

МЕТОДИКА СОЗДАНИЯ ВОЗОБНОВЛЯЕМЫХ РАБОЧИХ ПРОЦЕССОВ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ

Шульга Е. С., Филипук Д. С., Сурков К. А.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: egorshulga@outlook.com, filipukd@mail.ru, kirill.surkov@gmail.com

Рассматривается выполнение рабочих процессов в автоматизированных системах. Сбор информации о ходе их выполнения позволяет анализировать успешность выполнения их этапов. После преднамеренного или непреднамеренного отказа выполнение рабочего процесса можно продолжить с этапа, следующего за последним успешно завершённым.

ВВЕДЕНИЕ

Типичным предметом автоматизированной системы масштаба предприятия является рабочий (производственный) процесс. Существуют рабочие процессы, выполняемые целиком на одном компьютере, и существуют рабочие процессы, выполнение которых распределено среди нескольких взаимодействующих компьютеров вычислительной сети. В случае применения распределенных рабочих процессов неизбежно возникают риски по отказам взаимодействующих программных средств и каналов взаимодействия. В связи с этим появляется задача создания возобновляемых рабочих процессов, выполнение которых может быть остановлено на любом этапе и в любой точке, а затем продолжено до успешного завершения.

I. ПОСТАНОВКА ЗАДАЧИ

Далее будем использовать следующие определения:

Рабочий процесс – набор действий для достижения определенной цели. Процессом может быть регистрация пользователя в системе, выполнение определенных вычислений и т.д.

Этап – независимая составная часть рабочего процесса. Этапы объединяются в некоторую последовательность, которая может быть представлена однонаправленным графом без петель. При этом в графе может быть один или несколько начальных, а также один или несколько конечных этапов.

Действие – атомарный элемент процесса, составная часть этапа. Действия могут выполняться последовательно, параллельно, комбинированно или не выполняться вовсе.

Выполнение процессов пользователем можно представить в виде дискретно-детерминированного конечного автомата [1].

Пример процесса приведен на рисунке 1. В данном процессе существует один начальный этап (этап 1), в него входит набор действий: А, В, С, D, Е. Для остальных этапов наборы действий опущены. Примеры ветвления: ветвь этапов 2.1,

3 и этапа 2.2, ветви этапов 5.1, 5.3 и с этапа 5.2 по этап 6. Наличие ветвлений означает, что некоторый компонент автоматизированной системы будет принимать решение, по какому пути графа будет выполняться рабочий процесс. Кроме того, в приведенном примере существует несколько завершающих этапов: этапы 5.1, 5.3 и 6. На данном примере рассмотрим методику создания возобновляемого процесса.

Предполагается, что все компоненты системы имеют возможность хранения некоторой информации. Серверные компоненты могут сохранять ее в базах данных. Клиентская часть сохраняет лишь небольшую часть информации, необходимую для ее работы.

Для внедрения возможности возобновления этапы процесса должны быть независимы друг от друга: обработка действий на каждом этапе должна завершаться сохранением необходимых данных и передачей управления следующему этапу. Данные между этапами могут передаваться через постоянные хранилища (базы данных) или сохраняться в клиентских компонентах автоматизированных систем (при поддержке ими такой возможности).

II. СБОР ИНФОРМАЦИИ О РАБОЧЕМ ПРОЦЕССЕ

По ходу выполнения рабочего процесса необходимо вести сбор данных для анализа времени, успешности выполнения рабочих процессов, этапов и действий, анализ прочей статистики. Кроме того, собранная информация может быть использована для возобновления прерванных рабочих процессов.

Для реализации сбора данных в автоматизированной системе предлагается создание специального компонента и использование постоянного хранилища. Пользователи и другие компоненты могут заниматься выполнением различных рабочих процессов, каждый из которых может содержать различное число этапов с многочисленными действиями. Создание записей о выполнении частей рабочего процесса будет проис-

ходить очень часто, в то время как их чтение, создание отчетов, создание заявок на возобновление рабочих процессов будет происходить намного реже. В связи с этим программный интерфейс компонента сбора данных должен быть максимально упрощён, а именно: он должен содержать единственный метод, который не осуществляет сложных вычислений, а лишь записывает в базу данных переданную ему информацию о выполненном действии.

Все рабочие процессы в автоматизированной системе должны быть описаны в терминах этапов и действий. Эта информация должна быть известна в каждом компоненте. Таким образом, сбор данных о выполненных действиях является сквозной функциональностью (cross-cutting concern) [2].

Для реализации возможности возобновления в каждом рабочем процессе введём дополнительное псевдодействие: «Окончание этапа N». Информация об этом действии должна быть сохранена на каждом этапе как маркер завершения.

Для поддержки ветвления также необходимо сохранять маркерные действия. Если после определенного этапа выполнение рабочего процесса может пойти по одной из веток, то автоматизированная система должна вынести решение об используемой ветке и затем записать через компонент сбора информации о процессе действие «Начат этап M». В примере рабочего процесса на рисунке 1 после выполнения этапа 1 должно быть записано одно из действий: «Начат этап 2.1» или «Начат этап 2.2». Аналогично после этапа 4 сразу же должно быть записано одно из действий: «Начат этап 5.1», «Начат этап 5.2» или «Начат этап 5.3».

III. ВОЗОБНОВЛЕНИЕ РАБОЧЕГО ПРОЦЕССА

Если рабочий процесс прерван, то он может быть возобновлён. Прерывание рабочего процесса может произойти по ряду причин:

- непреднамеренно при отказе одного из компонентов автоматизированной системы;
- непреднамеренно при сбое связи между компонентами;
- преднамеренно при отказе пользователя продолжать рабочий процесс.

Для возобновления прерванного рабочего процесса предлагается следующий алгоритм:

1. Найти последний этап с записанным маркерным действием «Окончание этапа S_i ».
2. Если после этапа S_i следует единственный этап S_{i+1} , то он и является следующим для выполнения. Процесс нужно продолжить именно с него.
3. Если после этапа S_i следует несколько этапов S_{i+1}^k , то необходимо проверить, есть ли записанные действия в каждом из них.
4. Искомым этапом для возобновления процесса является первый с непустым множеством записанных этапов.
5. Если ни для одного из этапов S_{i+1}^k не записано действий, то необходимо вернуться на один этап назад. Этапом для восстановления процесса будет S_i .

Запуск механизма возобновления может происходить различными способами. В случае отказов серверные компоненты могут периодически совершать опрос на предмет восстановления функционирования других компонентов автоматизированной системы. В случае преднамеренной остановки процесса пользователем можно отправить ему уведомление с предложением закончить выполнение рабочего процесса.

IV. ЗАКЛЮЧЕНИЕ

Описанная методика позволяет создавать возобновляемые рабочие процессы в автоматизированных системах. Возобновление выполняется после преднамеренных или непреднамеренных отказов. Предлагается внедрение специального компонента по сбору информации о выполнении процессов в системе, который способен обработать большое число одновременных заявок на сохранение данных. В случае прерывания рабочего процесса система не оказывается в неконсистентном состоянии, и выполнение рабочего процесса можно продолжить на следующем этапе за последним успешно завершённым этапом.

V. СПИСОК ЛИТЕРАТУРЫ

1. Моделирование систем / Б. Я. Советов, С. А. Яковлев. – М.: Высш. шк., 2001. – 343 с.
2. Kiczales, G. Aspect-Oriented Programming / G. Kiczales, J. Lamping // Proceedings of the 11th European Conference on Object-Oriented Programming (ECOOP 1997) – Jyväskylä, Finland, 1997. – P. 220–242.

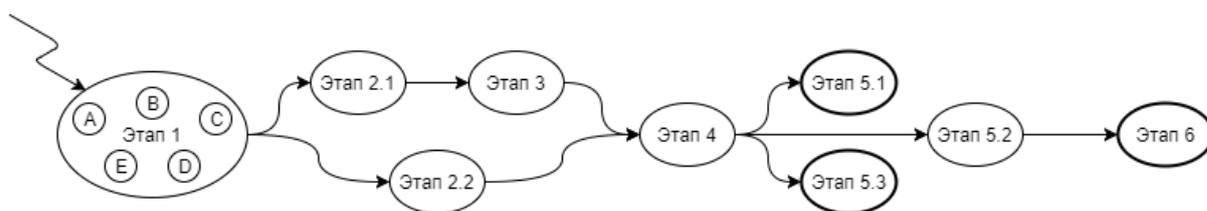


Рис. 1 – Пример процесса