



6. IPC/JEDEC-9702, MONOTONIC BEND CHARACTERIZATION OF BOARD-LEVEL INTERCONNECT, Developed by the JEDEC Reliability Test Methods for Packaged Devices Committee, 2004

И.Н. Курганский, С.В. Востокин

ПРОЕКТ ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ УДАЛЕННОГО ВЫПОЛНЕНИЯ ПОТОКА РАБОТ НА ТЕРМИНАЛАХ ANDROID

(Самарский университет)

На практике часто возникают задачи, требующие организации координированной работы большого числа исполнителей. Например, может потребоваться произвести разметку изображений для обучения систем распознавания, выполнить ручное фотографирование заданных объектов на местности, расклейку объявлений в заданных местах, распределённые вычисления и т. п. Перечисленные виды деятельности представляют собой пример бизнес-процессов, моделируемых в виде потока работ. В тоже время в современном мире почти у каждого человека есть смартфон или планшет. По данным сайта gartner.com [1] большая часть смартфонов управляется операционной системой Android. Это даёт техническую возможность направить работу из управляющей компьютерной системы непосредственно на персональное Android-устройство исполнителя и отслеживать её выполнение. При этом управляющая система при формировании работ и контроле их выполнения может использовать персонализацию исполнителя, его геолокацию, а также данные геопривязки объектов на местности, видео- фото- аудио- данные, консольный ввод-вывод, контроль времени и другие возможности мобильного устройства. В связи с этим актуальна идея реализации программы для выполнения потоков работ, которая бы использовала удаленные терминалы на платформе Android. В работе были проанализированы технические требования к данной системе и подходы её реализации.

Разрабатываемая система будет иметь распределенную клиент-серверную архитектуру. К серверу системы предъявляются следующие требования: (а) авторизация по логину-паролю или токену; (б) добавление и изменение существующей задачи; (в) предоставление возможности проверки результата задачи; (г) возможность добавления задач типа «сделать фотографию», «написать текстовое сообщение», «предоставить вычислительную мощность»; (д) данные системы должны храниться в базе данных SQL; (е) взаимодействие с системой должно осуществляться через REST API.

К клиенту предъявляются следующие требования: (а) авторизация на сервере при помощи логина-пароля; (б) возможность приложить снимки к задаче; (в) возможность запуска кода, полученного с сервера.

Для разработки приложения для ОС Android можно расширить существующее приложение или создать новое, которое делегирует все запросы, связанные с распределёнными вычислениями, официальному приложению BOINC



из Google Play. Создать сервер возможно несколькими способами: (а) использовать фреймворк Spring, чтобы не разрабатывать сервер с нуля [2]; (б) расширить готовый BOINC [3] сервер под свои задачи. Рассмотрим каждый пункт отдельно, отметив его плюсы и минусы, после чего выберем наиболее оптимальный вариант.

Фреймворк Spring написан на языке программирования Java и содержит в себе инструменты для быстрого развертывания сервера с авторизацией и созданием REST API сервисов. Однако для того, чтобы создавать распределенные задачи, необходимо чтобы код, который будет выполняться на клиентах, был скомпилирован в готовую программу. Сложность заключается в том, что клиент должен работать под управлением операционной системы Android. По умолчанию язык разработки для приложения на Android – Java. Политика безопасности данного языка такова, что запускать исходный код из внешних источников без предварительной компиляции этого кода на клиенте невозможно. Вторым способом создания приложения является использование NDK и языка C++. В данном подходе Android также позволяет запускать программы, полученные из внешних источников, с ограниченными правами.

Платформа BOINC — это открытый программный комплекс, разработанный университетом Беркли для организации распределенных вычислений. Серверная часть состоит из HTTP-сервера с веб-сайтом проекта, базы данных MySQL и набора служб (генератор заданий, планировщик, валидатор, ассимилятор результатов). Основным преимуществом данной системы является реализация клиентов под большинство операционных систем, таких как Windows, Linux, MacOS и Android. Для реализации собственного BOINC-проекта достаточно расширить базовый функционал комплекса.

Таким образом, наиболее подходящим с точки зрения простоты реализации и удовлетворения заявленным техническим требованиям является использование Spring-сервера с агрегированием BOINC платформы.

Литература

1. Gartner Says Worldwide Sales of Smartphones Grew 9 Percent in First Quarter of 2017 [Электронный ресурс] - Режим доступа: <https://www.gartner.com/newsroom/id/3725117> [Дата обращения 24.01.2018].
2. Spring: the source for modern java [Электронный ресурс] - Режим доступа: <https://www.spring.io> [Дата обращения 24.01.2018].
3. BOINC: Open-source software for volunteer computing [Электронный ресурс] - Режим доступа: <https://boinc.berkeley.edu> [Дата обращения 22.02.2018].