

Информационная система учреждения культуры - как система обработки объектов электронного каталога

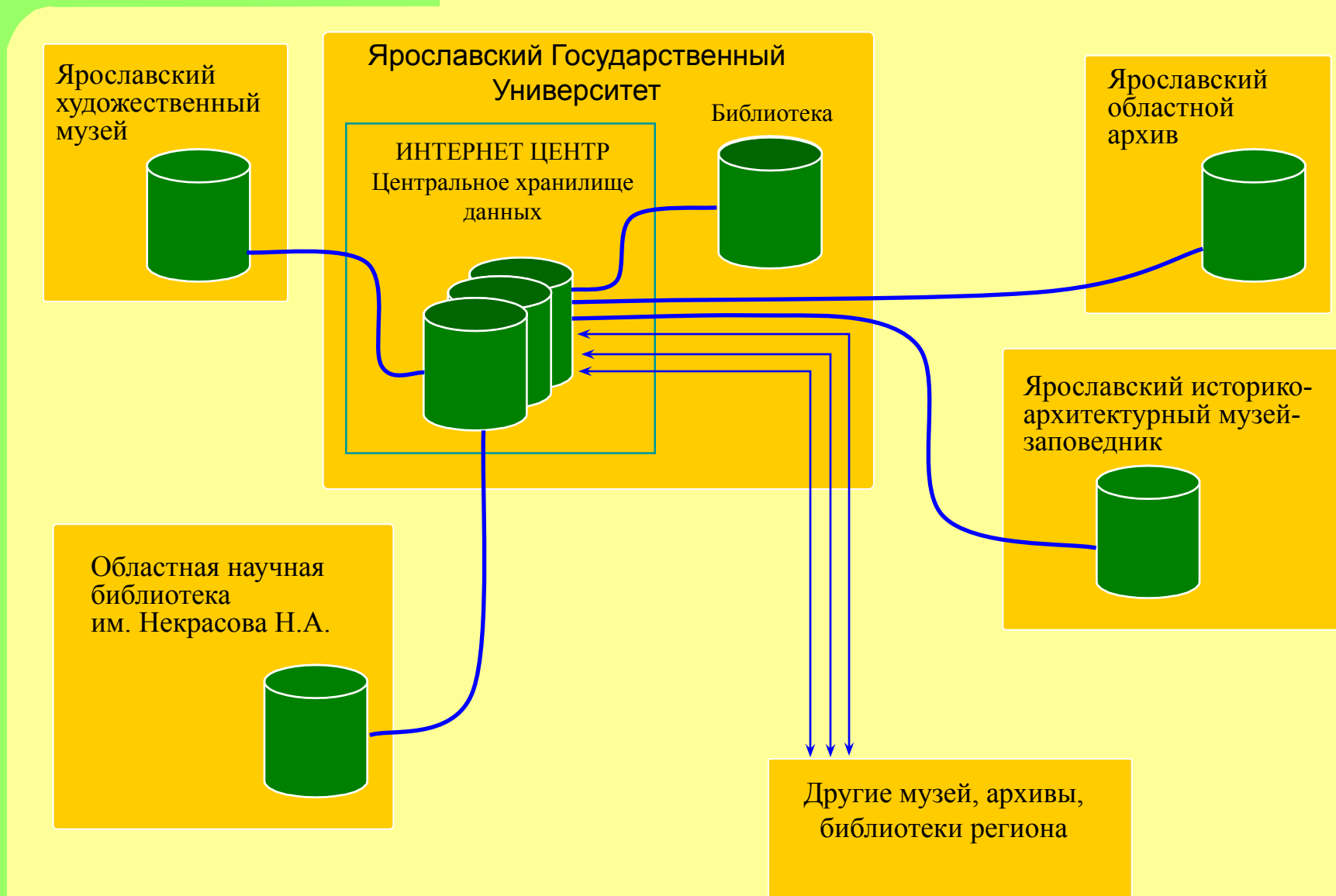
История вопроса

Проект создания Цифровой библиотеки Ярославского региона существует с 1998 года и ориентирован на решение информационных задач учреждений культуры, образования и науки и развивается на протяжении нескольких лет.

Изначально предполагалось, что в цифровой библиотеке (ЦБ) будет описано достаточно большое количество артефактов - физических объектов (предметов, документов, картин, персон, памятников архитектуры и т.д.) реального мира. Все объекты связаны друг с другом различными отношениями. При этом одним из основных условий было то, что описание артефактов изначально имеет нечеткую структуру. Т.е. формально полное описание структуры большинства объектов неизвестно или может меняться с течением времени. Более того, описание структуры может меняться уже после занесения некоторой части данных из предметной области. То же самое можно сказать и про связи между объектами.

В проекте принимали и принимают участие специалисты ведущих учреждений культуры, образования, науки.

Структура системы



Автоматизированная библиотечная информационная система



Введение в задачу

В настоящее время на рынке присутствует достаточно большое количество информационных систем (ИС) предназначенных для автоматизации учета фондов библиотек, ('Библиотека 2000', 'Ирбис', 'Marc-SQL') музеев ("КАМИС", АС "Музей"), архивов.

С точки зрения методологии построения, все эти системы являются специализированными приложениями

- Основу хранения и представления данных составляет стандартный для данной предметной области формат данных, например для библиотек MARC формат в различных вариациях.

- При моделировании предметных данных реализован ставший 'классическим' подход, при котором выделяют несколько стандартных сущностей, основными из которых являются книги, экземпляры коллекции, читатели, авторы и т.д. Собственно коллекции экземпляров этих сущностей и обрабатываются приложениями.

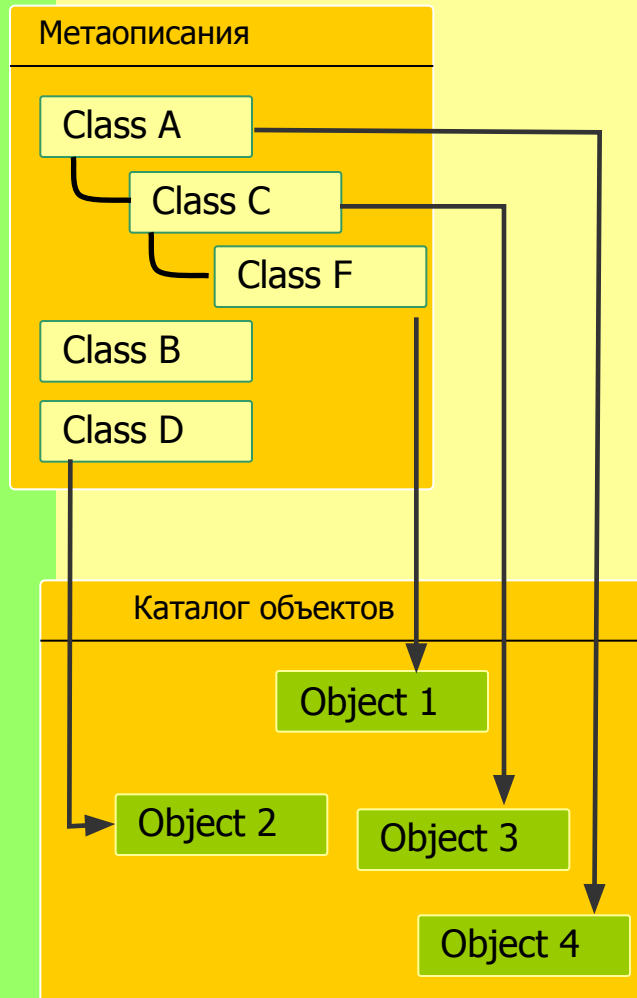
- Взаимодействие между сущностями осуществляется и обрабатывается на уровне бизнес логики. Подчеркнем, логические и семантические связи между объектами систем жестко заданы и поддерживаются на уровне бизнес логики (за исключением некоторых типов описания литературы, которые связываются на уровне MARC формата). Эти связи обычно определяются при проектировании системы на этапе постановки технического задания. Соответственно, создание новых типов связей, или их изменение, внесение дополнительных функций в бизнес логику связано с изменением системы (т.е. собственно с 'программированием') и доступно только непосредственно разработчикам.

Введение в задачу

Существующие подходы, являются достаточно эффективным для выполнения традиционных задач, таких как создание электронных каталогов литературы и документов, создание коллекций описаний музейных и архивных документов, выполнения бизнес процессов, связанных с обработкой массивов этих данных. Вместе с тем, при попытках расширить функциональность информационных систем для решения задач, которые возникают в настоящее время, появляются ощутимые проблемы. К таким задачам, в первую очередь можно отнести:

- хранение и обработка полных текстов документов, изображений;
- хранение альтернативных описаний или версии документов или экземпляров коллекций;
- обработка многоверсионности документов и значений их атрибутов;
- изменение в процессе работы стандартных описаний сущностей, данные о которых хранятся в системах;
- моделирование средствами системы новых сущностей с неизвестными заранее наборами атрибутов;
- изменение существующих, создание нестандартных связей между сущностями системы и поиск, извлечение данных с их учетом

Электронная библиотека - основа системы



Согласно поставленной задаче были предлагаются следующие принципы построения ЦБ и организации хранения данных.

Цифровая библиотека содержит метаописания артефактов в виде классов. Каждый класс определяется набором атрибутов и методов. Возможно наследование классов, объединяющее их в иерархии отношение "родитель->наследник".

Данные размещаются в экземплярах классов – объектах электронного каталога (ЭК). Объект содержит информацию о некоторой реальной сущности, согласно правилам, заданным описанием его класса. Объединение объектов электронного каталога в единую логическую структуру осуществляется при помощи связей. Связи могут иметь различный тип и определяться на различных уровнях (объект-объект, атрибут-объект, атрибут – атрибут и т.д.). Также на системном уровне поддерживается набор различных сервисов - интерфейсы доступа к данным, система авторизации и т.д.

Сеть объектов

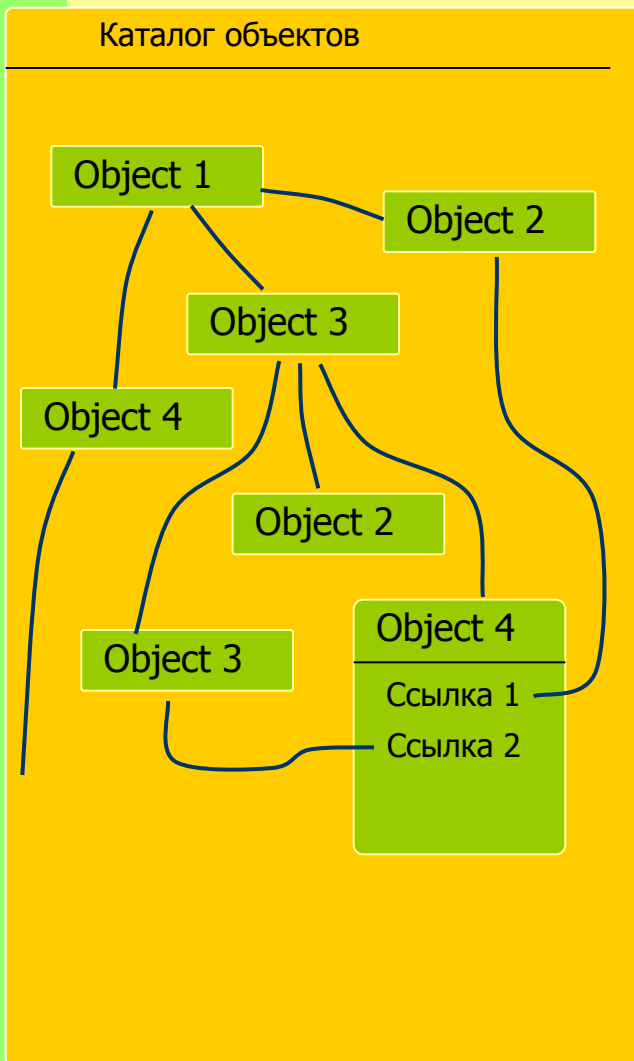


Специфика задачи создания ЦБ предполагает наличие между объектами большого количества связей. Поэтому на начальном этапе реализации проекта была выбрана следующая схема построения каталога.

Каждый объект ЭК мог иметь связь с любым другим объектом. Причем эти связи (между объектами) определялись не на уровне метаданных, а на уровне каталога объектов. Для обеспечения универсальных методов доступа к данным и уменьшения накладных расходов на поддержку ЭК представилось разумным наложить некоторые ограничения на возможные определения классов. Основное из таких ограничений - невозможность атрибутов объектного типа. Предлагалось заменить их отношениями объектов типа "главный-подчиненный".

Таким образом, ЭК каталог был реализован в виде некоторой сети равноправных объектов. Точкой входа в общем случае мог служить произвольный объект сети. Преимуществом такого подхода являлся большой универсализм и широкие возможности по моделированию данных произвольной структуры.

От сети объектов к иерархии



В процессе опытной эксплуатации выяснились существенные недостатки:

- Пользователи системы испытывали значительные трудности при навигации по сети объектов.
- Весьма существенными оказались трудности по администрированию объектов и определению и модификации связей объекта при его создании или изменении его атрибутов. Особенно это стало заметно, когда количество объектов в каталоге превысило значение 500. Фактически ЭК становился неуправляем.

Т.е. исходная схема, несмотря на свою универсальность, к сожалению оказалась мало пригодной для практической реализации и использования.

Подходы к организации связей между объектами были пересмотрены. Все объекты были объединены в жесткую иерархию. Электронный каталог при этом стал представлять собой дерево, в узлах которого находятся объекты, связанные между собой отношениями типа “главный-подчиненный”. Все остальные связи организуются при помощи ссылок, которые могут содержать атрибуты каждого объекта каталога на другие объекты и/или на их атрибуты.

Создание ‘виртуальных’ каталогов

К сожалению, описанный подход имеет ограничения связанные с жесткой иерархией расположения объектов.

Должна быть предусмотрена возможность создавать “виртуальные” каталоги объектов для конкретного пользователя (групп пользователей). Т.е. каждый объект может иметь некоторый набор связей с другими объектами, которые не отражаются в “основной” иерархии и описываются при его создании или модификации. Эти связи в общем случае могут быть как произвольны, так, и типизированы и организуют “сеть” объектов. Представление данных для конечного пользователя определяется выбранным им из существующего набора некоторым правилом (типом связей, набором и типом исходных объектов для просмотра и т.д.), по которому из исходного набора связанных сущностей формируется иерархический каталог объектов. При этом пользователь получает дерево объектов, оптимальное для решения его конкретных задач.

Повторное использование данных

Метаописания

Class **Персона**

Class **Сотрудник**

Class **Читатель**

Каталог объектов

Иванов
(class **Персона**)

Иванов
(class **Сотрудник**)

Иванов
(class **Читатель**)

Важной задачей, стоящей перед любым разработчиком информационной системы, является обеспечение эффективного повторного использования данных.

Для ЭК это особенно актуально, т.к. часто возникают задачи создания объектов, значения некоторого набора атрибутов которых совпадали бы со значениями соответствующих атрибутов другого объекта. Для решения этой проблемы была предложена концепция “наследования данных”

Положим, есть набор объектов, являющихся экземплярами класса “персона”. Атрибуты этих объектов содержат полную информацию о персональных данных. Положим далее, что при реализации какого-либо проекта понадобилось определить классы “сотрудник”, “пользователь библиотеки”, “исследователь”, порожденные от класса “персона” и отличающиеся от него наборами дополнительных атрибутов. Положим далее, что экземпляры порожденных классов будут описывать некоторое подмножество людей уже описанных объектами класса “персона”. В этом случае для создания новых объектов разумно уже использовать имеющуюся информацию.

Повторное использование данных

Важной задачей, стоящей перед любым разработчиком информационной системы, является обеспечение эффективного повторного использования данных.

Для ЭК это особенно актуально, т.к. часто возникают задачи создания объектов, значения некоторого набора атрибутов которых совпадали бы со значениями соответствующих атрибутов другого объекта. Для решения этой проблемы была предложена концепция “наследования данных”

Положим, есть набор объектов, являющихся экземплярами класса “персона”. Атрибуты этих объектов содержат полную информацию о персональных данных. Положим далее, что при реализации какого-либо проекта понадобилось определить классы “сотрудник”, “пользователь библиотеки”, “исследователь”, порожденные от класса “персона” и отличающиеся от него наборами дополнительных атрибутов. Положим далее, что экземпляры порожденных классов будут описывать некоторое подмножество людей уже описанных объектами класса “персона”. В этом случае для создания новых объектов разумно уже использовать имеющуюся информацию.

Проблемы и пути их решения

Сложность описания сущностей в ЭК для их пользователей

Например, правила библиографического описания книги (руководство по применению одного из MARC форматов) представляют собой солидную книгу. Понятно, что работник библиотеки не сможет без специальной подготовки самостоятельно сделать адекватную модель книги в виде описания класса ЭК.

Выход из этой ситуации возможен в использовании имеющихся на сегодняшний день представлений в виде xml-schema или профилей протокола Z39.50, которые уже прошли практическую апробацию и достаточно широко используются [11]. Задача решается путем создания механизма автоматического импорта описаний в том или ином виде в ЭБ и генерации на их основе определений классов.

Конечные пользователи просто используют уже готовые наборы определений классов ЭК или автоматически их создают с использованием сервисов ЭБ.

Проблемы и пути их решения

Скорость доступа и поиска данных в ЭК

Авторы предлагают совместное использование вариантов поиска, основанных на построении справочных индексных таблиц (по значению атрибута) и поиска с учетом связей между объектами и их атрибутами.

Механизм построения индексных таблиц является широко распространенным в настоящее время. Его эффективность для поиска и извлечения данных подтверждена практикой. Применительно к ЭБ особенностью (и во многих случаях преимуществом) является то, что таблицы могут строиться автоматически по значениям атрибутов выделенных множеств объектов. Идентификация объектов на принадлежность множеству может выполняться по нескольким критериям, таким как класс объекта, положение объекта в иерархии каталога, наличие связей с заданными объектами. Подчеркнем, что все эти критерии могут быть автоматически обработаны на уровне ЭК. Например, при создании объекта класса 'персона' в индексной таблице 'ФИО', настроенной на все объекты этого класса, возникнет новая учетная запись.

Поиск так же можно осуществлять с учетом связей объектов. Фактически, в этом случае, индексной таблицей выступает набор всевозможных путей на заданном множестве объектов и/или их атрибутов. Простейшим примером такого поиска является извлечение всех объектов, зависящих от указанного..

Проблемы и пути их решения

Возможности интеграции с существующими системами учета.

Т.к. важнейшей частью любой информационной системы являются содержащиеся в ней данные, то для успешного функционирования ЭБ необходима возможность непосредственного доступа к данным ИС музея или библиотеки с возможностью автоматического конвертирования информации в обе стороны. При решении задач такого рода всегда возникают трудности (описанные выше) совмещения универсальных и специализированных подходов. Так для систем автоматизации библиотек это требование выливается в необходимость обработки MARC форматов обмена библиографическими данными на уровне ядра электронной библиотеки.

Предлагается решить эту задачу, поместив описание соответствующих форматов непосредственно в объекты электронного каталога. В общем случае можно выделить несколько уровней интеграции

- 1) Интеграция на основе ссылки 1:1 (запись - объект). При этом на одну MARC запись ссылается только один объект ЭБ (каждая MARC запись ссылается на один объект ЭБ)
- 2) Интеграция на основе ссылки 1:М (запись - объекты). Одна MARC запись может ссылаться на фиксированный набор из одного или нескольких объектов ЭБ заранее известных типов (например на объекты классов “автор”, “организация” и т.д.).
- 3) Интеграция на основе семантического анализа содержимого MARC записи.

Заключение

Представленные в докладе подходы воплощаются в практические решения группой разработчиков технического центра ЯрГУ в рамках выполнения проектов поддержанных РФФИ (03-07-90178в) и РГНФ (04-06-12016в).

paley@yars.free.net