Mobile Media Purchases Are Difficult

Scott Weiss
Usable Products Company
143 West 29th Street, Suite 1102
New York, NY 10001 USA
+1.212.929.8599

sweiss@usableproducts.com

Abstract

The four components to the media download user experience are the mobile telephone handset, the carrier portal, the “vending machine” that dispenses media, and the media rights owner. Each component has its own agenda: the handset to be sold, the portal to provide information, the vending machine to dispense media, and the media rights owner to both protect and derive revenue from its property. The problem is that these disparate needs conspire to create a confusing user experience that frequently results in dissatisfied customers. This article presents examples of user interface problems and offers ideas for improvement.

Keywords

Mobile, Media, Download, Usability, Handheld, Phone, Wallpaper, Game, Ringtone, Ring, Tone

Introduction

Media, in the form of ringtones, wallpaper, games, music, and video can be purchased and used on mobile telephones today. However, acquisition of mobile media is a difficult proposition, riddled with bad user experiences, confusing user interfaces, hidden charges, and cheating vendors. The interplay of the players in the mobile media space is painful and competitive, and is getting further complicated by the introduction of digital rights management (DRM).

The Players

Media downloads to mobile phones require several parties: the handset (phone), carrier (operator), vending machine, and the media rights owner. Each of these players has issues, be they user interface, legal, logistical, or otherwise. Their interplay results in an extremely complicated and disturbing user experience. The following description of the players is for most implementations. A discussion of Qualcomm’s BREW UI, which is far more confusing, follows.

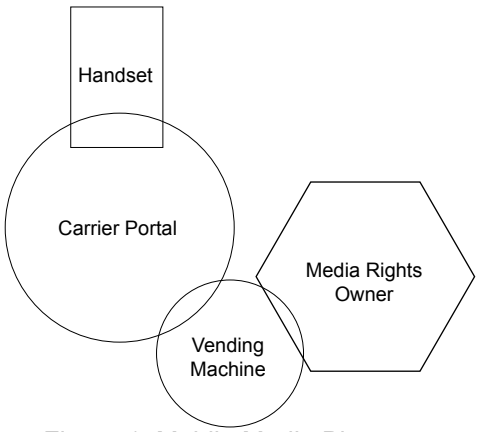


Figure 1. Mobile Media Players

Handset

The phone's user interface is used for accessing the carrier portal, and assigning the media once it is downloaded to the phone. In between, the phone's user interface is mostly irrelevant, though the hard buttons can cause some confusion. Namely, if there is a 'Menu' button, to what does the menu refer—the phone, the browser, or the application being viewed? The same is true of a 'OK' or 'Select' button. When is the OK or Select button active? There is typically no obvious indicator that the button has any function.

Consumers must get from the base handset UI to the carrier portal in order to find media to purchase. Better handsets provide cross-links from media lists to deeper locations within the portal or even directly to the vending machine.

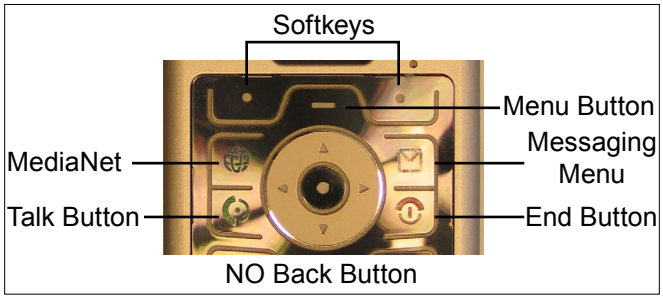


Figure 2. Complex Keypad

Carrier Portal

Carrier portals are often called "walled gardens" because the portals are generally nice places, but they can't be exited into the wider web. Most operators have a catchy term for their portals, such as "T-Zones" from T-Mobile or "Live!" from Vodafone. These catchy phrases tend to confuse consumers. Portals generally guide consumers to entertainment and news content, but also to vending machines for media that can be purchased for their phones, like ring tones, wallpaper, and games.

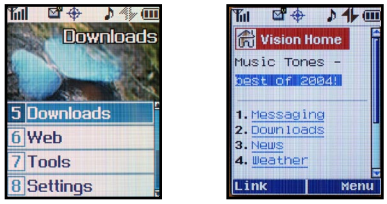


Figure 3. Carrier Portal

Vending Machine

The vending machine is always invisible to the consumer. It is linked directly to the carrier portal. The vending machine presents the consumer with lists of media to purchase, and guides the consumer through the process of acquiring media. Vending machine services are always provided by a third party: carriers are not in the business of selling ring tones, but instead provide access to the vending machine. Being a third party, the vending machine knows little about the handset and the media that it presents for sale.

Vending machines for mobile media are like vending machines for candy or cold beverages. The machine itself knows how to take money and dispense things, but the user experience is limited to just that: vending machines tend not to be helpful, nor do they present a nice way to sample media before purchase.

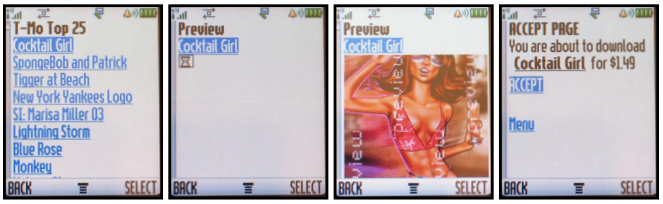


Figure 4. Vending Machine

Media Rights Owner

The owner of the media figures into this equation by way of legal relationships. Media rights owners must be paid for their property, but they rely on the carriers and the vending machines to sell it to consumers. Information about the media is scant—usually limited to a one-line text description—and merchandising is outside the media rights owner's control.

Task Flow

In most implementations, consumers start with the idea that they want to enhance their phone with the addition of new media. Their mental model stops there, and they go into “hunting mode” in order to figure out how to accomplish this seemingly-easy task.

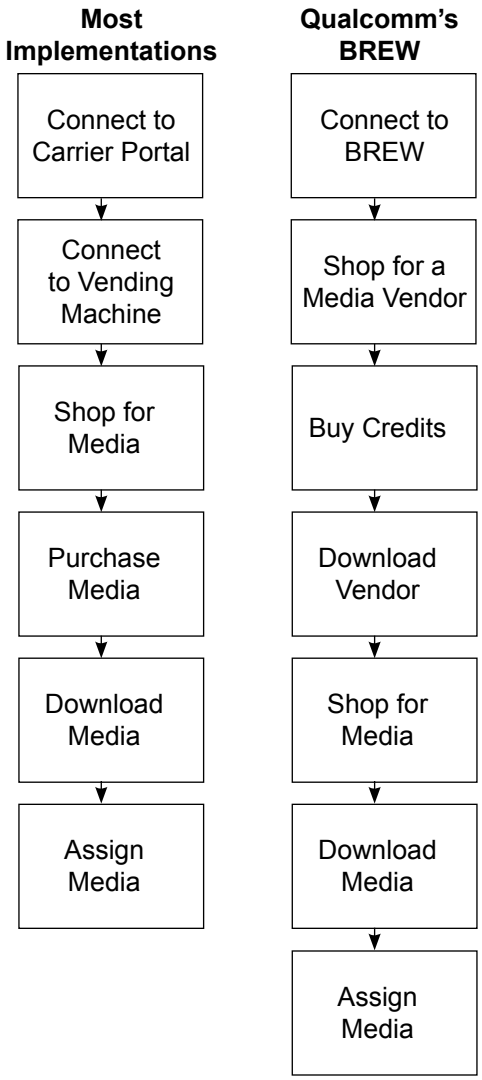


Figure 5. Task Flow

Connect to Carrier Portal

There are several methods for connecting to the portal, which differ by handset and by carrier. In Figure 2, “MediaNet” can be accessed from a hard button on the handset keypad. Typically, however, the portal is accessed from the handset’s main menu. Sometimes, as in Figure 6, the portal has several entries in the handset’s main menu.

There were five sections from the Main Menu that allowed access to some section of the Media Net.

Figure 6. Main Menu Portal Entry Points

Connect to Vending Machine

The most effective means of accessing the vending machine is from existing media lists, as shown in Figure 7. However, the direct access method is the first “hidden charge,” since mobile data fees start accumulating once these links are accessed, often without warning to the consumer.

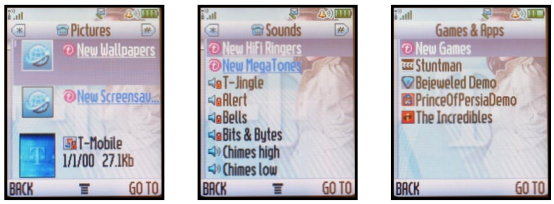


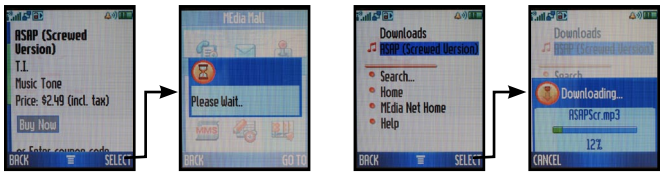
Figure 7. Media List to Vending Machine

Shop for Media (From the Vending Machine)

Shopping for media is usually a straightforward but uninspiring perusal of lists. Some shopping UIs feature previews, but the number of preview-capable vending machines is surprisingly low. Vending machines look much like other areas of carrier portals, mainly because vending machines are almost always built in WAP.

Purchase Media

It is always easy to spend money with vending machines, but it is not always easy to get one's media, shown in Figure 8. In this case, one first must purchase the media, then proceed to the download step before the media is transferred to the handset. In Usable Products' *Media Download Usability* report (2005) [1], 6 of 20 respondents (30%) failed to initiate their download from this page at least once during the test sequence. The carrier, vending machine, and media rights owner all got paid. However, the consumer did not get to enjoy the media: not a good recipe for repeat purchases.



Detail of the process required to purchase a Ringtone showing the Downloads Page (third from left). 6 of 20 respondents failed to initiate their download from this page at least once during the test sequence.

Figure 8. Purchase Without Download

Download Media

Downloading the media to the handset is highly handset-specific. In Figure 9, the media got downloaded, but the next step was to "store" it or else it will remain in a temporary location within the handset. In the Usable Products *Media Download Usability* study mentioned above, 22% of respondents missed the "Store" softkey.

Newly downloaded wallpaper (top) and ringtones (bottom) were displayed immediately after their download. From these display screens the "Store" softkey, which displayed a list of storage options, was often overlooked.

Figure 9. Download, Then Store?

Assign Media

Assignment means setting the ringtone, wallpaper, or playing the game. Some systems allow instant assignment as a handy option, such as in Figure 9, above. Other systems require complicated navigation through handset menus in order to locate the just-purchased media.

The Wallpapers were displayed in two groupings in the Settings menu, Wallpaper (left) and Downloaded (right). Pressing left or right switched between these two groups.

Figure 10. Assignment

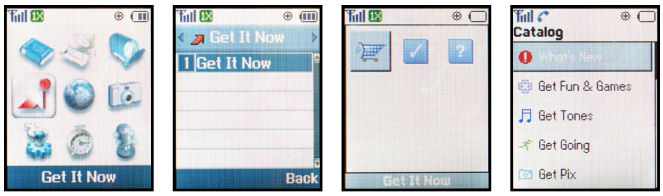
How the BREW UI Model Differs

In non-BREW systems, there is only one vending machine that aggregates content. BREW aggregates vending machines, which are Java applications that contain content. When the content is a ring tone, the model is unusable because BREW hides the content of the individual vending machines. BREW requires purchasing credit from the vending machine, which must then be downloaded and run in order to view the content available for sale.

The BREW model is akin to a room full of vending machines, for which only their names are visible. Consumers must select a vending machine based on its name, and either download a trial version or purchase credits. Only once the vending machine is downloaded and run are its contents visible.

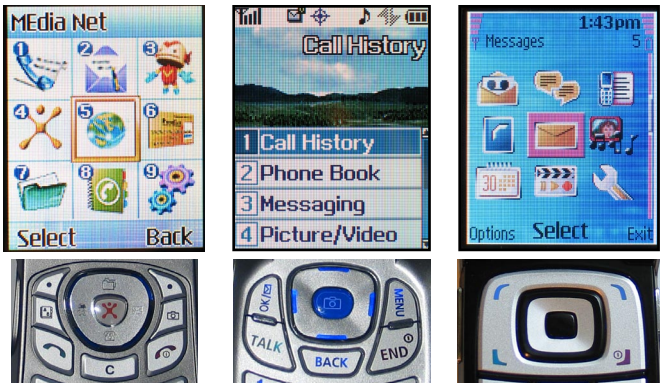
It's all much simpler when the vending machine has only one item, which is usually the case for games. Most games are purchased by name, which is also the name of the vending machine, such as "Pac Man" or "Tetris." The trial version concept also works for games, since most consumers are familiar with game trial versions on PCs. The trial version concept does not work for vending machines selling ring tones and wallpapers, however. In these cases, the vending machine cannot dispense anything since it has no credits.

BREW further falters because credits must be purchased from the BREW home page; credits cannot be purchased from within a given vending machine. The consumer must instead go back to the BREW service entry point and purchase credits there, and re-download the media vending machine.



The process to get from the handset's main menu to the Get It Now catalog.

Figure 11. BREW Access



The left handset has a "C" key, the middle handset has a "BACK" key, and the right handset has no "Back" key at all. The left and center handsets have an unlabeled center button, but the right handset's center button functions as a third softkey, with an accompanying label ("Select").

Figure 12. Keypad Variations

Key Opportunities & Associated Implementation Challenges

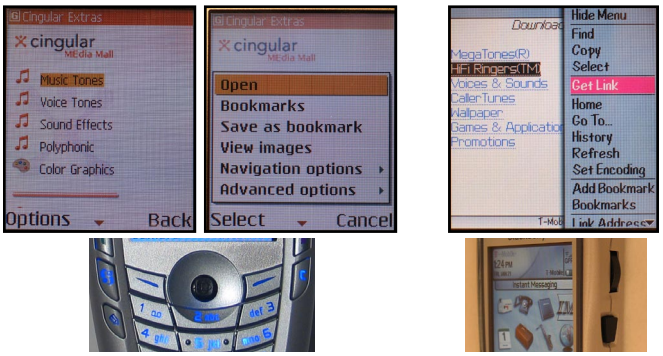
Except for BREW, all mobile-phone based media purchase engines are based in WAP, and they are unattractive. Few systems feature well-integrated previews, especially for ringtones.

Tight Integration with Handset Buttons

Existing WAP applications know little about the handset buttons. Increasing the device information available to the developer would be a great step, but would make the developer's job even harder. Some phones have two soft keys and only an up/down navigation control. Other phones have a four-way navigation control. Increasingly, phones feature a four-way navigation control that allows a center press for "Enter," in addition to the two soft keys. These three variations result in a tragic compromise between usability and universal handset support. Ultimately, developers must come up with a UI for each type of handset, but as mentioned above, the knowledge of the handset buttons is limited and often unreliable.

Increased Interactivity

Today's WAP-based media shopping experience is a far cry from the high level of interactivity otherwise enjoyed within the handset, such as for the address book, text message entry, and ring tone assignment. WAP 2.0 has increased the available widgets, but the UI is still forms-based, and does not afford direct canvas interaction, like Java does. SVGt 1.2 (Scalable Vector Graphics Tiny Edition) promises a high level of interactivity and less complex programming, but as of early 2006, it is not yet deployed widely on mobile terminals.



At left, the menu is accessed with the left softkey ("Options"), instead of the center press on the joystick. At right is a context menu accessed from the lower side button, navigated with the upper roller control. Notice the difference in appearance between what is visible underneath the menus, WAP renderings, to the menus, which are handset-generated, and more attractive.

Figure 13. Menuing Variations

The BREW shopping experience, as noted above, is broken. However, the individual BREW stores are highly interactive and engaging, being Java-based at their lowest level. All BREW stores are built in Java, a highly-customizable and interactive programming environment. The problem is that all the stores have different user interface paradigms, and they all are accessed through the BREW portal, which differs from the operator's portal. This problem begs a standardized shopping user interface, optimally pre-loaded to the phone. A standardized shopping user interface would offer user interface consistency, requiring consumers to learn only a single UI, even if that UI is deployed with different visual branding to enable vendor differentiation.

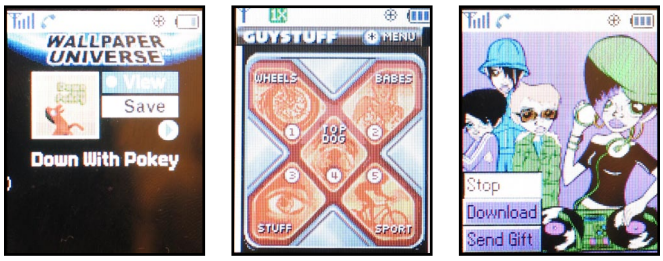


Figure 14. BREW Stores

Pre-Loaded Shopping UI

A resident shopping UI, linked from the existing media lists, would be the ideal. This UI could be designed and built by the operator (or a vendor engaged by the operator), for consistency across handsets. Or the UI could be designed and built by the handset manufacturer, for tighter integration with the handset.

The key problem with this strategy is the operations. While the UI would be much more usable, getting someone to design and build it would be quite a challenge. Manufacturers are not media vendors, and so there is little incentive for them to design and build this UI. Third party companies could do the UI design, but they need to be paid—and each handset has a unique user interface, requiring a great deal of integration effort.

Today's handset-based shopping user experience is entirely via WAP, BREW, or other third-party technology. Making that shopping experience native to the handset would enable greater interactivity and consistency, but it would also place a much stronger burden on the handset manufacturer, who stands to gain little from improving the shopping experience, since it is unlikely to sell more handsets.

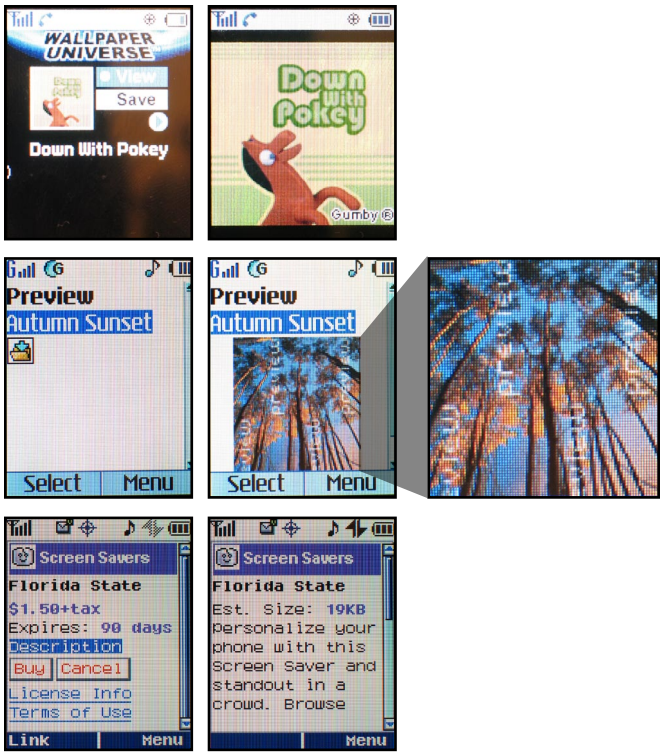
Above is a Java-based shopping user interface. Below is a WAP-based shopping user interface.

Figure 15. Java vs. WAP

An operator-built Java-based shopping UI is therefore a likelier outcome, but the key challenge there is the relationships with the media vendors [2]. If the operator builds the applications, the operator is turning into a vending machine company, a business model that is probably undesirable. That leaves the operator to license and distribute a UI application to the handset manufacturers, who would need to integrate it with their handsets. This last strategy represents the best opportunity for increased usability and consumer satisfaction, but is likely to also be quite distasteful to the manufacturers.

Richer Media Previews

Today's previews, when offered, are hard to get to, and are often not very enjoyable. It is easy for consumers to become confused by them, not knowing if they are getting the preview or magically getting free media. However, that challenge lies in user interface design, and proper presentation to the consumer.



At top, a Java-based preview set. At center, a WAP-based preview set. Notice the "exploded" graphic showing the 'preview' graphics overlaid on the image. At bottom is a text description of a graphic.

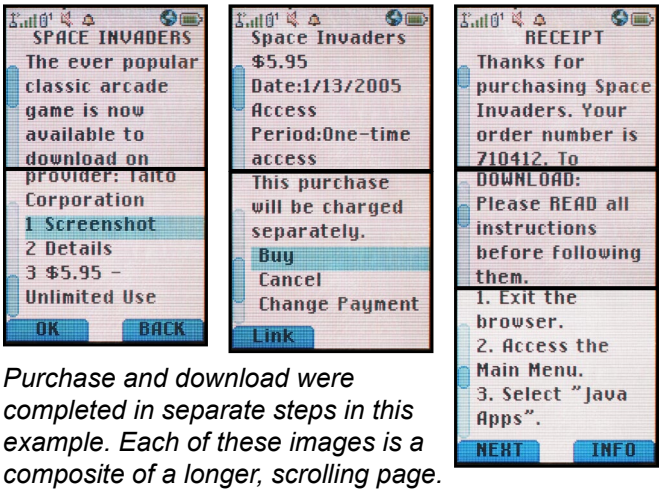
Figure 16. Previews

Usability Testing

We must assume that the vending machine companies are *not* doing user testing, as their user interfaces are so awkward and confusing. Testing the usability of the media vending user experience would enable all parties to know exactly what the pain points are. Observing user tests would enable key players to spark ideas for how to reduce this pain and the associated inevitable high Customer Support costs that accompany bad user interfaces. Furthermore, better user interfaces for media vending would certainly increase media sales. Happy customers buy more, and more frequently.

Following are some examples of painful UIs discovered during usability testing.

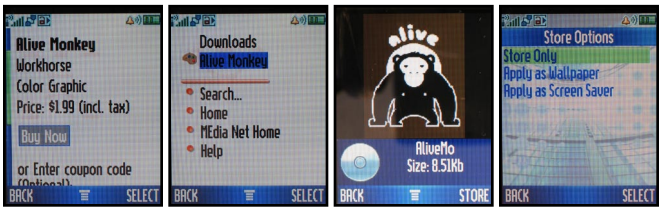
Purchase, Then Download



Purchase and download were completed in separate steps in this example. Each of these images is a composite of a longer, scrolling page.

Figure 17. Purchase, Then Download

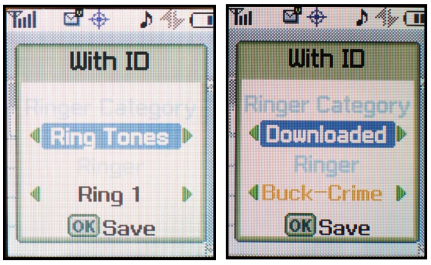
Save as an Extra Step



Purchase, save, and download were all separate steps.

Figure 18. Save As An Extra Step

Downloaded Media Hidden



Downloaded media was in a separate list from built-in media.

Figure 19. Downloaded Media Hidden

Conclusions

The media download market is growing, but to some degree it is through customer pain rather than customer satisfaction. The hardware is partly to blame, but the key issues are three software sources and their lack of usable integration. The combination of carrier portal, vending machine, and handset-resident software has a steep learning curve. Java-based technology offers the best opportunity for technology to help solve this problem, but usability testing of the designs before they are finalized will yield the best results.

References

[1] “Mobile Media Download Usability.” Usable Products Company. New York: January, 2006.

[2] “An Alternative Business Model for Addressing Usability: Subscription Research for the Telecom Industry.” Scott Weiss. *Interactions*. ACM, New York: July + August, 2005.