

OWL Human Interface Package (HIP™) PJ & J2

OWL leverages SIPTECH's J2SDK competence for development of OWL Human Interface Package

A SIPTECH CASE STUDY

Version: 1.03
Created Date: 13.Dec.2004

Inside this document ...

01. Background
02. Problem / Challenges
03. Method / Intervention / Solution
04. Results Achieved
05. Lessons Learned / Conclusion



© 2005 All rights reserved. SIP Technologies and Exports Ltd. The information contained in this document is CONFIDENTIAL and PROPRIETARY in nature, and subject to the rights and ownership of SIPTECH. Any and all unauthorized copying or use of the contents hereof is prohibited.

1. Background

Our World Live, Inc. (OWL) is an emerging-technology company based in Orange County, CA, which is developing a middleware platform for delivery of digital assets/media. OWL was evaluating an engineering partner who could build the product rapidly and economically. With its track record of providing quality offshore product engineering services, SIPTECH was a logical choice for OWL.

2. Problem / Challenge

In the first phase of the project, OWL wanted to develop a Human Interface Package (HIP™) that significantly extended the capabilities of the GUI widgets that are provided by the Java™ platform. The enhanced functionality needed to behave the same on any device ranging from a browser running on a PC desktop to an application running on a Java-powered handheld device.

In addition to the tight financial constraints of the project, OWL had an aggressive deadline to meet - the HIP™ had to be completed before March 2002 to enable it to be showcased in JavaOne 2002. This gave SIPTECH a window of less than 3 months to design, develop, test and deliver the package.

3. Method / Intervention / Solution used

SIPTECH delivered the OWL HIP™ project using an offshore project execution model within the specified time frame. The OWL UI elements extend the capabilities of the AWT/Swing™ components to support free form shapes, dynamic transparency (24-bit transparency mask with numeric transparency property) and draggability. The requirements were for support of the above features on (1) AWT on JDK1.1.8 or Personal Java (Both were equivalent) and (2) Swing UI framework available at that time on the latest JDK. The first was code named as OWL HIP™ PJ and the second OWL HIP™ J2.

Additional APIs as well as component classes are provided to construct user interfaces and their respective UI elements that are fully customizable in their looks and behaviour.

Additionally, the OWL HIP™ components are flicker-free and are fully compatible with AWT/Swing™ components. The greatest challenge while developing the OWL HIP™ was to ensure that all these features are easy to implement, render fast, show high performance and are user friendly.

4. Results Achieved

The result was the OWL HIP™ PJ (fully compatible with PersonalJava™ 1.2 specification and JDK 1.1.8 from Sun Microsystems) and OWL HIP™ J2 (fully compatible with Java™ 2 platform from Sun Microsystems) that extend the capabilities of AWT and Swing™ to render consistent graphical user interfaces across multiple environments.

Competitive Comparison		
Sun AWT	Sun Swing™	OWL HIP™
Standard regular shapes	Standard regular shapes	Fully customizable free form shapes
Text	Image Icons	24-bit dynamic transparency
Look and feel provided by OS	Transparency of GIF images	Direct support for images

5. Lessons Learned / Conclusion

The project dealt with the small-scale devices like Palmtop and Sharp Zaurus. It was important that the performance was not affected by the constraints in these devices with the level of rendering and masking that was supported by the UI components. Importance was given on testing the components on an actual device.

The constraints set by the client requirements helped in planning early and strategizing the work i.e. development and testing were prioritized and separate groups formed to complete the testing and development processes.