

Михалева Н.Д., Полтавцев А.А. Использование архитектуры Workflow-систем для построения приложений электронного документооборота. // Проблемы информатики в образовании, управлении, экономике и технике: Сб. статей XV Междунар. научно-техн. конф. – Пенза: ПДЗ, 2015. – С. 185-189.

УДК 004

ИСПОЛЬЗОВАНИЕ АРХИТЕКТУРЫ WORKFLOW-СИСТЕМ ДЛЯ ПОСТРОЕНИЯ ПРИЛОЖЕНИЙ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА

Н.Д. Михалева, А.А. Полтавцев

THE USING ARCHITECTURE OF WORKFLOW-SYSTEM WORKS FOR BUILDING ELECTRONIC WORKFLOW APPLICATIONS

N.D. Mikhaleva, A.A. Poltavtsev

Аннотация. Рассматриваются характеристики Workflow-систем, которые имеют критическую важность при реализации и эксплуатации приложений электронного документооборота. Описана схема взаимодействия Workflow-компонент при построении ЭД приложений коммерции.

Ключевые слова: документооборот, электронный документ, Work-flow система, приложения ЭД.

Abstract. Considered the characteristics of Workflow systems that are critical in the implementation and operation of electronic document applications. Given the scheme of interaction Workflow-component in the construction of ED-commerce applications.

Keywords: document flow, electronic document, Workflow-system, ED appendices.

1. Существует огромное множество бизнес-моделей, используемых в электронном документообороте (ЭД). Во всех этих моделях бизнес-процессы могут быть смоделированы как набор predetermined шагов, в рамках которых происходит управление взаимосвязанными потоками данных. Это соответствует Workflow-процессу, где координация, управление и взаимосвязь задач автоматизированы, хотя сами задачи могут быть автоматизированы или выполнены людьми [1].

2. Workflow-системы, которые могут быть использованы в ЭД, должны обладать определенными характеристиками, имеющими критическую важность при реализации и эксплуатации приложений электронного документооборота [2]:

3. Большое количество участников бизнес-процесса, занятых в прикладных задачах ЭД, определяют высокие требования к производительности Workflow-систем. Необходимо поддерживать распределение загрузки (Load balancing), чтобы гарантировать этот вид масштабируемости приложений. Кроме того, Workflow-системы для приложений ЭД должны иметь возможность быстро адаптироваться к сетевым изменениям.

4. Workflow-системы для ЭД приложений должны легко приспосабливаться к различным изменяющимся требованиям клиентов. Поэтому Workflow-модель, скорее всего, должна быть представлена шаблоном, в котором наряду со стандартными действиями управления и манипуляций потоками данных необходимо реализовать дополнительные возможности изменения структурной схемы бизнес-процесса и динамического подключения дополнительных прикладных задач.

5. ЭД процессы должны быть многофункциональны. Чтобы соответствовать этому требованию, они должны быть в состоянии выполняться в средах с недостаточными ресурсами и иметь открытую архитектуру. Эта функциональность Workflow-системы должна быть толерантной к потребностям заказчика и требуемым им ресурсам. Кроме того, такая система должна иметь возможность выполняться через Internet и быть основана на открытой и интероперабельной инфраструктуре.

6. Частые сбои в работе, и недоступность Workflow-серверов немедленно замедляют или просто останавливают работу ЭД системы. Поэтому требуется организовать эффективное выполнение функций восстановления после сбоев и минимизировать время ответов Workflow-серверов на запросы клиентов.

Монолитные Workflow-системы не в состоянии выполнять эти требования. Приходится даже для очень простых Workflow-процессов запускать мощные многомасштабные Workflow Engines (Диспетчера потоков работ), требующие дополнительных ресурсов. Как правило, это приводит к тому, что в таких системах не реализуются:

- балансировка загрузок;
- эффективное восстановление серверов после сбоев;
- изменение экземпляров Workflow-процессов в процессе функционирования.

Предлагаемая базовая архитектура Workflow-системы, предназначенная для реализации приложений ЭД, спроектирована таким образом, чтобы соответствовать перечисленным требованиям (рисунок):

1. Каждый из экземпляров классов базового пакета (Event, Workflow, Task, Workflow Manager, Arc и др.) может быть представлен как CORBA-объект. В этих объектах содержится вся необходимая информация не только о потоках системных и прикладных данных, но и данные об истории выполнения управляющих событий обрабатываемыми объектами. Использование CORBA дает возможность мигрировать эти объекты в сеть, обеспечить балансировку загрузок, динамически изменять базовую схему Workflow-процесса в режиме выполнения приложения. Кроме того, сбой в работе или отсутствие доступа к некоторому Workflow-серверу приведет лишь к невозможности функционирования конкретных экземпляров задач или процессов, выполняющихся только на этом сервере [3].

2. Спроектированная система состоит из набора взаимосвязанных функциональных компонентов: Event Manager, Workflow Manager (Workflows), Task Agent, Monitoring Subsystem, Administration Subsystem и др. Такая компонентная архитектура дает возможность объединять функциональность всех подсистем и использовать их сложность только когда это действительно нужно. В процессе выполнения приложения создаются и загружаются только те экземпляры компонентов, которые необходимы системе в данный момент. В результате, резко возрастает эффективность использования сетевых и системных ресурсов.

3. Клиентские приложения могут быть запрограммированы в виде Java-апплетов. Таким образом, конечный пользователь может получить необходимые Workflow-компоненты через сеть, не устанавливая дополнительного программного обеспечения на своем компьютере. Поскольку такие компоненты могут модифицироваться на серверной стороне (прозрачно для конечного поль-

зователя), то это значительно повышает мобильность пользовательских приложений и упрощает их использование.

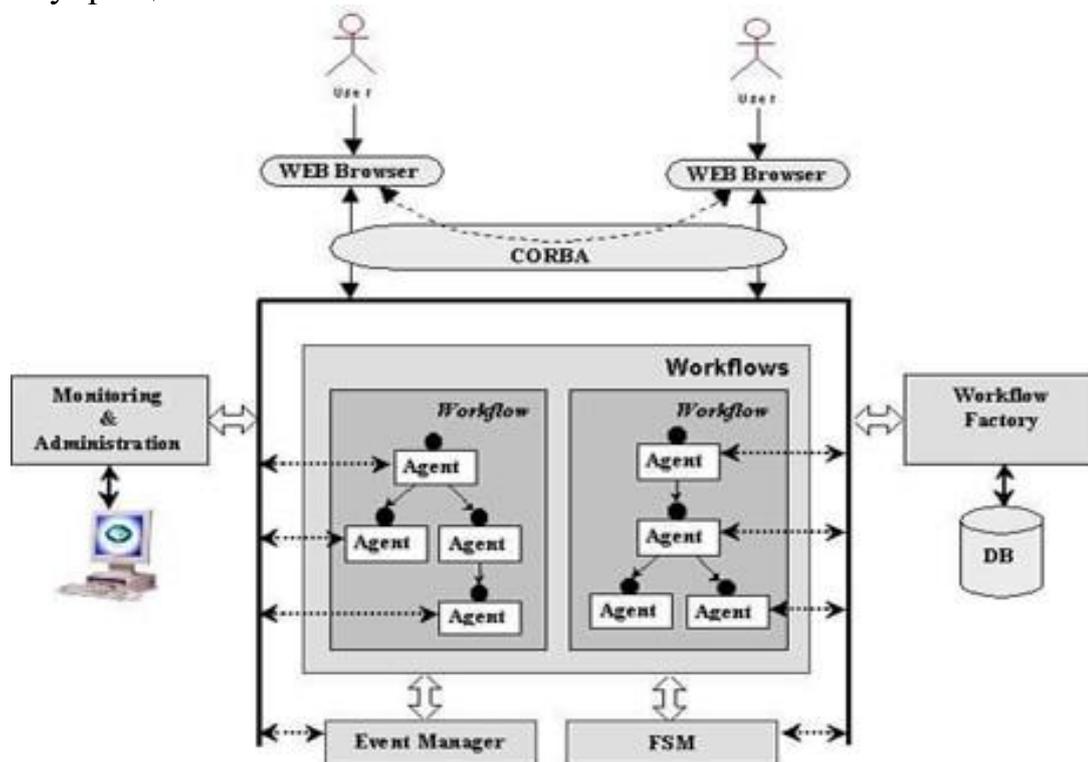


Схема взаимодействия Workflow-компонент при построении ЭД приложений

4. При использовании распределенной инфраструктуры приложения, построенного с использованием CORBA, вызов методов CORBA-объектов можно осуществлять при помощи XML. В этом случае нет необходимости размещать ORB на клиентской стороне. Клиент может послать запрос на вызов методов удаленного объекта и некоторые параметры этого запроса, выраженные в XML, используя «post» метод HTTP протокола. HTTP сервер, получив этот запрос, может передать его CORBA-серверу через CGI. И затем CORBA-сервер, разобрав полученный XML-документ, вызовет требуемый метод. Если же ORB размещен на клиентской стороне, то клиент может послать запрос напрямую CORBA-серверу. Использование такой технологии и стандарта XML позволяет строить интегрируемые, открытые, интероперабельные приложения.

Библиографический список

1. Dogac A., Durusoy I., Arpinar S., Tatbul N., Koksall P., Cingil I., Dimililer N. A Workflow-based Electronic Marketplace on the web.
2. Muth P., Weissenfels J., and Weikum G. What Workflow Technology Can Do for Electronic Commerce, in Current Trends in Database Technology, A. Dogac, T. Ozsu, O. Ulusoy, editors, Idea Group Publishing, 1998.
3. Orfali R., and Harkey D. The Essential Client/Server Programming with JAVA and CORBA, John Wiley, 1997.

Михалева Надежда Дмитриевна

Тверской государственной

технический университет,

г. Тверь, Россия

E-mail: nadyamihaleva@mail.ru

Полтавцев Анатолий Алексеевич

Тверской государственной

технический университет,

г. Тверь, Россия

E-mail: aapolt@gmail.com

Mikhaleva N.D.

Tver State Technical University,

Tver, Russia

Poltavtsev A.A.

Tver State Technical University,

Tver, Russia