



ПРИМЕНЕНИЕ ПОДХОДОВ НЕПРЕРЫВНОЙ ИНТЕГРАЦИИ И РАЗВЕРТЫВАНИЯ В АКАДЕМИЧЕСКИХ СИСТЕМАХ УПРАВЛЕНИЯ ЗАДАЧАМИ

(Самарский университет)

Разработка любого программного обеспечения (ПО), используемого широкой аудиторией – это очень трудоёмкий процесс, включающий в себя помимо собственно разработки два не менее сложных и настолько же важных этапа: тестирование и поставку ПО конечному пользователю. Одним из решений данных проблем является автоматизация процессов с помощью внедрения подходов непрерывной интеграции и развертывания (англ. CI / CD), что на сегодняшний день де-факто стало стандартом и применяется повсеместно [1]. Суть этих подходов заключается в уменьшении времени введения в эксплуатацию любых изменений, вносимых в продукт. На рисунке 1 схематично представлены зоны ответственности перечисленных выше принципов.



Рисунок 1 – Зоны ответственности принципов CI / CD

Данные принципы применимы и для отдельных частей программного обеспечения. В работе [2] представлена система, основанная на принципе добровольных вычислений и обеспечивающая доступ к суперкомпьютерному кластеру Самарского университета. Изначально эксплуатация этой системы заключалась в запуске и мониторинге задач, позднее в неё была внедрена облачная среда разработки, предоставляющая единую точку входа с целью создания и запуска программ для высокопроизводительных научных вычислений [3].

Одна из концепций, используемых для упрощения работы с системой – использование так называемых скелетных программ: predefined шаблонов, реализующих некоторый алгоритм [4]. Однако в отличие от типичного



представления шаблонов, скелетные программы могут быть скомпилированы и запущены. В рамках вышеуказанной системы скелетоны реализованы на языке Templet, описанном в работе [5]. Главный мотив – позволить пользователям, не имеющим опыта, применять параллелизм при решении их задач.

Данные скелетоны могут как подвергаться некоторым модификациям с целью оптимизации или исправления ошибок, так и использоваться для создания новых версий. И так как данные шаблоны являются своего рода фундаментом при создании параллельных программ для высокопроизводительных вычислений, критически важно гарантировать их корректность. Эту задачу может помочь решить внедрение CI / CD системы. Далее приведены основные задачи, которые могут быть возложены на такую систему:

1. модульное тестирование – успешная компиляция и запуск в качестве самого простого критерия отдельно взятого шаблона;
2. функциональное тестирование – ход работы простейшей программы, основанной на данном шаблоне, соответствует алгоритму, заложенному в шаблоне;
3. развертывание – сборка пакета шаблонов и доставка на целевое окружение;
4. информирование – обновление статуса шаблона в основной системе.

На рисунке 2 представлен основной алгоритм работы системы непрерывной интеграции и развёртывания.

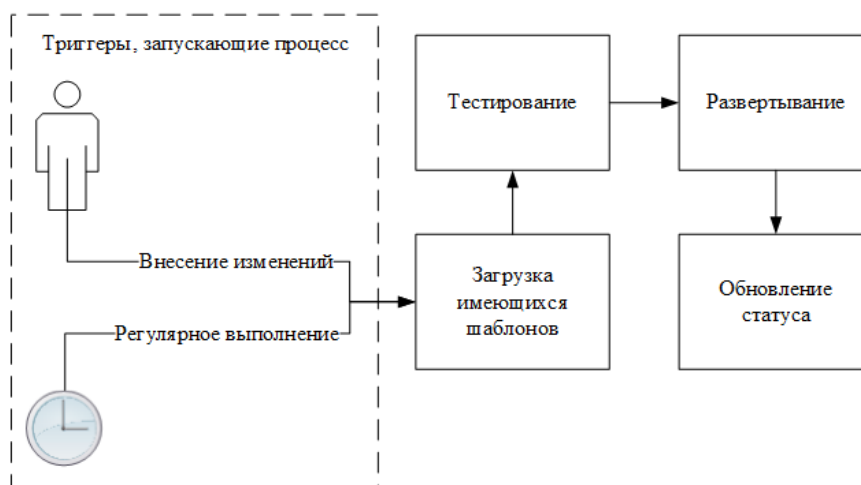


Рисунок 2 – Алгоритм работы CI / CD системы

Таким образом, использование подходов и систем непрерывной интеграции и развёртывания позволяет снизить трудоёмкость процесса разработки скелетонов, повысить качество и гарантировать стабильность работы параллельных программ, созданных на их основе. Показано, каким образом данная технология может быть внедрена в академическую систему управлением задач.

Литература

1. Gusev, V Use of Continuous Integration in BOINC Projects Development [Текст] /V. Gusev //BOINC:FAST – 2017.



2. Артамонов, Ю.С. Востокин, С.В. Инструментальное программное обеспечение для разработки и поддержки исполнения приложений научных вычислений в кластерных системах [Текст] /Ю.С. Артамонов, С.В. Востокин //Вестн. Сам. Гос. техн. ун-та. Сер. Физ.-мат. науки – 2015 – С.785-798.

3. Царёв, Д.А. Артамонов, Ю.С. Сравнение основных возможностей и классификация облачных инструментов разработки [Текст] /Д.А. Царёв, Ю.С. Артамонов //Перспективные информационные технологии (ПИТ 2016): труды Международной научно-технической конференции /под ред. С.А. Прохорова – Самара: Издательство СНЦ РАН – 2016 – С.539-542.

4. Царёв, Д.А. Востокин, С.В. Технология развёртывания скелетных программ для автоматизации вычислений на суперкомпьютере «Сергей Королёв» [Текст] /Д.А. Царёв, С.В. Востокин //Перспективные информационные технологии (ПИТ 2017): труды Международной научно-технической конференции /под ред. С.А. Прохорова – Самара: Издательство СНЦ РАН – 2017.

5. Vostokin, S.V. Templet: A markup language for concurrent actor oriented programming [Текст] /S.V. Vostokin //CEUR Workshop Proceedings – 2016 – С.460-468.

Х.М. Шерматова

ПАРАЛЛЕЛЬНОЕ МИКРОПРОГРАММИРОВАНИЕ АЛГОРИТМА ВЫЧИСЛЕНИЯ СТАНДАРТНЫХ ФУНКЦИЙ

(Ферганский государственный университет)

Аннотация. В данной статье рассматриваются вопросы создания параллельного микропрограммирования для алгоритма вычисления стандартных функций и на их основе – параллельной работы микропрограммного автомата с памятью произвольной выборки.

В результате исследований основанный на средства параллельного представления алгоритмов и способов их микропрограммной интерпретации называется параллельным микропрограммированием.

Распараллеливание процесса обработки информации обнаруживается на всех уровнях организации вычислительного процесса, причем прогресс технологии способствует проникновению параллельности на самые низкие уровни, вплоть до элементарных операций (предельное распараллеливание). Это проявляется в основном в применении устройств массовой обработки информации (матричные арифметические устройства, ассоциативные процессоры). Такие устройства предназначены для реализации узкого класса алгоритмов, которые естественным образом распараллеливаются до микроопераций.

Устройства, в котором возможна реализация параллельной микропрограммы, можно себе представить в виде сети элементарных автоматов, совокупная внутренняя память которых образует общую память данных. Логика