

BAUHINIA CORONATA AND NITIDA: WII REMOTE TECHNOLOGIES FOR A SUSTAINABLE WORLD

BAUHINIA CORONATA E NITIDA: TECNOLOGIAS REMOTAS WII PARA UM MUNDO SUSTENTÁVEL

Data de entrega dos originais à
redação em: 23/02/2016
e recebido para diagramação em:
30/11/2016

André Willik Valenti¹

Coronata and nitida are open-source software projects that enable using a Wii Remote on a computer for tasks like remote controlling a slides presentation. The Wii Remote, also known as Wiimote, is the main input device for the Nintendo Wii videogame console. It can be connected to a computer via Bluetooth and controlled by software. coronata is a Java library that makes the development of Wiimote applications easy. nitida is an application built on top of coronata that transforms Wiimote into a remote presenter, game controller or similar; this is accomplished by generating keyboard events. Similar applications and libraries exist; however, nitida and coronata are much simpler to use and are cross-platform, with easy deployment both on Windows and Linux environments. The technologies presented in this paper enable people to harness electronic devices that they already have to perform daily tasks in a user-friendly and environment-friendly way.

Keywords: Sustainability. Remote Presenter. HCI. Wii. Wii Remote. Wiimote. Bluetooth.

Coronata e nitida são projetos de software de código aberto que permitem usar um Wii Remote num computador para tarefas como o controle remoto de uma apresentação de slides. O Wii Remote, também conhecido como Wiimote, é o principal dispositivo de entrada para o console de videogame Nintendo Wii. Ele pode ser conectado a um computador via Bluetooth e controlado por software. Coronata é uma biblioteca Java que torna o desenvolvimento de aplicações Wiimote fácil. Nitida é um aplicativo construído em cima de coronata que transforma Wiimote em um apresentador remoto, controlador de jogo ou similar; Isso é conseguido gerando eventos de teclado. Existem aplicações e bibliotecas semelhantes; No entanto, nitida e coronata são muito mais simples de usar e são multi-plataforma, com fácil implantação, tanto em ambientes Windows e Linux. As tecnologias apresentadas neste documento permitem às pessoas aproveitar os dispositivos eletrônicos que eles já têm para executar tarefas diárias de uma maneira amigável e amigável ao meio ambiente.

Palavras-chave: Sustentabilidade. Apresentador Remoto. HCI. Wii. Wii Remote. Wiimote. Bluetooth

¹Professor at IFSP Campus Campinas, <awvalenti@gmail.com>

1 INTRODUCTION

Sustainability is a major issue of today's world. In the technology field, many countries have to deal with the problem of electronic waste. In 2013, e-waste production reached one million tons in Brazil, according to Grossman (2013). Considering the whole world, figures are even greater:

The electronic waste problem is huge: More than 20 million tons of e-waste are produced every year. Americans alone generate about 3.4 million tons of e-waste per year. If you put every blue whale alive today on one side of a scale and one year of US e-waste on the other, the e-waste would be heavier. (IFIXIT.ORG, 2015)

An environment-friendly attitude should be part of our daily lives. Concerning waste management, this is summarized by the principle called The Three R's: Reduce, Reuse, Recycle. Reduction is the preferred attitude: it completely avoids waste production by eliminating unnecessary purchases. Reusing is also a very good alternative. It means giving new purposes for materials that one already has. Recycling is the least preferred option because it involves an industrial process that still generates some waste, although it is much better than direct discard. The Three R's principle is embraced by many organizations worldwide, including the University of São Paulo (2015) and the United States Environmental Protection Agency (2015). This scenario encourages research and development of solutions to daily problems utilizing consumer goods that one already has, which is a form of reducing purchases and reusing items. In this paper, we present an example of how this mindset can be applied.

Coronata (/ko-ro-'nah-tah/) and nitida ('nee-ti-dah/) are two open-source projects hosted on Bauhinia (/ba-'wii-nee/), a public GitHub repository. coronata is a Wii Remote Java library that simplifies development of cross-platform applications. nitida is a remote controller application that enables using a Wii Remote for tasks like controlling a slides presentation, playing a game or controlling a media player. nitida aims to be: 1) as easy to install as possible; 2) usable by non-technical people; 3) portable across different operating systems; 4) capable of running as ordinary user (i.e., without administrative privileges).

Nintendo Wii console has been released in 2006. It is a very successful product, with 101.56 million units sold worldwide as of June 30, 2015 (NINTENDO, 2015). The main means of interaction between a player and a Wii console is its default controller, the Wii Remote. This controller, also called Wiimote, is a wireless, Bluetooth device with many capabilities, including buttons, LEDs, speaker, vibration motor, accelerometer and infrared tracking camera. It may be connected to a personal computer for a wide range of uses (LEE, 2008).

A remote presenter is usually a dedicated wireless device that connects to a computer and enables its user to control a slides presentation from distance. This avoids the need of staying close to the computer to advance slides, offering mobility for the person who is presenting. Many commercial remote presenters are available on the market (TOP TEN REVIEWS, 2015). It is quite convenient to use such a device in lectures. However, buying a dedicated, commercial product is not the only choice. It is possible to achieve the same effect by utilizing already owned hardware (such as a Wii Remote), combined with proper software.

2 RELATED WORK

Connecting a Wiimote to a personal computer is not something new. Many software projects with that goal exist. Johnny Chung Lee is known for many of these projects, including a low-cost interactive whiteboard (LEE, 2007). In his paper “Hacking the Nintendo Wii Remote” (LEE, 2008), he briefly explains some technical aspects of the Wii Remote and points to references with more detailed information. He also states that hacking a Wiimote is relatively simple:

In many cases, the most difficult part of this process is getting the Bluetooth pairing to occur successfully. Because the Wii remote isn't 100 percent HID compliant, it might work only with certain Bluetooth chipsets and driver software. However, once a pairing is successful, the configuration is typically quite reliable. (LEE, 2008).

For the use case of remote controlling a presentation, the only feature of Wiimote that needs to be handled are the buttons. In addition, keyboard events must be generated. The next subsections describe existing tools that have these capabilities.

2.1 GlovePIE

GlovePIE (KENNER, 2010) is one of the first and most widely known tools. It is a Windows application that allows programming scripts to handle Wiimote events, such as pressing buttons and tilting the controller. It is capable of generating mouse and keyboard events, as well as many other features. It may be used as a remote presenter if properly configured.

GlovePIE is targeted mostly at the hacking community. Although powerful, it has some disadvantages. It has a non-intuitive interface and requires some configuration to work, which means that out-of-box use is not possible. Some technical skills are necessary to configure it. In addition, GlovePIE requires administrative privileges to run, which limits its application on public machines. Finally, it works only on Windows.

2.2 Wiimote Presenter

Wiimote Presenter (Jason SMITH; KITZMANN, 2009) is a Windows application whose main goal is to serve as a remote presenter. It allows mapping Wiimote buttons to key/mouse events, like GlovePIE. However, it is targeted at a more general audience and has an intuitive interface.

Out-of-box use is possible, since many buttons-to-keys mappings are provided and a default one is automatically loaded. Customization can be easily done by changing the values of drop-down lists and these custom mappings can be saved for future use. It also offers some advanced features, like controlling the mouse pointer using the infrared camera of Wiimote and setting timers to help the user control his or her presentation time. The main disadvantage of Wiimote Presenter is that it works only on Windows.

2.3 CWiiD

On Linux environments, CWiiD (pronounced “seaweed”) (Donnie SMITH, 2010) is a popular collection of programs for using Wiimote on a computer. It is available as an official Ubuntu package and may be installed on Ubuntu (or derivatives) by running the command “sudo apt-get install libcwiiid1 lswm wmgui wminput”. wminput is the program that connects to a Wiimote, reads its events and generates mouse and/or keyboard events accordingly.

Like Wiimote Presenter, it comes with predefined mappings and loads them from configuration files. To install CWiiD, one needs administrative privileges. This is quite common among Linux packages. However, to configure and run it, one also needs administrative privileges (CONTRIBUTORS, 2015). CWiiD is targeted at experienced users. Many configuration steps are necessary and they must be done by a privileged user. By default, even the wminput program requires sudo. It is not possible to use CWiiD out-of-box. In addition, it works only on Linux.

3 THE BAUHINIA PROJECTS

Bauhinia is a public GitHub repository that hosts the projects described in more detail below.

3.1 Coronata

Coronata is a Java library for Wiimote programming. Although primarily designed to support the nitida application, it is intended to be used separately, for any Wiimote application. Its main advantages are that it is cross-platform and is easy to work with, thanks to its very clean API. The implementation is based on the following libraries: WiiuseJ/wiiuse (DUCHÉ, 2008; LAFOREST, 2008) and BlueCove (TOTTERMAN et al., 2008). wiiuse is a native C library for handling Wiimotes, while WiiuseJ is its Java port. They are used on the Windows platform. BlueCove is a general Bluetooth library and is used on the Linux platform. coronata automatically detects the operating system and selects the appropriate option accordingly.

The main motivation for developing a new Wiimote library is that the existing ones were not completely portable. BlueCove works well on Linux, but is not fully compatible with Windows when the default Bluetooth drivers from Microsoft are used; therefore, other drivers would have to be installed – and they are not compatible with all Bluetooth adapters. WiiuseJ and wiiuse work well on Windows, but require writing specific files to a system directory (/usr/lib) to work on Linux. These issues would require extra steps of administrative operations, which would complicate configuration and limit access only to technical users.

By combining the best features of these libraries, coronata has less system requirements. On Linux, the user has to install, only once, a system package. On Ubuntu or derivatives, this can be accomplished by running “sudo apt-get install libbluetooth-dev”. This limitation is inherited from the BlueCove library. Compared to CWiiD, it is a very easy configuration procedure and is the only operation that needs administrative privileges. After that, applications can be executed by ordinary users. On Windows, coronata needs write access to the user's temp directory – a very common and simple requirement. It also requires pairing the Wii Remote to Windows every time

the controller is turned on. This manual process is somewhat tedious, although Wiimote Presenter and GlovePIE exhibit the same limitation. Among all the Wiimote features, coronata currently supports listening to button pressing, turning LEDs on and vibrating the controller.

3.2 Nitida

Nitida is a portable remote controller application, built on top of coronata. It consists of a single, executable JAR file, which can be run as an ordinary user. The only configurations necessary to use nitida are the ones required by coronata: on Windows, Bluetooth pairing; on Linux, installing a system package. Once those requirements are met, running nitida automatically starts a search for a Wiimote, connects to it (if one is found) and activates the keyboard input generator, which uses the `java.awt.Robot` class from the Java standard library. The remote presenter key mapping is loaded by default, which allows the user to control a remote presentation.

Nitida aims to be usable by non-technical people. For that reason, it focuses on the simplest and used features, which are loaded by default, not requiring any user intervention (i.e., it works out-of-box). For old computers, nitida has the advantage of very low CPU time consumption. On a computer running Windows XP on a 2007 processor (AMD Sempron LE-1100 1.9GHz), nitida's CPU usage stayed on 0% most of the time, with a maximum of 4%. By contrast, on this same machine, Wiimote Presenter consumed between 40% and 60% of CPU time.

4 CONCLUSIONS AND FUTURE WORK

This paper presented coronata and nitida, two technologies that allow utilization of Wii Remote for simple goals, like remote controlling a presentation or playing a game. These technologies enable leveraging tools that a person already has to carry out daily tasks. Motivation for this work came from the author's own desire to remote control his presentations. Developing solutions using consumer goods that one already has is a good alternative to buying new products, both for environmental and economic reasons.

As for future work, there are several possible improvements for the Bauhinia projects. The user interface of nitida could observe enhancements in usability. Also, some interesting new features to have are warnings about low-battery, long inactivity period and remaining presentation time; this last feature is available on Wiimote Presenter.

Improvements can be made on coronata, too. One problem observed on Windows is that a disconnection of Bluetooth adapter or Wii Remote is not being detected and handled properly. When this occurs, the library should inform the program using it and recover to a consistent state. More testing is also necessary. coronata and nitida have not yet been tested neither on the Mac OS platform nor with the newer version of Wii Remote, identified as "Nintendo RVL-CNT-01-TR". Finally, coronata could be extended to support more Wiimote features, such as tracking infrared light, reading accelerometer values and playing sounds through the speaker.

Coronata and nitida are open-source projects. The reader is encouraged to download the source code, study it and participate in the development by opening pull requests. The repository is located at <https://github.com/awvalenti/bauhinia>.

REFERENCES

- CONTRIBUTORS, Ubuntu Documentation. **CWiiD**: Community Help Wiki. 2015. Available at: <<https://help.ubuntu.com/community/CWiiD>>. Accessed on: 2015-09-18.
- DUCHÉ, Guilhem. **Wiiusej**: Java API for Wiimotes. 2008. Available at: <<https://code.google.com/p/wiiusej/>>. Accessed on: 2015-09-16.
- GROSSMANN, Luís Osvaldo. **Brasil já produz 1 milhão de toneladas de lixo eletrônico por ano**. 2013. Available at: <<http://convergenciadigital.uol.com.br/cgi/cgilua.exe/sys/start.htm?infoid=34032>>. Accessed on: 2015-09-18.
- IFIXIT.ORG. **The Problem with E-Waste**. 2015. Available at: <<http://ifixit.org/ewaste>>. Accessed on: 2015-09-18.
- KENNER, Carl. **GlovePIE**: Programmable Input Emulator. 2010. Available at: <<http://glovepie.org/>>. Accessed on: 2015-09-18.
- LAFOREST, Michael. **Wiiuse**: The Wiimote C Library. 2008. Available at: <<http://www.macs.hw.ac.uk/~ruth/year4VEs/Labs/wiiuse.html>>. Accessed on: 2015-09-16.
- LEE, Johnny Chung. **Low-Cost Multi-touch Whiteboard using the Wiimote**. 2007. Available at: <<https://www.youtube.com/watch?v=5s5EvhHy7eQ>>. Accessed on: 2015-09-18.
- LEE, Johnny Chung (Ed.). **Hacking the Nintendo Wii Remote**. 2008. Available at: <<http://www.cs.cmu.edu/~15-821/CDROM/PAPERS/lee2008.pdf>>. Accessed on: 2015-09-09.
- NINTENDO. **IR Information: Sales Data - Hardware and Software Sales Units**. 2015. Available at: <http://www.nintendo.co.jp/ir/en/sales/hard_soft/>. Accessed on: 2015-09-18.
- SMITH, Donnie. **CWiid Wiimote Interface**: Linux Nintendo Wiimote interface. 2010. Available at: <<https://github.com/abstrakraft/cwiid>>. Accessed on: 2015-09-18.
- SMITH, Jason; KITZMANN, Ingo. **Wiimote Presenter**. 2009. Available at: <<https://sites.google.com/site/jasonlpsmith/wiimotepresenter>>. Accessed on: 2015-09-18.
- TOP TEN REVIEWS. **Wireless Presenters Review**. Available at: <<http://wireless-presenters-review.toptenreviews.com/>>. Accessed on: 2015-09-18.
- TOTTERMAN, Paul et al. **BlueCove**: BlueCove JSR-82 project. 2008. Available at: <<http://bluecove.org/>>. Accessed on: 2015-09-18.
- UNITED STATES ENVIRONMENTAL PROTECTION AGENCY. **Reduce, Reuse, Recycle**. 2015. Available at: <<http://www2.epa.gov/recycle>>. Accessed on: 2015-09-19.

UNIVERSITY OF SÃO PAULO. **Dicas de Atitudes Sustentáveis - Redução, Reutilização e Reciclagem.** 2015. Available at:
<http://www.ifsc.usp.br/~ambiental/index.php?option=com_content&view=article&id=24:dicas-de-atitudes-sustentaveis-reducao-reutilizacao-e-reciclagem&catid=2:uncategorised>. Accessed on: 2015-09-19.