

УДК 004.4

Мельниченко О.В.

Хмельницький національний університет

МЕТОД ТА ПІДСИСТЕМА САМОВІДНОВЛЕННЯ ПІСЛЯ КРИТИЧНИХ ЗБОЇВ

Система самовідновлювання після критичних збоїв є комплексним врахуванням вимог розподіленості, децентралізованості та багаторівневості до розробленої архітектури інформаційної системи. Впровадження системи самовідновлювання на основі цих вимог дає змогу створити розподілені системні компоненти, які функціонують автономно і самостійно приймають рішення про самовідновлення після критичного збою. Ключовою характеристикою, яка впливає на рівень функціонування програмної системи загалом є час перебування в критичних станах. Події перебування частини програмних модулів в одному й тому ж системному стані є сумісними подіями. Водночас перехід різних програмних компонентів системи до одного стану всієї системи не змінює функціоналу цих компонентів, що свідчить про їхню незалежність один від одного. Результатом є запропонований механізм самовідновлення після критичних збоїв. Де одним із основних принципів досягнення ефективності під час роботи інформаційної програмної системи є побудова розподілених та самоорганізованих компонентів.

The system of self-recovery after critical failures is a comprehensive consideration of the requirements of distribution, decentralization and multi-levelness to the developed architecture of the information system. The implementation of a self-healing system based on these requirements makes it possible to create distributed system components that function autonomously and independently make decisions about self-healing after a critical failure. The key characteristic that affects the level of operation of the software system in general is the time spent in critical states. The events of a part of the program modules being in the same system state are compatible events. At the same time, the transition of various software components of the system to one state of the entire system does not change the functionality of these components, which indicates their independence from each other. The result is a proposed self-healing mechanism after critical failures. Where one of the main principles of achieving efficiency during the operation of an information software system is the construction of distributed and self-organized components..

Якщо час T перебування в певному стані дуже тривалий стосовно часу інших станів та програмної системи загалом, що зумовлено зовнішніми чинниками (з'єднання мережі, відгук від сторонніх інтегрованих систем тощо), тоді така подія буде врахована, як небезпечна. Автономний системний компонент, який перебуває у критичному стані формує відповідний сигнал критичного збою усім іншим програмним модулям. Після отримання повідомлення, усі інші програмні модулі переходять у стан збереження поточної інформації, яку вони обробляють. Оскільки всі програмні компоненти є самостійними, але підключаються через програмні

інтерфейси у централізовану систему управління, важливим елементом визначення категорії критичного збою є формування системного журналу подій. Головною особливістю поділу критичних збоїв на категорії є можливість побудови системи самовідновлення для кожного програмного компонента окремо. Аналізуючи отриману інформацію, центральний програмний модуль приймає рішення про спробу відновити критичний стан того чи іншого системного компонента, або перевести програмну місію у стан аварійного завершення. Характерним є те, що тривалість самовідновлення залежить від типу системного компонента.

Оцінка спроможності на самовідновлення визначена через термін очікування системного компонента у критичному стані. Системний компонент, який відповідає за обчислення даних та пов'язаний з іншим стороннім інтерфейсом, переходячи в критичний стан, може перебувати у ньому до 2 хвилин. Таким чином, кожен програмний модуль очікуватиме зворотної відповіді для початку самовідновлення, що виражається обробкою даних, які були збережені у момент переходу системного компонента у критичний стан. Якщо термін очікування переходить за межі 2 хв, централізована система реагує на таку подію, як недопустимість продовження виконання програмної місії і посилає всім програмним компонентам сигнал про аварійне завершення місії..

Оскільки завдання від центральної програмної системи представляються процесами, то кожен програмний компонент самостійно їх аналізує та приймає рішення щодо їхнього виконання. Зазначимо, що ці процеси можуть призвести не тільки до зниження продуктивності виконання програмної місії, але і до взаємоблокування. Для недопущення таких дій більшість програмних компонентів наділені властивостями перевіряти життєву діяльність окремих модулів, які мають залежність від сторонніх систем та мережевого з'єднання. Тому ключовою вимогою до можливостей центральної програмної системи є здатність до планування виконань процесів без суттєвого впливу сторонніх чинників та встановлення такого порядку надходження всіх процесів, що не допускати блокування взаємозалежних програмних модулів..

Отже, запропонований механізм самовідновлення після критичних збоїв є одним із основних принципів досягнення ефективності під час роботи інформаційної програмної системи, побудованих на основі розподілених та самоорганізованих компонентах. Використання розроблених програмних модулів із системами самовідновлення дає змогу організувати підтримку цілісності системи та здійснення передачі знань між окремими структурними компонентами. Отримані аналітичні залежності рівнів функціонування програмних модулів забезпечують динамічний вплив поточних станів та впливати на подальші дії інших компонентів та переведення їх у робочі стани. Запропонований механізм самовідновлення є основою для поєднання усіх автономних програмних компонентів, що забезпечує роботу самоорганізованої системи для динамічного отримання зображень у тривимірному просторі.