

# Система паралельної оцінки ризиків програмних проектів

Ірина Кириєнко

Кафедра інформаційних та комп'ютерних систем, Чернігівський державний технологічний університет, УКРАЇНА,  
м.Чернігів, вул.Шевченка, 95, E-mail: irakirienko@gmail.com

*Description and results of creation of risk of programmatic projects management system is resulted in the article. This system allows to execute the parallel estimation of all types of risks of programmatic projects with the in advance set exactness of estimation.*

Risk of programmatic project management system, programmatic project, parallel estimation, object model, object, inheritance.

## I. Вступ

Створення будь-якого програмного продукту пов'язане з ризиками – це, перш за все, ризики виходу за рамки бюджету або розкладу, ризики використання нових технологій і мов програмування, а також багато інших. Всі вони ставлять під загрозу досяжність цілей проекту. Саме з цієї причини процеси управління ризиками є одними з найважливіших складових процесів управління проектами і ухвалення рішень в рамках проектів, оскільки вони дозволяють максимізувати позитивні і мінімізувати негативні наслідки настання ризикових подій.

## II. Аналіз задачі

По відношенню до програмних проектів ризиками називають невизначені події, що негативно або позитивно впливають або не впливають на хід розвитку проекту. Всі ризики програмних проектів можна класифікувати наступним чином:

- ризики, пов'язані з вимогами;
- технологічні ризики;
- ризики, пов'язані з кваліфікацією персоналу;
- політичні ризики.

Для управління перерахованими категоріями ризиків існують такі етапи управління ризиками, як:

1. Планування управління ризиками — вибір підходів і планування діяльності по управлінню ризиками проекту;
2. Ідентифікація ризиків — визначення ризиків, здатних вплинути на проект, і документування їх характеристик;
3. Якісна оцінка — якісний аналіз ризиків і умов їх виникнення з метою визначення їх впливу на результат проекту;
4. Кількісна оцінка — кількісний аналіз умовірності виникнення і впливу наслідків ризиків на проект;
5. Планування реагування на ризики — визначення процедур і методів з послаблення негативних наслідків ризикових подій і використання можливих переваг;
6. Моніторинг і контроль ризиків — моніторинг ризиків, визначення ризиків, що залишаються, виконання плану управління ризиками і оцінка ефективності дій з мінімізації ризиків [1].

Оскільки етапи управління ризиками ідентичні для всіх категорій ризиків програмних проектів, то стає

можливим створення системи управління ризиками, яка б виконувала одночасну оцінку всіх ризиків. Ця оцінка полягає у визначенні ймовірності настання несприятливої події, пов'язаної з проектом, прогнозуванні величини втрат від настання цієї події та оцінці її вартості.

Проаналізувавши данні досліджень успішності проектів в сфері інформаційних технологій з 1994 року можна помітити, що в 2008 році 18% проектів провалилось, більш ніж половина – 51% не вклалися в заданий бюджет і терміни, і тільки 31% були визнані успішними [2]. Як відомо, вихід за рамки бюджету і термінів – це одні з найпоширеніших категорій ризиків програмних проектів. Саме тому, виходячи з досліджень, можна зробити висновок про те, що визначальну роль для успіху будь-якого програмного проекту має рівень управління ризиками в цьому проекті.

Аналіз систем управління ризиками програмних проектів показує, що на сьогоднішній день не існує універсальної, вільно поширюваної і доступної системи, яка б одночасно виконувала оцінку декількох категорій ризиків з наперед заданою точністю. Розглянуті системи, такі, як система управління ризиками Welcomrisk розроблена компанією Welcom Software Technologies [3] і розділ управління ризиками в рамках Microsoft Enterprise Project Management Solution (EPM) [4], використовують для оцінки ризиків лише один математичний апарат, що обмежує точність обчислень певним значенням і орієнтований на конкретні види ризиків. Крім того, ці системи розраховані на взаємодію з конкретною наперед заданою системою управління проектами. Актуальним є створення системи управління ризиками, яка б не була прив'язана ні до однієї системи управління проектами, а надавала API для взаємодії з будь-якою системою управління проектами, що дозволить використовувати її в будь-якому проекті, незалежно від того, яка сама система управління проектами для нього використовується.

## III. Структура системи

Запропонована система включає такі процеси управління ризиками:

- ідентифікація — процес полягає в визначенні ризиків, що безпосередньо впливають на проект;
- введення і опис ризиків — процес полягає у вказанні назви і визначенні категорії ризиків;
- вибір можливого типового ризику з бази даних — можливий такий випадок, що той чи інший ризик вже існує в базі даних системи, тоді користувач не матиме необхідності додатково вводити всю інформацію про ризик, достатньо буде вибрати необхідний ризик із бази даних і вказати його нові параметри;

—введення причини виникнення ризику — як тільки ризик введений, необхідно вказати точну причину його виникнення, оскільки вона матиме важливе значення при визначенні дій в разі необхідності його ліквідації;

—вибір власника ризику — процес полягає в визначенні зі списку задіяних в проекті осіб особи, котра буде виконувати діяльність щодо ліквідації чи мінімізації ризиків в разі виникнення такої необхідності;

—визначення експертів — вибір експертів, головною задачею яких буде контроль за діяльністю власника ризику;

—якісний аналіз — процес полягає в визначенні впливу ризиків на результат проекту;

—кількісний аналіз — процес полягає в безпосередній оцінці імовірності виконання ризиків;

—розрахунок значущості ризиків — полягає в побудові згідно з пріоритетами головної таблиці ризиків;

—порівняння значущості з рівнями толерантності - ризику, що не представляють собою загрозу для проекту, можуть прийматися, інші ризики можуть лише мінімізуватися, а є і такі, котрі обов'язково необхідно ліквідувати. Саме на етапі порівняння значущості з рівнем толерантності і визначаються дії, які необхідно застосувати до того чи іншого ризику;

—визначення заходів щодо ліквідації або мінімізації ризиків — безпосереднє визначення які саме дії необхідно застосувати до того чи іншого ризику.

Для представлення ризиків, подій, математичних апаратів і зв'язків між ними в запропонованій системі була розроблена об'єктна модель. Базовий об'єкт ризику містить параметри, характерні всім видам ризиків, такі як імовірність, загроза, тимчасові обмеження та ін., а похідні об'єкти – чинники, характерні конкретному виду ризику.

Об'єкти подій містять такі властивості як дату, назву, опис та інші. Математичний апарат є об'єктом, який виконує оцінку ризиків згідно із заданим алгоритмом. Ці об'єкти підключаються до системи як бібліотеки-плагини, дозволяючи тим самим отримати точнішу оцінку ризику, ніж та, яку надає застосований за умовчанням в системі математичний апарат.

Результатом виконання ризику є подія, при виникненні якої необхідний одночасний перерахунок імовірності інших ризиків. Оскільки чинники, необхідні для розрахунку одного ризику, не впливають на чинники інших, то стає можливим використання паралельних обчислень для одночасної оцінки ризиків. Паралельна оцінка полягає в розрахунку кожного виду ризику в незалежному потоці і подальшій побудові головної таблиці ризиків, яка містить інформацію про причини виникнення ризиків, потенційні наслідки і очікувану величину ризику.

Інформація для даної таблиці розраховується виходячи з ймовірності, загрози і додаткових, індивідуальних для кожного виду ризику, параметрів. Більшість з цих даних система бере з системи управління проектами, з якою вона взаємодіє, і лише деякі дані вводяться користувачем.

## Висновок

Таким чином, запропонована система дозволяє ефективно проводити оцінку ризиків програмних проектів, виконуючи її паралельно, що важливо при високій динаміці зміни ризиків. Використання об'єктної моделі істотно спрощує обробку ризиків і математичних апаратів, дозволяючи додавати в систему нові об'єкти, що дає можливість вільно розширювати систему, додаючи в неї нову функціональність. Таке розбиття на процеси дозволяє застосовувати дану систему для будь-якої моделі життєвого циклу розробки програмного забезпечення, наявність API дозволяє системі взаємодіяти з будь-якою системою управління проектами.

[1] Audrey J. Dorofee, Julie A. Walker, Christopher J. Alberts et al, Continuous Risk Management Guidebook.

[2] CHAOS Research Results, 2008 Third Quarter Research Report. The Standish Group International, Inc., 2008.

[3] WelcomRisk - портал управління ризиками, URL: [//subscribe.ru/archive/economics.school.aproject/200610/16163620.html](http://subscribe.ru/archive/economics.school.aproject/200610/16163620.html)

[4] @Risk - Professional for Excel, URL: <http://www.prodmag.ru/catalog/info/245>.