

ПАТТЕРНЫ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ

М. К. Буза, О. М. Кондратьева

Белорусский государственный университет

Минск, Беларусь

e-mail: bouza@bsu.by, kondratjeva@bsu.by

Представлены результаты исследования, посвященного паттернам параллельного программирования, и программный инструмент для автоматизации проектирования параллельных программ с общей памятью.

Ключевые слова: шаблон проектирования; паттерн параллельного программирования; автоматизация распараллеливания.

PATTERNS FOR PARALLEL PROGRAMMING

M. K. Bouza, O. M. Kondratjeva

Belarusian State University

Minsk, Belarus

We present the results of a study on the patterns of parallel programming, and software tools to automate the design of parallel programs with shared memory.

Keywords: design pattern; a pattern of parallel programming; automated parallelization.

ВВЕДЕНИЕ

В задачах параллельного программирования главная цель – сокращение реального времени расчетов на многопроцессорной вычислительной системе, а также времени, потраченного на разработку параллельного приложения.

При разработке параллельной программы важным является выбор технологии параллельного программирования. При ее выборе рекомендуется оценить следующие характеристики: возможность создания эффективных программ, возможность быстрого создания параллельных программ и гарантии сохранения эффективности параллельной программы при ее переносе с одного компьютера на другой.

Эффективные программы для высокопроизводительных параллельных компьютеров обычно обладают высокой степенью сложности реализации. Поэтому в настоящее время ведется значительная работа по упрощению процесса проектирования и реализации параллельных алгоритмов. В частности, создаются программные инструменты для поддержки проектирования, разработки и исполнения параллельных приложений в различных вычислительных средах. Эти инструменты ориентированы на упрощение работы пользователя на параллельной вычислительной системе и представляют визуальные инструменты конструирования приложений или даже полнофункциональные среды разработки параллельных приложений.

Представлены результаты исследования, посвященного паттернам параллельного программирования, и авторский программный инструмент для автоматизации проектирования параллельных программ с общей памятью.

ПАТТЕРНЫ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ

Шаблон проектирования, или паттерн, в разработке программного обеспечения – повторяемая архитектурная конструкция, представляющая собой решение проблемы проектирования в рамках некоторого часто возникающего контекста. Обычно шаблон не является законченным образцом, который может быть прямо преобразован в код; это лишь пример решения задачи, который можно использовать в различных ситуациях.

Основная польза от использования паттернов состоит в снижении сложности разработки за счет готовых абстракций для решения целого класса проблем. За счет паттернов производится унификация деталей решений: модулей, элементов проекта; снижается количество ошибок. Применение паттернов концептуально сродни использованию готовых библиотек кода. Правильно сформулированный шаблон проектирования позволяет, отыскав удачное решение, пользоваться им снова и снова.

Паттерны могут быть реализованы различными способами: как конвенции, с использованием генераторов кода, в виде новых конструкций языка, в виде библиотек [1].

В настоящий момент существует много материалов, посвященных паттернам параллельного программирования. В одной из самых известных книг [2] рассматривается процесс создания параллельного решения на основе паттернов и паттерны, представленные на каждой стадии создания решения. Схема создания решения включает стадии: поиск параллельности, выбор структуры алгоритма, поддерживающие структуры и механизмы реализации. Каждая из вышеизложенных стадий содержит ряд паттернов. То, что предлагается в [2], относится к реализации типа «конвенция».

Мы рассматриваем паттерны параллельного программирования как коллекцию полезных техник параллельного программирования, которая может быть организована в виде иерархии классов или C++-шаблонов. Как правило, такие паттерны опираются на некоторую библиотеку, позволяющую организовать параллельный счет. При этом вся сложность использования библиотеки скрывается за простым и гибким интерфейсом. В общем виде паттерн – это заготовка, пригодная для организации логики параллельной программы. Для того чтобы воспользоваться паттерном, программисту необходимо добавить конкретные данные в абстрактные конструкции.

В настоящее время существует довольно много кроссплатформенных многопоточных библиотек для языка программирования C++, например, Boost Thread, C++ Standard Library Thread, Parallel Patterns Library, POSIX Threads, Threading Building Blocks (TBB) и т. д. Эти программные средства предназначены для написания параллельных программ на языке программирования C++. Они относятся к классу программных средств «библиотека» или «фрэймворк». Некоторые из библиотек включают реализации различных паттернов. Например, Parallel Patterns Library от Microsoft содержит паттерны `for`, `for_each`, `map` и другие, а Threading Building Blocks от Intel – `for`, `reduce`, `do`, `scan`, `while`, `pipeline`, `sort`.

ПРОГРАММНЫЙ ИНСТРУМЕНТ

Предлагаемый программный инструмент может быть использован для проектирования параллельных приложений с общей памятью. Рассмотрим основные этапы цикла разработки параллельной программы с использованием данного инструмента.

Выбор и настройка паттерна программы с помощью Мастера. Реализованы общие паттерны параллельных программ на базе следующих моделей создания и функционирования потоков: модель делегирования, модель с равноправными узлами, конвейер, модель «изготовитель-потребитель». Также реализованы специализированные паттерны параллельных программ, которые являются конкретизацией общего паттерна для решения определенного множества задач. Пользователь выбирает подходящий паттерн и уточняет его параметры в зависимости от решаемой задачи и цели проектирования. В результате выполнения этого этапа строится каркас приложения.

Ручная конкретизация паттерна. В случае необходимости полученный каркас приложения вручную дорабатывается пользователем.

Получение экспериментальных характеристик. Здесь предполагается проведение экспериментов с полученной параллельной программой, например в целях определения ее эффективности.

ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ

Программный инструмент предназначен для проектирования параллельных программ с общей памятью на языке C/C++. При проектировании программ используются готовые паттерны, что делает процесс написания программ более простым и надежным, избавляет программиста от рутинной работы.

Результаты, полученные при апробации предложенного инструментария, иллюстрируют снижение трудоемкости разработки, а также подтверждают, что избранная методика позволяет ускорить и улучшить процесс проектирования параллельного кода, при этом несущественно влияет на эффективность исполнения созданных программ.

Разработанный инструмент может быть использован для решения следующих задач:

- проектирование параллельных программ для многопроцессорных систем;
- оценка важнейших параметров исполнения программ в параллельном режиме;
- обучение технологиям параллельного программирования.

БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

1. McCool M. D. Structured Parallel Programming with Deterministic Patterns // 2nd USENIX Workshop on Hot Topics in Parallelism, 14–15 June 2010, Berkeley, CA.
2. Mattson T. G., Sanders B. A., Massingill B. L. Patterns for Parallel Programming // Addison Wesley, 2002.