записи. В результате обработки запроса на

поиск сервер формирует результирующее множество - набор указателей на извлекаемые записи. Впоследствии клиент может извлекать найденные записи, указывая номер элемента из результирующего множества. С помощью параметров `smallSetUpperBound` и `largeSetLowerBound` клиент может указать, что считать "малым" множеством результатов (число записей меньше или равно `smallSetUpperBound`), что считать "большим" множеством результатов (число записей больше или равно `largeSetLowerBound`) и что считать средним множеством результатов (ни "большим", ни "малым"). Записи из "малого" множества сразу возвращаются клиенту в ответе о результатах поиска (`SearchResponse`). Когда множество результатов - среднее, `mediumSetNumber` является числом возвращаемых записей, которое меньше общего количества записей. В нашем примере клиент попросил сервер не возвращать записи в ответе о результатах поиска.

Элемент `replaceIndicator` установлен в `TRUE`, это означает, что множество результатов с именем `resultSetname` может заменено другим с таким же именем. Множество результатов в данном случае названо "default". "BOOKS" - имя базы данных, в которой производится поиск. Протокол позволяет искать записи более чем в одной базе. Z39.50 различает несколько типов запросов. Любая реализация Z39.50 должна по крайней мере поддерживать запрос типа `type-1`, также известного под именем запроса с обратной польской записью - Reverse Polish Notation (RPN) query. В данном случае запрос состоит из одного операнда.

С каждой базой данных связан как минимум один набор атрибутов, определяющих смысл искомого выражения (терма). Существует несколько наборов атрибутов для ключевых выражений. Эти наборы определяются т.н. идентификаторами объектов. Наиболее используемым набором атрибутов является Bib-1. Атрибут «use» из этого набора может принимать значения `author`, `title`, `ISBN`, `abstract`, `subject`, `publisher` и т.д. Сервер реагирует посылкой

клиенту следующего PDU:

Таблица 4

Пример формата PDU результатов поиска

Имя PDU / тега	Тег / значение параметра	Краткое описание
SearchResponse	[23]	Результат поиска
resultCount	[23],3	Количество найденных записей
numberOfRecordsReturned	[24],0	Количество возвращенных записей
nextResultSetPosition	[25],1	Номер следующей извлекаемой записи
searchStatus	[22],TRUE	Результат поиска - успешен

Клиент на основе этой информации сообщает пользователю о результатах поиска. Затем пользователь может попросить показать найденные записи. Для этого клиент посылает PresentRequest - запрос представления.

Таблица 5

Пример формата PDU запроса на представление

Имя PDU / тега	Тег / значение параметра	Краткое описание
PresentRequest	[24]	Запрос представления
resultSetId	"default"	Идентификатор результирующего множества
resultSetStartPoint	[30],1	Номер первой возвращаемой записи
numberOfRecordsRequested	[29],3	Требуемое количество записей
recordComposition	[19],"B"	Форма представления записи - "B" - сокращенная
preferredRecordSyntax	[104],1.2.840.10003.5.101	Формат записи - SUTRS - простой текстовый формат

Z39.50-1995 определяет возможности управления формой представления найденных записей - клиент может указывать элементы (поля, параграфы, разделы, части) записи или документа, которые он бы хотел получить от сервера и формат (SGML, Postscript, TIFF и т.д.) этих элементов. В данном примере используется более простой механизм (Z39.50-1992), где способ представления записи (тег [19]) указан буквой "B" или "F" соответственно для сокращенной и полной формы. Форма, в которой запись передается клиенту также определяется синтаксисом записи. Z39.50-1995 определяет 27 различных синтаксисов записей, из которых 15 относятся к

различным диалектам формата MARC. Наиболее простым синтаксисом является синтаксис записей простых неструктурированных текстов - Simple Unstructured Text Record Syntax (SUTRS): строки текста длиной не более 72 символов, завершаемые символом LF (ASCII 0A16). В данном примере клиент просит сервер использовать SUTRS, что определяется идентификатором объекта (тег [104]). В PDU PresentResponse сервер передает записи из множества "default":

Таблица 6
Пример формата PDU ответа представления

Имя PDU / тега	Тег / значение параметра	Краткое описание
PresentResponse	[25]	Ответ представления
numberOfRecordsReturned	[24],3	Количество возвращенных записей - 3
nextResultSetPosition	[25],0	Номер следующей записи - 0 - уже все записи возвращены
presentStatus	[27],0	Результат представления - 0 - успешно
responseRecords	[28]	Возвращаемые записи
dbname	[0],"BOOKS"	Имя базы данных, из которой извлечена запись
retrievalRecord	[1]	Извлекаемая запись
recordSyntax	1.2.840.10003.5.101	Формат записи - SUTRS - простой текстовый формат
record	"Dinesen I. Winter's tales..."	Собственно запись
dbname	[0],"BOOKS"	Имя базы данных, из которой извлечена запись
retrievalRecord	[1]	Извлекаемая запись
recordSyntax	1.2.840.10003.5.101	Формат записи - SUTRS - простой текстовый формат
record	"Armstrong T. The Crowthers of Bankham..."	Собственно запись
dbname	[0],"BOOKS"	Имя базы данных, из которой извлечена запись
retrievalRecord	[1]	Извлекаемая запись
recordSyntax	1.2.840.10003.5.101	Формат записи - SUTRS - простой текстовый формат
record	"Красиков И., Михайлов И. Уборка и хранение овощей..."	Собственно запись

PDU presentRequest также может быть использован для запроса записи в полной форме:

Таблица 7
Пример формата PDU запроса на представление

Имя PDU / тега	Тег / значение параметра	Краткое описание
PresentRequest	[24]	Запрос представления
resultSetId	"default"	Идентификатор результирующего множества
resultSetStartPoint	[30],1	Номер первой возвращаемой записи
numberOfrecordsRequested	[29],3	Требуемое количество записей
recordComposition	[19],"F"	Форма представления записи - "F" - полная
preferredRecordSyntax	[104],1.2.840.10003.5.101	Формат записи - SUTRS - простой текстовый формат

Сервер возвращает 3-ю запись из набора "default":

Таблица 8
Пример формата PDU ответа представления

Имя PDU / тега	Тег / значение параметра	Краткое описание
PresentResponse	[25]	Ответ представления
numberOfrecordsReturned	[24],1	Количество возвращенных записей - 1
nextResultSetPosition	[25],0	Номер следующей записи - 0 - эта запись - последняя
presentStatus	[27],0	Результат представления - 0 - успешно
responseRecords	[28]	Возвращаемые записи
dbname	[0],"BOOKS"	Имя базы данных, из которой извлечена запись
retrievalRecord	[1]	Извлекаемая запись
recordSyntax	1.2.840.10003.5.101	Формат записи - SUTRS - простой текстовый формат
record	"Красиков И., Михайлов И. Уборка и хранение овощей,-Л.: Ленингр. газ.-журн.,1942"	Собственно запись

В данном примере рассмотрено использование лишь нескольких основных услуг Z39.50: поиска и извлечения. Существуют и такие услуги, как:

- сортировка результирующего множества по какому либо из полей

записей (например, по имени автора, по году издания и т.п.);

- получение списка термов (scan) - пользователь может получить список всех слов, по которым в базе данных может быть произведен успешный поиск;
- поддержка (именование и удаление) нескольких результирующих множеств - пользователь может работать одновременно с несколькими наборами записей, используя их в последующих запросах, объединять их, сохранять и удалять.

Механизм расширенных услуг предоставляет возможности подписки на получение новых библиографических описаний, заказа изданий по найденным описаниям. Таким образом, можно с уверенностью сказать, что Z39.50 является удобным средством для построения систем обработки библиографической информации, ориентированным как на профессиональных, так и непрофессиональных пользователей.

6 Система Web-IRBIS

6.1 Общие сведения

Программное обеспечение Web-ИРБИС предназначено для обеспечения доступа пользователей Интернет к электронным каталогам и другим библиографическим базам данных, подготовленных с помощью CDS/ISIS, построенных на его базе и системы ИРБИС. Web-ИРБИС используется в качестве одной из основных компонент библиотечных Интернет- серверов и Интернет- комплексов (БИК).

6.1.1 Основные системные характеристики Web-ИРБИС

- Единая технология обслуживания локальных и удаленных пользователей.
- Единая технология обслуживания запросов, поступающих в режиме

онлайн и по электронной почте.

- Низкие первоначальные затраты на программно-технический комплекс.
- Низкие затраты на сопровождение и администрирование комплекса.
- Возможность использования единого библиографического массива, как для технологических операций, так и для обслуживания локальных и удаленных пользователей.
- Наличие системы автоматического распознавания кодировки входящего пользователя (в онлайн-режиме и в режиме электронной почты).
- Возможность работы с любым количеством библиографических баз данных.
- Возможность формирования запроса в фиксированной и свободной формах с использованием большинства операций информационного поиска.
- Возможность использования любых форматов вывода, определяемых пользователем, в том числе RUSMARC, UNIMARC, USMARC.

6.1.2 Базовые операции Web-ИРБИС

- Поиск в произвольной базе данных, имеющей структуру CDS/ISIS – ИРБИС по неограниченному числу полей, по любым элементам описания и их комбинаций, в том числе определенных набором Bib-1¹, принятым де-факто, как стандарт поисковых определений библиотечных систем, с применением логики «И», «ИЛИ» и «ФРАЗА ЦЕЛИКОМ», с возможностями определения префиксов и квалификаторов поисковых терминов, грамматической нормализации слов русского языка и применения аппарата усечений.

¹ Bib-1 – набор поисковых и уточняющих атрибутов, принятых в качестве обязательных при построении БИК и корпоративных систем.

- Использование при поиске статических словарей и рубрикаторов, включенных в поисковые формы или подключаемых извне, с возможностью комбинирования элементов словарей с любыми другими поисковыми предписаниями.
- Использование динамических словарей баз данных с возможностью получения списка терминов словаря с последующим поиском по выбранным терминам; навигация по словарям, включая задание начала сканирования по первым символам, и в терминах «следующие», «предыдущие».
- Показ записей из произвольной базы данных во всех типах стандартных форматов, включая информационный, в виде каталожной карточки, в метках и расшифрованных RUSMARC, UNIMARC, USMARC. Возможно применение любых форматов, определяемых пользователем в нотации ИРБИС.
- Квантование выводимых записей в формате, определяемым пользователем, с последующей навигацией в терминах «следующие», «предыдущие».
- Выбор записей из числа найденных, с последующей их обработкой, включая выгрузку выбранных записей в стандартных форматах RUSMARC, UNIMARC, USMARC.
- Удаленное авторизованное внесение изменений в базу данных, включая редактирование существующих записей, добавление новых, удаление без актуализации.

Программное обеспечение разрабатывалось таким образом, что все функции и возможности полностью управляются администратором библиотечного информационного комплекса, включая настройку поисковых

форм, добавление и удаление поисковых полей, настройку их атрибутов, включение рубрикаторов и статических словарей.

Особое внимание при разработке Web-ИРБИС уделялось дополнительным сервисным функциям, направленным на улучшение обслуживания пользователей и интегрирование внутрибиблиотечных и межбиблиотечных технологий посредством библиотечного информационного комплекса.

Это особенно важно для приближения к современным технологиям удаленного обслуживания читателей и электронных библиотек. Известно, что типичные библиотечные информационные комплексы имеют существенный недостаток – разрыв в технологии поиска и последующего доступа к найденным первоисточникам. Безусловно, достаточно неудобна для пользователя ситуация, когда он может найти описание необходимого ему издания вне зависимости от временных и географических условий и не может затем использовать современные коммуникационные технологии для получения нужного ему документа.

Для решения этой проблемы, были введены элементы поддержки электронной доставки документов и прямого доступа к полным текстам (если они имеются) по результатам поиска (при этом, разумеется, учитывались правовые ограничения, связанные с необходимостью соблюдения авторского права, т.е. размещение полнотекстовых документов и условия их использования в Интернет проходили требуемые согласование с правообладателями).

Необходимо отметить созданную возможность использования различных режимов доступа к базам данных. Для этого в процессе исследований и разработки были введены 4 базовых режима, свойства которых могут комбинироваться и переопределяться для каждого

конкретного Интернет - комплекса.

Также следует подчеркнуть, что технология авторизованного доступа построена не на уровне стандартных методов Web- серверов, а на базе оригинальных решений с использованием динамически формируемых страниц и библиотек форм-файлов. Это было впервые реализовано для библиотечных информационных комплексов с целью расширения возможностей адаптации к конкретным условиям и повышению устойчивости к несанкционированному вмешательству.

Следующей важной особенностью Web-ИРБИС является встроенная поддержка включения в электронный каталог и базу данных ссылок на полные тексты документов. При этом реализована свободная технология формирования ссылок в виде URL, что позволяет использовать программного обеспечения не только для формирования собственной коллекции полных текстов, но и для каталогизации и включения в базу данных ресурсов Интернет, что крайне важно в современных условиях. Таким образом, Web-ИРБИС может использоваться в качестве базовой технологии при создании виртуальных и электронных библиотек.

Процесс создания ссылок достаточно прост и полностью интегрирован с технологией создания каталогизационного описания ИРБИС: при форматировании записи автоматически генерируется стандартная ссылка и в описании появляется прямая ссылка на полный текст документа.

К достоинствам Web-ИРБИС можно отнести высокую скорость выполнения запросов различной степени сложности.

Web-ИРБИС осуществляет поддержку расширенных форматов, установку параметров выдачи; допускает возможность использования расширенных средств формирования запросов, использования нескольких терминов в одном поле; имеет расширенные функции сервиса.

6.1.3 Основные программно-технологические компоненты

Web-ИРБИС базируется на следующей совокупности основных компонент (см. рис. 4):

- Модуль обеспечения связи между Web- сервером и БД, созданного на основе CGI (Common Gateway Interface – общий интерфейс шлюзов) технологии для Web- серверов Apache, WebSite. Кроме того, он обеспечивает работу программного обеспечения с любыми другими серверами, поддерживающими данную технологию, которая является стандартом для выполнения приложений в Интернет. Для IIS используется технология поддержки внутреннего API (внутреннего интерфейса приложений). Существуют различные режимы для поддержки двух стандартных методов взаимодействия POST и GET. Для создания всех видов модулей был использован язык программирования C++. Модули, использующие CGI, выполнены в виде исполняемых программ, API – в виде динамических библиотек. Данные модули являются единственными специфическими для различных Web- серверов - все остальные не зависят от типа указанных серверов и методов взаимодействия.
- Модули поиска и форматирования результатов являются общими для Web-ИРБИС и ИРБИС и выполнены в виде динамических библиотек. Управляются модулем связи и обеспечивают поиск и вывод предварительно форматированных результатов в соответствии с переданными параметрами. Окончательное форматирование и конвертирование в HTML осуществляется модулем связи.
- Библиотека форм-файлов – основа технологии безопасного авторизованного доступа. Разработанный алгоритм позволяет не хранить постоянно поисковые страницы и страницы вывода результатов – они создаются при получении запроса на выполнение

операции и удаляются после завершения процесса. Поэтому невозможно осуществить несанкционированный доступ ко всем режимам из Интернет по прямым URL (универсальный локатор – указатель – ресурсов).

- Технологические базы данных. Содержат информацию о регистрации удаленных пользователей и поступивших заказах.
- Дерево статических страниц. Используется для первоначального изучения и в качестве примеров для оформления своего сервера.

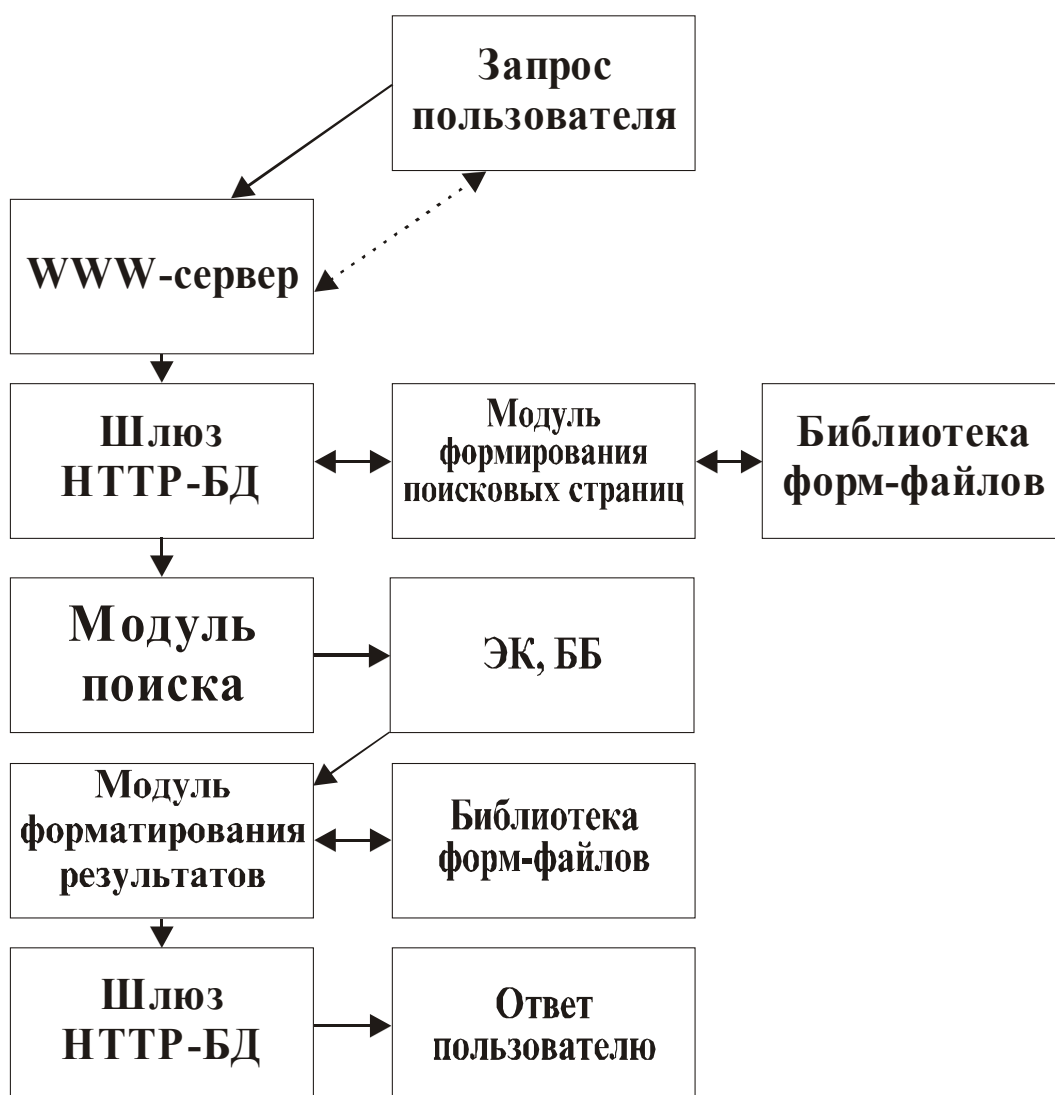


рис. 4. Общая функциональная блок-схема алгоритма работы Web-ИРБИС

6.2 Особенности работы Web-ИРБИС

В общем виде решение функционирует, как показано на рис. 5.

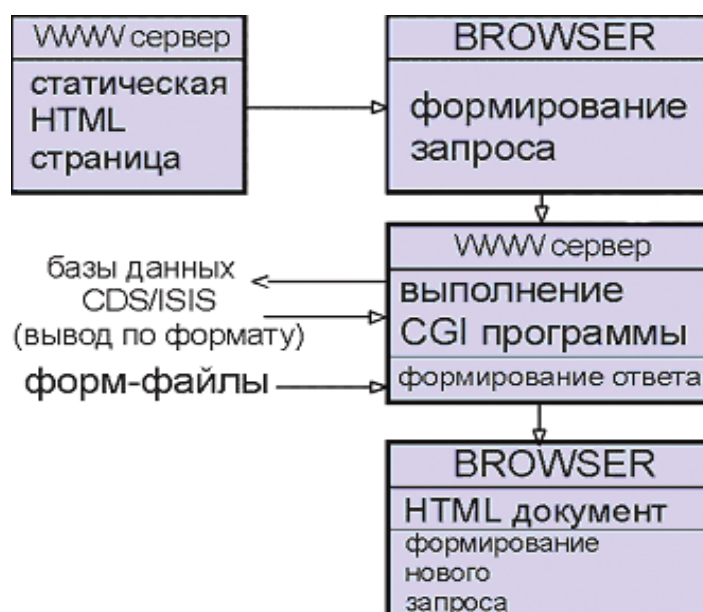


рис. 5. Функциональная схема типового решения.

Работа Web-ИРБИС основана на использовании форм-файлов. Запрос, обрабатываемый CGI программой, формирует ответ в виде очередной HTML формы для дальнейшей работы. Ответ строится на основе predetermined форм-файлов. Для построения ответа на CGI запрос по каждой команде используется строго определенная последовательность вывода форм-файлов. Эти последовательности вывода формируют законченную HTML страницу.

[10]

7 Результаты

7.1 Основные различия между Z39.50 и HTTP

Одним из различий между Z39.50 и HTTP является способ связи клиента и сервера. HTTP-серверу, как уже было сказано, ничего не известно о результатах предыдущего взаимодействия с клиентом, в то время как Z39.50 ориентирован на поддержание какого-либо режима работы (поиск, представление данных и т.п.), причем каждый режим работы использует информацию о результатах выполнения предыдущих операций. Например, извлечение документов производится при помощи результирующего множества, сформированного в результате обработки запроса на поиск и хранимого сервером до закрытия Z-соединения. На практике это различие относительно. Существует несколько способов имитации постоянного соединения при использовании HTTP. С другой стороны, непрерывно увеличивающаяся производительность машин-клиентов и серверов снижает потребность в поддержании постоянного Z-соединения в реализациях Z39.50 (т.е. отпадает необходимость в хранении и сложном администрировании результатов поиска - серверу проще обработать весь запрос снова).

Следующим критерием сравнения протоколов может являться сложность их реализации. HTTP - простой протокол, по сравнению с Z39.50. По существу, единственной услугой, которую предлагает HTTP, является возможность получения документа, указанного URL. Это также возможно и с помощью Z39.50, но Z39.50 предлагает множество других услуг. Сила HTTP - в его комбинации с HTML и CGI. Хотя HTTP - развивающийся протокол и непрерывно усложняется, по сравнению с Z39.50 он очень прост. С точки зрения разработчиков программного обеспечения HTTP-сообщения представляют собой обычные текстовые документы, передаваемые по сети, в то время как Z39.50-сообщения являются протокольными блоками данных,

передаваемыми между клиентом и сервером.

Еще одним важным свойством рассматриваемых протоколов является их интеллектуальность. Рассматривая HTTP и Z39.50 как протоколы извлечения информации, следует отметить, что Z39.50-клиент более пассивен, чем HTTP-клиент, но более интеллектуален. HTTP-клиент запрашивает у сервера поисковую форму, определяющую сложность запроса, составляемого конечным пользователем путем заполнения этой формы. Z39.50-клиент имеет возможность посылать сложные запросы без консультаций с сервером. Z39.50-клиент может информировать сервер о своем состоянии, также как сервер клиента о результатах поиска информации.

И, наконец, Z39.50 и HTTP можно сравнить по их назначению. Z39.50 является специализированным протоколом с возможностями, которые необходимы в контексте автоматизации библиотечных процессов и поддержки доступа к базам данных в реальном времени. HTTP же является протоколом общего назначения.

7.2 Возможности построения информационных систем на базе Z39.50 и HTTP

Нетрудно видеть, что оба протокола имеют свои достоинства и недостатки. Поэтому имеет смысл строить информационные системы с использованием обоих протоколов так, чтобы скомпенсировать недостатки одного протокола достоинствами другого, тем более, что как уже было сказано выше, нет принципиальных ограничений на построение таких систем. Предполагаемая укрупненная структурная схема такой системы представлена на рис.6.

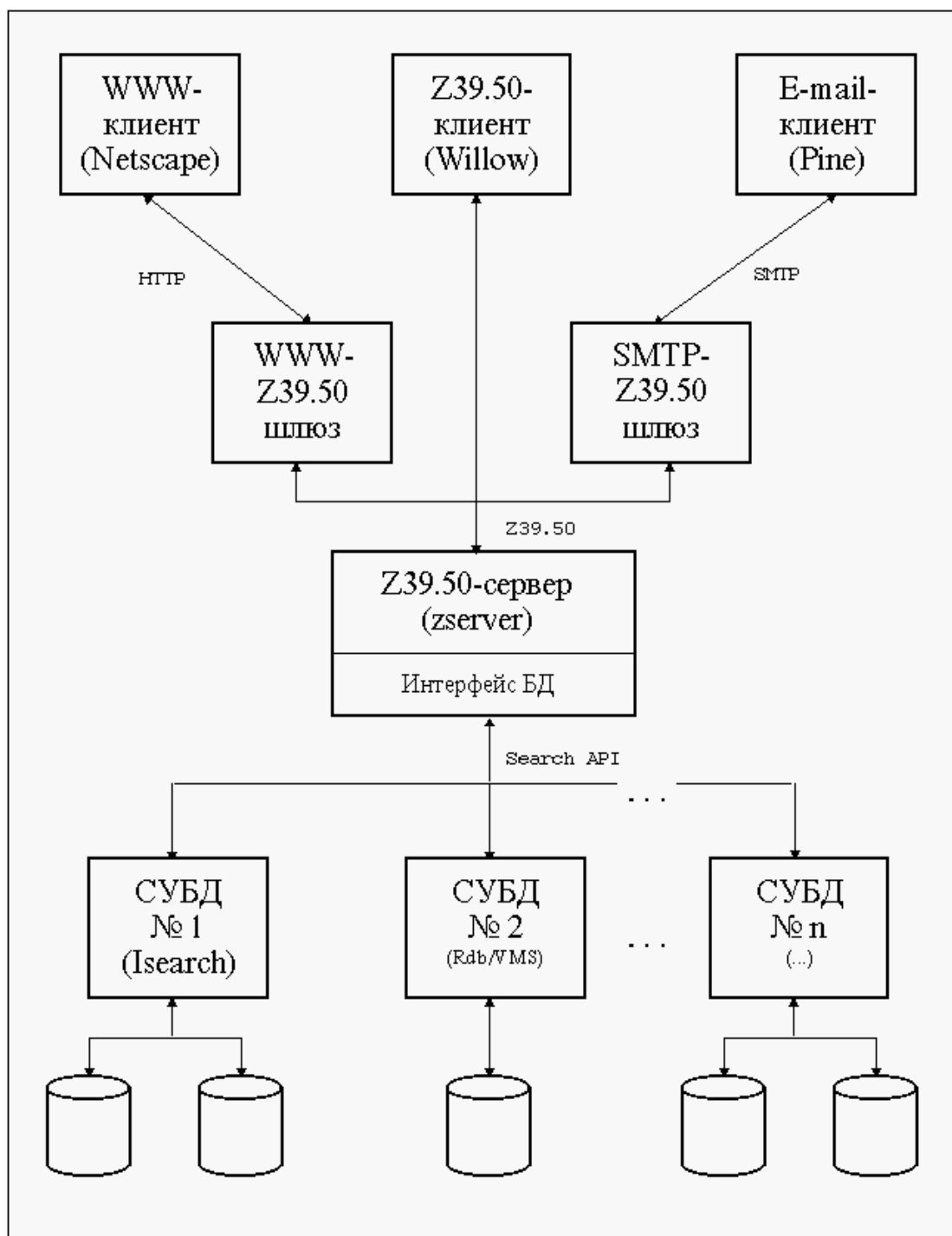


рис.6 Концептуальная схема интегрированной HTTP/Z39.05 системы

Z39.50-сервер обеспечивает, по сути дела, единый способ доступа к разнообразным базам данных - полнотекстовым (Isearch), реляционным (Rdb/VMS), и любым другим. Пользователи получают доступ к базам данных либо при помощи Z39.50-клиентов (Willow и др.), либо с помощью WWW-клиентов (Netscape Navigator и т.п.). Существует практика организации

почтовых интерфейсов к различным поисковым службам, так что представляется возможным организация доступа пользователей к базе данных по электронной почте (при помощи соответствующих клиентов - например Pine). Очевидна необходимость построения соответствующих шлюзов, преобразующих протокольные запросы соответствующих клиентов (HTTP и SMTP) в Z39.50-запросы, направляемые затем серверу и организующих возврат результатов поиска и извлекаемых записей или документов пользователям. Взаимодействие же Z39.50-сервера с различными СУБД осуществляется при помощи соответствующих программных интерфейсов (Search API).

Оба рассмотренных протокола постоянно развиваются, предоставляя разработчикам новые функциональные возможности обработки информации. По-видимому, это развитие будет происходить параллельно, представляя два альтернативных способа коллективного доступа к библиографическим базам данных. Однако, как уже было сказано, существуют возможности взаимодействия систем, использующих этих протоколы.

За относительно небольшой срок существования HTTP и Z39.50 создано довольно большое количество соответствующих серверов и клиентов. Однако общедоступные их реализации обладают только ограниченным набором возможностей. При построении сложных систем обработки информации требуется, создание оригинального программного обеспечения, выполняющего все необходимые функции или приобретение коммерческих систем, если таковые имеются.

8 Заключение

8.1 Выводы

По результатам проведенного исследования можно сделать следующие выводы:

- Система Веб-ИРБИС полностью удовлетворяет требованиям по экспорту данных, имея возможность работы с форматами семейства MARC 21
- Для удобства обмена данными желательно интегрировать в систему инструменты MARC XML для конвертаций в форматы XML и MODS
- Помимо используемого в настоящее время протокола HTTP желательно добавить поддержку Z39.50 как более гибкого и мощного специализированного протокола, что позволит расширить взаимодействие со сторонними библиотечными системами. Протокол допускает реализацию не всех своих возможностей, что позволяет вести разработку постепенно, контролируя результаты итераций.

8.2 Перспективы

Консорциум Open Archives Initiative предлагает интересное альтернативное решение вопроса межбиблиотечной коммуникации в виде протокола верхнего уровня OAI-PMH. [13]

При этом сохраняется существующая инфраструктура, поверх которой реализуется HTTP/XML протокол обмена данными (см. рис. 7 и 8). [14]

Положительной стороной решения является более высокое качество получаемых результатов при поиске.

К минусам решения относятся:

- необходимость реализации дополнительного протокола в каждой из

систем, с которой требуется взаимодействие.

- новый протокол, который не застрахован от ошибок и неточностей
- повышенная нагрузка сети

В связи с этим рекомендуется рассмотреть реализацию такой возможности позднее, по мере «взросления» протокола и исправления его недочетов.

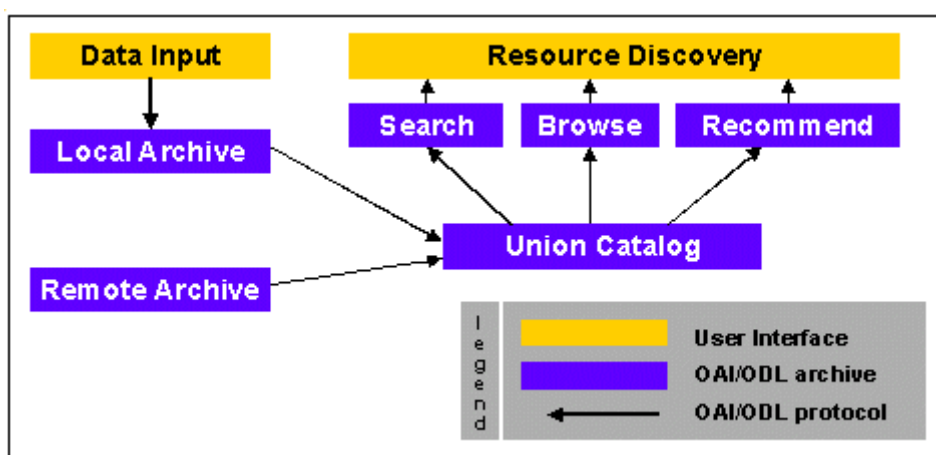


рис. 7 Пример сетевой архитектуры Open Digital Library

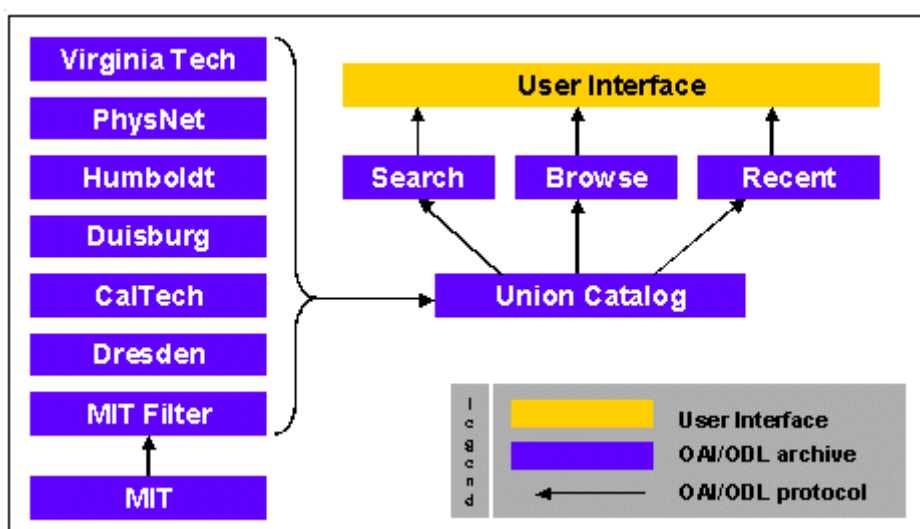


рис. 8 Архитектура системы NDLTD ODL

9 Список литературы

1. Племяк А.И., Усманов Р.Т., Сова Д.Н. Использование протоколов Z39.50 и HTTP в современных библиотечных информационных системах // Информационный бюллетень РБА.-1998.-N 12.-с.240-257.
2. Племяк А.И., Усманов Р.Т. Z39.50: Открытый доступ к библиографической информации // Научные и технические библиотеки.-1998.-N 8.-с.24-28.
3. Публикации участников RUSLANet
<http://www.unilib.neva.ru/rus/olsc/publications/>
4. Е.Н. Филинов, А.В. Бойченко. Вопросы разработки профилей электронных библиотек. Труды 2-ой Всероссийской научной конференции "Электронные библиотеки: перспективные методы и технологии, электронные коллекции". Протвино, 26-28 сентября 2000 г.
5. Библиотека Конгресса США, раздел «Библиотечные стандарты»
<http://www.loc.gov/standards/>
6. ГПНТБ России, Форматы USMARC
<http://www.gpntb.ru/win/usmarc/>
7. MARC 21 XML Schema
<http://www.loc.gov/standards/marcxml/>
8. Metadata Object Description Schema
<http://www.loc.gov/standards/mods/>
9. Web - ИРБИС. Руководство пользователя — М.:ГПНТБ России, 2003. — 77 с.

10. Сайт разработчиков системы «ИРБИС»

<http://irbis.gpntb.ru/index.php>

11. DELOS Network of Excellence on Digital Libraries

<http://www.delos.info/>

12. European Conference on Digital Libraries

<http://www.ecdlconference.eu/>

13. The Open Archives Initiative Protocol for Metadata Harvesting

<http://www.openarchives.org/OAI/openarchivesprotocol.htm>

14. A Framework for Building Open Digital Libraries

<http://www.dlib.org/dlib/december01/suleman/12suleman.html>