A Study on the WIPI-to-iOS Mobile Game Converter

YangSun Lee*, YunSik Son**

*Dept. of Computer Engineering, Seokyeong University
16-1 Jungneung-Dong, Sungbuk-Ku, Seoul 136-704, KOREA
yslee@skuniv.ac.kr

**Dept. of Computer Engineering, Dongguk University
26 3-Ga Phil-Dong, Jung-Gu, Seoul 100-715, KOREA
Corresponding Author : sonbug@dongguk.edu

Abstract. In this paper, the WIPI-to-iOS automatic mobile game converter was designed to automatically translate mobile game contents from the WIPI platform to the iOS platform for smartphones. Through the WIPI-to-iOS converter, resources such as images and sounds can be converted, APIs can be converted using a platform mapping engine with wrapper functions. These and all other content conversion functions were examined. Test results indicate that the graphics, image output, sound output, and other functions of converted iOS platform games were equivalent to those of the WIPI platform games before conversion.

Keywords: WIPI-to-iOS Mobile Game Converter, WIPI(Wireless Internet Platform for Interoperability), iOS, Platform Mapping Engine, Source Translator

1 Introduction

In the mobile market, the appearance of various smart phone platforms such as Android, iOS and Window Phone has led to game developers to repeatedly develop game contents to suit the different mobile communication companies’ platforms in order to service mobile game contents. Furthermore, to use each of the game contents developed on the existing feature phone platform, they need to be recreated based on the smart phone platform. [1-3].

This research aims to solve these problems by inventing a resource translator for the WIPI-to-iOS mobile game converter which automatically converts game contents from the existing feature phone platform WIPI to the smart phone platform iOS. By automatically converting the existing mobile game contents used in the WIPI feature phone platform to game contents for use in the iOS platform, existing game contents can be ported into a different platform within a short period of time.

*This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No.20100023644)*

ISSN: 2287-1233 ASTL
Copyright © 2014 SERSC
2 Relative Studies

2.1 WIPI

WIPI is legislated by KWISF (Korea Wireless Internet Standardization Forum) and a standardized standard chosen by KTTA (Korea Telecommunications Technology Association) as an application program execution environment for mobile communication platforms.

2.2 iOS

The iOS platform developed by Apple is the world’s most advanced mobile operating system, continually redefining what people can do with a mobile device. Together, the iOS SDK and Xcode IDE make it easy for developers to create. Derived from core OS X technologies, the amazing user experience of iOS has been streamlined to take maximum advantage of iPhone, iPad, and iPod touch hardware. Technologies shared between iOS and OS X includes the OS X kernel, BSD sockets for networking, and Objective-C and C/C++ compilers for native performance.

3 The WIPI-to-iOS Mobile Game Converter

3.1 The WIPI-to-iOS Converter System

The WIPI-to-iOS automatic mobile game converter receives WIPI game content in source form and converts it into the source form that is run on the iOS platform. Figure 3 shows a model of the WIPI-to-iOS mobile game content automatic converter system.

![WIPI-to-iOS Mobile Game Converter Model](image-url)
3.2 Content Analyzer

The content analyzer [1-6] is a system that analyzes the WIPI game content that is input, and produces an output in which the resource data and source code stored within the content are separated. Before conversion, the source code and resource data must be separated and converted individually.

3.3 Resource Converter

The resource converter [1-6] is a system that converts the resource data, which is in text or binary form, into image data, sound data, and user data for use in the target platform's file system. The image file formats used in each platform (e.g., BMP, PNG, JPEG), sound formats (e.g., WAV, MP3, MMF), and user data must be researched and converted for use in the target platform.

3.4 Platform Mapping Engine

The platform mapping engine [1-6] creates an execution environment on the iOS platform that is identical to that in WIPI platform so that the same environments can be executed identically. Thus, a WIPI game is enabled to run in its original form on the iOS platform. On the basis of the created execution environment, a wrapper function format is provided that enables identical execution of WIPI C's API on the iOS platform.

3.5 Source Translator

The source translator [1-6] receives the WIPI C source code output by the content analyzer and translates it into iOS C source code that is semantically equivalent and executes the same actions as the WIPI C source code. Because WIPI C and iOS are both C-based platforms, the characteristics of the languages are the same. However, some parts have been altered to suit each platform's virtual machine.

4 Results

In this paper, the WIPI-to-iOS mobile game converter was used to automatically convert mobile game content from the feature phone WIPI platform to the smartphone Android platform. The results of the conversion were then compared. Test results indicate that the graphics, image output, sound output, and other functions of converted iOS platform games were equivalent to those of the WIPI platform games before conversion. Figure 1 shows the execution results of an "Aiolos" game.
5 Conclusions and Further Researches

The automatic mobile game converter WIPI-to-iOS presented in this paper offers a means to solve the problems of different mobile platforms. It can ensure quick and automatic conversion of existing WIPI game content into game content for the iOS smart phone platform, thus increasing the reusability of existing content and providing smart phone users with more diverse content.

References

1. WIPI (Wireless Internet Platform for Interoperability), KWISF(Korea Wireless Internet Standardization Forum), 2004.