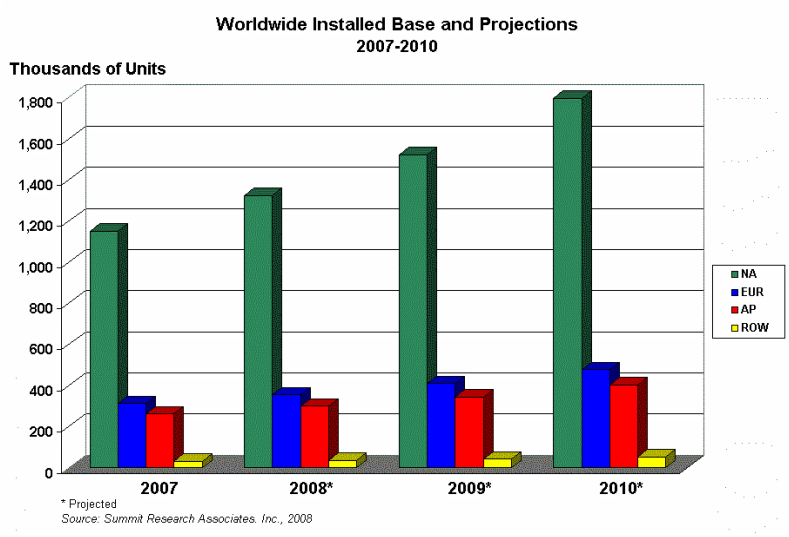
The Solution To Challenging Times:

The Integrated Approach

A Summit Research White Paper
October 2008

Summit Research Associates, Inc.
7728 Warbler Lane
Rockville, MD 20855-1034

Summit's recently published research, the Seventh Edition of their flagship report ***Kiosks and Interactive Technology,*** provides an optimistic picture for the continued growth and deployment of kiosks worldwide. Summit estimated the worldwide installed base will grow from 1.76 million units at the end of 2007 to more than 2.73 million by 2010. While all the major world regions will participate in this expansion, North America—primarily the United States—will see the greatest activity. Summit estimates the North American region will grow to nearly 1.8 million units by the end of 2010. The chart below illustrates these trends.



The world's current financial crisis, the worst in 70 years, has resulted in a major reevaluation and tightening of all types of economic activity. How will these events impact the kiosk industry? Summit recognizes some projects may be delayed or cancelled altogether. Funding for new projects and the expansion of existing initiatives may be more difficult to obtain.

However, Summit believes this financial crisis will likely have the opposite effect on kiosk deployment, and may actually cause a mini-boom to the kiosk industry. Because self-service kiosks replace personnel that are expensive to hire, train, manage, and compensate, they provide significant cost savings to the enterprise. And, there is little or no loss of functions performed.

In an exciting development, the new trend toward an **integrated approach** to kiosk development and deployment can significantly allow current and future projects to proceed expeditiously while attaining impressive cost savings.

With cost containment becoming the key factor in any discussion of kiosk development, deployment and maintenance, how does one achieve the goals of a kiosk deployment project without breaking the bank? Through an integrated kiosk platform. Summit believes the development of an integrated platform transforms the kiosk decision-making process and delivers the best of both worlds -- easy and rapid kiosk application development coupled with the best hardware to tackle the project and ensure its success.

A kiosk integration platform represents the future of this industry. Hardware and software will not have to be "cobbled together" with deployers crossing their fingers and hoping that the solution will work. The total product is designed at one time and uses quality-

tested components. A discussion of this integrated approach and its benefits form the substance of this White Paper.

Description

The goals of any kiosk project are to produce a solution that meets customer needs, is intuitive and easy to use, resides in hardware that is robust and reliable and is easily maintained – all within a reasonable budget. How best to achieve these goals?

There are three major components to an effective integration platform approach. Companies embracing this approach must get all three parts "right" in order for the integration platform to deliver its many benefits.

First, hardware must work flawlessly as a unit. This is not trivial. Companies that use this integrated approach are highly selective in the components they choose for their motherboards and the peripherals they will incorporate into their kiosk configurations. And, they subject all parts to rigorous quality assurance testing that can run for months. Once the components and parts have passed these quality assurance tests, the unit as a whole is subject to its own set of quality control tests to ensure the motherboard and peripherals all work together as an integrated unit.

Second, it is important to develop a middleware that receives much more comprehensive information about hardware states. Companies offering a highly integrated hardware unit should take advantage of the opportunity to understand in greater detail a more comprehensive set of the hardware states. This additional knowledge of hardware states includes the motherboard and all the peripherals. The middleware can pass this information on to the software platform.

Third, a kiosk software platform that takes advantage of the additional information provided by the middleware can handle all exception coding locally. This frees the application developer from having to do tedious low-level coding for all the peripherals. These are handled by the kiosk software platform. Kiosk application developers can focus on developing the most effective and creative application for the client. This includes using web-based tools for application development.

Why use the web for this type of activity? There are many reasons. Since most web programming languages are platform-independent, they provide native capabilities for desktop as well as Internet-based computing. Because these languages are so widely used, there is an abundant supply of experienced web software developers, both in the United States and in other countries with much lower salary levels. As a result, using these seasoned less expensive web developers for kiosk software design will save a substantial amount of money, resulting in a more cost-effective kiosk project.

Web developers can use their web skills to bring a project to completion much more quickly than those using kiosk-specific software. We are not implying that one can simply port a website interface over to a kiosk without modification. That would be unwise; many unsuccessful kiosk interfaces occur when deployers try to make a direct port and take a website and simply install it on a kiosk. However, this potential problem is easily overcome when these developers—whose expertise facilitates rapid application development—recognize and incorporate ease-of-use, knowing who is the intended

audience and acknowledging that the pointing device will be a touchscreen and not a mouse. They can then modify their website design to make it kiosk-friendly.

Studies indicate, **the time savings by developing a kiosk application with web-based tools can exceed 40 percent compared to using kiosk-specific designers**. This is an important consideration and cannot be overstated or taken lightly. Software development has long been the most expensive line item in a kiosk project. The cost of using kiosk-specific developers, even for a modest-sized project of fewer than 100 kiosks, can easily run well into the six-figures.

Significant kiosk application development time and cost savings can be realized by using this web-based solution. Several of the time-consuming aspects of an application will have already been written and can simply be tapped into by the developer. In addition to the substantial application development time savings, the tasks include:

- The coding to access separate peripherals
- The coding of I/O driver control testing
- The coding of component testing
- The coding of system testing
- The coding of all fault combinations and most error exceptions
- The monitoring of terminals and peripherals to check for out-of bound conditions
- The coding of network disconnect conditions

Addressing and handling the issues listed above comprise one of the main bottlenecks in kiosk application development. The old adage states *time is money* and by being able to harness many of these pre-written modules, developers not only can bring a kiosk project to market much faster but the costs of doing so are reduced dramatically. A good example concerns Input/Output (I/O) devices and the COM(munications) ports they use. Kiosk application developers must understand the various I/O device specifications, the correct timing to facilitate successful peripheral operation, register bit configurations and fully understand all possible errors that can occur—along with the proper remedial action to be taken. Similar complexities are found in the interaction between the kiosk processor and the terminal (most often a touchscreen) as well as with network connectivity.

More time and cost savings are derived from the fact that the interaction between the kiosk and peripherals has been already provided. We have found that kiosk developers often have to spend a surprising amount of time getting in touch with the technical support staff of the various peripherals they are trying to integrate into their kiosk solution in order to obtain updated drivers or have other technical issues resolved. Too often the documentation included with a device is unclear or inaccurate and time-consuming follow-up activities have to be initiated. This problem is exacerbated by the use of many different types of peripherals that have not been adequately tested to work together seamlessly with the motherboard and with the other peripherals.

In addition, a kiosk software platform tightly integrated with the hardware can be an enormously effective tool to address the complex conditions of kiosk terminal status monitoring and maintenance, the management and servicing of peripherals and the logic flow of all possible fault conditions that may arise during the kiosk's operation.

A truly integrated approach will insure that the hardware works with the software, that the kiosk itself is as close to bullet-proof as possible, and that only peripherals that can successfully function with this hardware/software platform will be utilized. Companies that use this methodology are highly selective in the peripherals they will incorporate into these kiosk configurations. If a customer wants a certain peripheral to be included in the kiosk, the companies who adhere to this integrated philosophy will perform rigorous quality assurance testing, making sure the new hardware will work with the software with no interruptions in workflow before allowing the product to be incorporated. This testing is not conducted overnight; often it goes on for many weeks or even months to insure the new hardware can be successfully incorporated into the overall kiosk solution.

A kiosk integration platform avoids the common problem of too much finger-pointing when a kiosk does not function as intended, fails frequently or frustrates the intended end-user. Too often, kiosk software developers blame hardware inconsistencies for these failures while hardware suppliers/manufacturers insist the software is at fault; it has been improperly written to "speak" to the kiosk and its peripherals. The end result is that the project is fraught with complications, problem resolution is difficult —or very costly— and the project ultimately fails to live up to its promise. An integrated kiosk platform in which the hardware and the kiosk platform software are manufactured and developed by the same company accomplishes the tight integration that goes a long way towards resolving those issues.

Benefits

Studies have shown web developers proficient in web programming languages spend at least 40 percent less time in developing a kiosk application than kiosk-specific developers. The majority of this timesaving is due to their being able to use pre-written modules that handle the interaction between kiosk hardware and software including all the low-level coding for the peripherals and the coding to deal with fault combinations. For example, hardware drivers—often an area requiring much time spent in custom programming and problem resolution—are already available.

In addition to this impressive timesaving, a web-based kiosk software platform opens up the kiosk application world to the world's ample supply of web programmers. Many of these programmers are located in emerging nations where salary levels are a fraction of U.S. salaries for comparable talent. Specialized web techniques, such as Mashups are suddenly available to kiosk application developers.

Being able to tap into Mashups, the fastest-growing application ecosystem in the world today, is another significant benefit of using a web-based approach. Developers no longer have to code every single activity that comprises a kiosk application. They can avail themselves of the many APIs (Application Programming Interfaces) that are now widely distributed on the web. Each piece of code required for a kiosk application no longer has

to be custom-developed. There are countless companies that now provide these APIs. Developers can take these APIs and create an application to meet their needs far more quickly and at a drastically reduced cost.

Because the cost of a kiosk project continues to be of prime importance, the ability to maximize the ROI through the use of an integrated approach is clearly the trend of the future. Being able to develop the application quickly, thoroughly and above all, accurately, and tying it into rugged and high-quality hardware presents a great opportunity for deployers. Because the expertise for application development is readily available in the web community, the time to develop an application that meets the requirements of the project is attainable much faster and far more affordably than if a custom kiosk developer has to be retained.

Another advantage is that when the company using the integrated approach uses hardware and a kiosk service platform from the same provider, a more rugged and dependable solution is produced. The days of trying to cram together a kiosk solution made up of mismatched and often incompatible components are over. It is much easier in the long run to start with an integrated approach than try to fit disparate pieces together.

Conclusion

The case for using kiosks in these challenging times has never been stronger. For many vertical markets, kiosks can provide the same or better services to the customer at far lower costs. Currently we are faced with a bunch of lemons. Kiosks can play a vital part in turning them into lemonade. Even if the world economy were running on all cylinders, using an integrated platform in kiosk design and implementation would make a great deal of sense. Additionally, using a web-based solution is an integral part of this system. The development costs are considerably lower because experienced web programmers are in plentiful supply and are much more affordable than dedicated kiosk developers.

Using the advantages of proven web Mashups will result in a much-reduced development timeframe while ensuring that the individual modules are thoroughly tested and highly suitable to the tasks required of them. Pairing these software enhancements with a rugged and extensively-tested hardware platform results in a robust and effective total kiosk solution. The many interface problems usually encountered when hardware and software are expected to work together seamlessly become almost nonexistent when this integrated approach is used. We highly recommend this innovative and well-conceived solution to kiosk development and implementation.