

Journal of • Virtual Worlds Research

jvwresearch.org ISSN: 1941-8477

Volume 2, Number 3
Technology, Economy, and Standards.

**Community
Creation
Commerce**

Artwork by Anshe Chung Studios

Volume 2, Number 3

Technology, Economy, and Standards

October 2009

Editor

Jeremiah Spence

Guest Editors

Yesha Sivan
J.H.A. (Jean) Gelissen
Robert Bloomfield

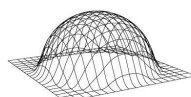
Reviewers

Aki Harma
Esko Dijk
Ger van den Broek
Mark Bell
Mauro Barbieri
Mia Consalvo
Ren Reynolds
Roland LeGrand
Vili Lehdonvirta

Technical Staff

Andrea Muñoz
Kelly Jensen
Roque Planas
Amy Reed

Sponsored in part by:



virtual worlds research consortium

**The Journal of Virtual Worlds Research
is owned and published by:**

The JVWR is an academic journal. As such, it is dedicated to the open exchange of information. For this reason, JVWR is freely available to individuals and institutions. Copies of this journal or articles in this journal may be distributed for research or educational purposes only free of charge and without permission. However, the JVWR does not grant permission for use of any content in advertisements or advertising supplements or in any manner that would imply an endorsement of any product or service. All uses beyond research or educational purposes require the written permission of the JVWR.

Authors who publish in the Journal of Virtual Worlds Research will release their articles under the Creative Commons Attribution No Derivative Works 3.0 United States (cc-by-nd) license.

The Journal of Virtual Worlds Research is funded by its sponsors and contributions from readers. If this material is useful to you, please consider making a contribution. To make a contribution online, visit: <http://jvwresearch.org/donate.html>



Journal of • Virtual Worlds Research

jvwresearch.org ISSN: 1941-8477

**Volume 2, Number 3
Technology, Economy, and Standards
October 2009**

Barriers to Efficient Virtual Business Transactions

By ArminasX Saiman, Virtual Business Owner/Operator

Abstract

With the availability of business transaction capability within virtual worlds like Second Life, enterprising individuals and teams have established businesses that operate entirely within the realm of virtual reality. These wholly-virtual business operations act much like real-life businesses; they must develop and manufacture products or services, advertise, sell and fulfill deliveries. A complete lifecycle of business events takes place within the virtual world.

The virtual business owner is presented with a seemingly complete set of tools to perform all actions required by each stage of the business lifecycle. However, over the past several years virtual business owners have begun to discover limitations and missing elements in these business transaction protocols. This paper will identify the more notable limitations facing today's virtual business owners.

The author has owned and operated such a virtual business for over two years, beginning from sales of a single virtual product on a web-based sales service in 2006, growing to a large in-world operation covering 1/4 of a region and selling over 200 unique products today. In real life, the author is a senior Information Technology manager.

Keywords: Second Life; economics; business; transactions; standards.

This work is copyrighted under the Creative Commons Attribution-No Derivative Works 3.0 United States License by the Journal of Virtual Worlds Research.

Barriers to Efficient Virtual Business Transactions

By ArminasX Saiman, Virtual Business Owner/Operator

One of the more unique features of the Second Life virtual world is its economy. Based on a virtual currency, the Linden Dollar, the Second Life monetary platform provides the means to reliably purchase and receive virtual items. While interesting, it became much more important when an easy-to-use currency exchange emerged, permitting avatars to convert Linden Dollars (L\$) into US dollars and vice versa.

The ability to transform virtual currency into real-world money spurred the development of a wide variety of virtual businesses catering to the needs of Second Life residents. The continued expansion of some of these businesses demonstrated successful approaches to virtual business operation. Intense competition within virtual market segments caused virtual business owners to adapt and increase their effectiveness in various ways.

Successful virtual business owners quickly realized that the process of business in virtual reality is in fact very similar. They must perform a repeated series of steps that gradually improve their product offerings, and ultimately their return on investment. In simplified form, the author's diagram illustrates the basic steps involved in operating a virtual (or any) business:

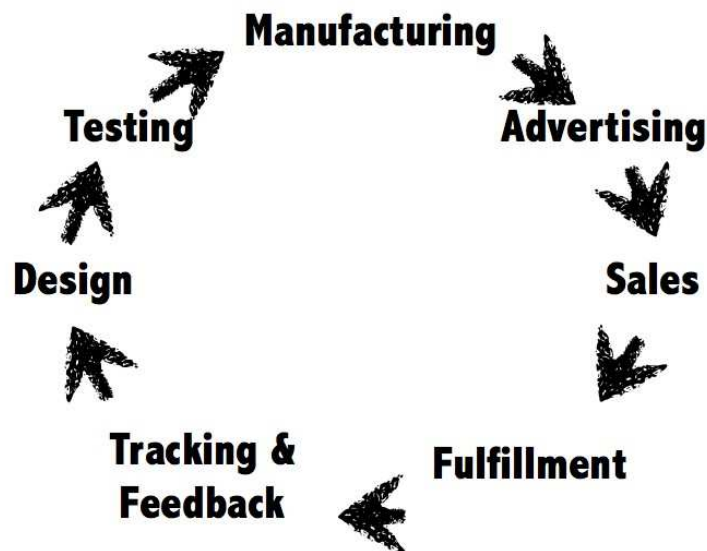


Figure 1: The Business Cycle

Second Life's economic features would appear to permit these activities to take place, and indeed a significant amount of business activity takes place. The key features of the Second Life platform that permit business transactions include:

- Ability to mark items for sale at a specified price
- Ability for avatars to receive goods after their Linden Dollar account is debited during "Buy" transactions
- Ability to "Gift" (or transfer) funds directly to another avatar
- The right to retain ownership of items created
- A permissions regime capable of protecting the intellectual property of the maker

Nevertheless, experienced virtual business operators encounter issues and inefficiencies during the business cycle, some of which are due to the primitive state of the features and the protocols they are built upon. It is the view of the author that efficiencies in virtual business could be obtained by instituting several additional standards.

In the absence of the required features and standards, virtual business owners have resorted to alternative solutions, sometimes involving complex custom made scripts and unusual manual procedures. Often these "by-passes" are unique to the store and thus are unfamiliar to visiting shoppers. Unfamiliarity is an impediment to business, as a percentage of shoppers will be baffled by even slight differences in procedure, resulting in fewer sales.

The Design, Testing and Manufacturing stages of the business process generally work well, but enhancements to protocols may be required to assist the efficiency of the other stages.

Advertising

Problem: Identification of Originating Store

Advertising in a virtual world is notoriously difficult, but a complete analysis of that subject is beyond the scope of this document. Nevertheless, one of the most effective advertising mechanisms is "word of mouth," in which great products are mentioned between friends, or more specifically, the objects in question are inspected to determine where they can be purchased. In fact, the inspector must execute a tortuous sequence to determine the location of the originating store.

A typical word-of-mouth sequence proceeds as follows:

- A object like a shirt or a car is identified as desirable
- The object is edited
- The maker's identity is located within the object's properties
- The maker's profile is opened
- The maker's profile's "Picks" and/or "Classified Ads" are inspected manually in an effort to determine the originating store. This is not always possible as Profile and Classified information is manually entered by the maker and may be out of date or even missing entirely.

Eventually, if avatars are sufficiently desperate for a purchase, a teleport to the identified store takes place where hopefully a purchase is actually made. However, in larger stores determining the location of a specific product can be quite difficult.

Clearly, this process is not only laborious, but also inaccurate and with no guarantee of a successful result. A standard could evolve to overcome this scenario, where objects could more directly enable a purchase. Potential solution approaches could include:

- Associate the object with a “store” or business identity rather than a personal avatar identity
- The object contains location information for the originating store, perhaps with a linked teleport button displayed in the object’s properties. The location could be either the store’s default location, or the current location of the product on the store shelves
- The object itself could remain in a “for sale” state, or quickly link to an external service that presents the object for sale

Sales

Problem: Custom Invoicing

Makers typically sell their wares in multiple ways, beyond the simple “click to buy” method provided in Second Life. Frequently makers are requested to create one-of-a-kind items by interested customers. Custom-made products are quite profitable, but there are no standard methods of invoicing customers. Today makers use a variety of techniques, including:

- Boxing up the custom-made products and setting the box for sale at the agreed price. This requires extra effort by the maker and also requires the buyer to virtually travel to the location where the custom box is “rezzed”
- Transferring a copy of the custom box to the buyer’s inventory and awaiting a gift of sufficient funds in return from the buyer. However, this requires one party to move first, and sufficient trust might not be available between parties for this to succeed

Potential solutions could include:

- A standard in-world escrow service that accepts items from makers and releases them to buyers when purchase conditions are met
- Linkage to an offline purchasing service that provides private and exclusive transactions
- An ability to purchase items without being in proximity to the virtual object. For example, a custom object could be offered for sale directly by an instant message purchase invitation

Problem: Shared Ownership of Salable Products

Many in-world businesses are owned and operated by partnerships, and this poses a problem for revenue sharing among the partnership. The existing sales protocol assumes that there is one and only one avatar selling a given object. But in many cases partnerships exist where a complex business is composed of avatars with differing skills. How can revenue be shared when the objects being sold are owned by only one avatar?

Ad-hoc solutions do currently exist, but in general they are all of the form of a vending script that splits incoming payments into shares for collaborating makers. However, the script itself must be owned by one avatar, and therein lies the problem – significant trust must exist between the collaborators, lest the script or its configuration be unethically modified. Worse, there are many types of such vendor scripts that vary significantly in usage and configuration, and many product types do not show well if sold from vendor boxes.

Potential solutions could include:

- A standard revenue-splitting vendor issued by a trustworthy source, such as the Grid's owner
- Ability to mark for-sale objects as being owned by a group, with a method of specifying payment splits

Problem: Gifts from One Avatar to Another

While shopping may be a very popular activity in Second Life, there is an underlying assumption that is incorrect and leads to inconvenience for many: Avatars always shop for themselves. In fact, avatars frequently shop for others; they buy gifts. Sometimes this is not an issue, because purchased objects might be marked with "Transfer" permissions, and in that case they can be freely given to a gift recipient. However, many products are marked with "Copy" permissions only for various reasons. This means that once the gifter purchases the object, it cannot be given away, even though the gifter has no intention of using the item.

In practice this issue is very problematic, as gifters must teleport the recipient to a store location and manually indicate which item to purchase. Another approach is to negotiate directly with the product's maker; the maker is paid directly and asked to transfer a copy of the item to the recipient. The gift typically manifests itself for the recipient as an object offer from an unknown avatar (the maker). Sometimes these offers are refused because the recipient has no expectation of an incoming object from an unknown avatar. Surprise "Copy" presents are thus a rarity.

Potential solutions could include:

- The ability to transfer "Copy" items, perhaps by transferring all copies of the item at once, thus preserving the original intent of "Copy" permission
- The ability to transfer "Copy" items once after purchase

Problem: Gift cards

An alternative to purchasing a gift is to provide a gift card, with which the recipient can theoretically purchase any item within the voucher's limit at the designated store. This practice is well-understood in real-world stores, but within Second Life there are no such standards.

De facto solutions are quite varied, as a variety of approaches are used to implement gift cards. The simplest is a notecard containing an explanation of the gift and redemption process in text. This notecard-voucher is given to the recipient. Meanwhile, the maker has recorded a credit for the recipient and manually decrements it as the recipient selects merchandise. This labor-intensive approach does work, but is entirely non-scalable and cannot be used in larger stores.

More complex solutions exist, involving sophisticated vendor scripts and gift tokens as objects. Gift card solutions are unpredictable from store to store, and as a result shoppers can become confused, overwhelmed or insufficiently interested to learn how to acquire and use a gift card from an unfamiliar device.

Potential solutions could include:

- A standard scripted object representing a gift card that is associated with a specific store. Makers can provide them to gifters who fill them with the desired amount of Linden dollars.
- Stores could sell currency units usable only at that store. The currency would be at par with the Linden dollar

Problem: Searching for Products

Shopping is a popular pastime, and is often done on a whim. But there are many times when very specific items are required. A typical scenario is the requirement for themed attachments for an event. Faced with such unusual needs, the shopping avatar must embark on a potentially endless search for the desired items. Some may enjoy a shopping challenge of this type, but many do not and become quite frustrated by the experience.

Solutions employed are quite varied, ranging from brute-force store-by-store search to using the external web-based XStreetSL search facilities. If a candidate item is identified through XStreetSL, the item is not necessarily purchased on XStreetSL, as the buyer may wish to visually inspect the item in-world. To do so, the buyer must translate the XStreetSL listing to an in-world location. Sometimes this is problematic if the XStreetSL seller has incorrect listing information.

The in-world search may also be used, but it is notoriously difficult to use due to limited results displayed, lack of filters, rankings that seem to be based on the wrong criteria and extraneous listings produced by vendors trying to “game” the search system.

Even if a buyer has identified the store selling the desired item, the search does not end. A large store may occupy an entire sim, with shelves, walls, levels and buildings full of hundreds or even thousands of products, any one of which might be the item. In-store searching can be extremely difficult as there are no standard scanning mechanisms for locating a specific item. If the search duration persists beyond the tolerance of the buyer, they may give up and go to another store. In the real world, physics tends to geographically trap the buyer in the real world store until the item is located, whereas in a virtual world, the next store is merely a teleport away.

Some stores overcome this difficulty by hiring onsite “guides” or “shopping assistants” to receive visitors and direct them to the appropriate item. However, this solution is manual, expensive and can be detrimental to future sales if inappropriately skilled individuals are hired into the role.

Potential solutions could include:

- A vastly improved in-world search facility, perhaps aided by an ability to tag for-sale objects with useful and standard meta-data
- An ability to link off-world sales systems such as XStreetSL with the for-sale object's current in-world location
- A standard capability to detect specified for-sale items within a scanning radius
- Exposure of for-sale object properties via XML, such that third parties may develop increasingly sophisticated or specialized search services

Problem: Reselling

Wal-Mart and similar large distribution operations do not really exist within Second Life, even though the presence of a large centralized shopping station would vastly simplify the search for desired items. Large distribution operations do not manufacture the goods themselves, they simply acquire them from an array of suppliers and place them on their shelves – the value for shoppers being the known central location and near-guarantee of finding the goods.

But such operations would have great difficulty implementing this approach, as a specific avatar must own every object. While a real-life supplier is paid per item by Wal-Mart, this is not possible in Second Life. Typically the maker is paid a one-time flat fee for the reseller's right to sell an infinite number of copies. The maker does not necessarily receive compensation proportionate to the eventual value of the item since makers frequently do not have the skills to negotiate an appropriate payment. As a result of these difficulties few mega-stores have emerged.

Potential Solutions could include:

- An ability to sell bulk quantities of items at set rates to a reseller
- Automated royalty payments for resold items, based on rates established when the reseller purchases the master item from the maker

Fulfillment

Problem: Returns

Invariably, products may be found deficient, incorrect or simply purchased in error, resulting in a request for product return. It's good business practice to accept returned items, as the goodwill generated usually far exceeds the value of the return. However, there are few standards in place to aid this process.

Typically the maker must receive and acknowledge a manual request for a return via instant message. The maker must then verify that the product was actually purchased via a manual inspection of the transaction log, sometimes based on an inadequate description from the requestor. This process is especially difficult if the buyer and maker do not speak the same language. If the transaction is verified, then the buyer must actually return the item. However, if it's a "Copy" item, then returns do not make sense: the buyer cannot transfer items back to the maker for verification and the maker must trust the buyer to delete all copies from their

inventory. In many cases the maker simply permits the buyer to keep the undesired items. Payments are returned by a manual gift payment from maker to buyer, which of course may be incorrectly entered.

Potential solutions could include:

- Just as there is a protocol for purchasing an item, there could be a protocol for returning an item, perhaps involving a pie-menu “request return” command for items that had previously been purchased
- Standards regarding product naming in the transaction log could make it easier to validate the purchase
- A more sophisticated permission regime could permit the return of “Copy” items

Problem: Upgrades

When products are enhanced and new versions are created, the maker has two possible actions: they can attempt to upgrade all previously sold versions, or they can simply place the new version on sale and existing customers may upgrade by a new purchase. In some cases, the nature of the product requires all deployed instances to be a common version and the maker has no choice but to push out the upgrade.

Solutions used by makers today again are quite varied as built by many ingenious scripters. Some products will self-upgrade by enquiring against a server, which may give a new copy to the owner of the item, others may notify the buyer that a new version may be obtained by presenting themselves at a proximity sensor. Yet another approach is for the maker to run a script that transfers new versions en masse to all known customers. These inconsistent approaches are, of course, confusing to customers who may not understand what to do. Customers may end up with multiple versions of the same object and if they are not careful, they may use them incorrectly.

Potential solutions could include:

- A built-in ability for items to upgrade themselves that could be enabled by makers
- Standard product naming conventions would simplify any upgrade approach
- An ability to easily determine the current owners of a given product

Problem: Discounts

Occasionally the maker may wish to offer a discounted price for existing customers of a product. For example, a new version of a product might be sold to existing customers at half price. Imagine if two identical products were displayed, one at half price and the other at full price for new customers. New customers would simply purchase the less expensive upgrade object and gain full benefit. Other solutions become very complex, particularly if the maker wishes the customer to have a straightforward upgrade experience.

Potential Solutions could include:

- A standard means of detecting the presence or ownership of a copy of a given object
- A product registration service in which scripts could easily determine whether an avatar has previously purchased (or received) the item

Problem: Delivery confirmation

For the most part, the buying and delivery sequence works very well. However, occasionally it does not, as the grid or client software may be having difficulties. During these periods advisory messages sometimes warn buyers not to purchase items because they might pay but not receive the goods. Nevertheless, such risky purchases are still attempted and some do indeed fail.

Typically the maker is contacted by the customer, who often accuses the maker of deliberately taking their money and not delivering the item. After lengthy and awkward explanations of database problems, either money is returned or another copy of the item is transferred to the customer. However, the maker never knows for certain whether the item's purchase actually failed. The customer could simply claim so, in an effort to obtain a second copy. In practice the maker must assume the customer is honest and follow through.

Potential Solutions include:

- A delivery confirmation sequence that records the successful delivery of each sold item
- More robust sales transaction handling
- A method of redelivering or re-running suspect transactions

Problem: Link to Offline sales

Most makers have both an in-world and web-based storefront, typically at XStreetSL. Like in-world purchasing, the XStreetSL purchasing experience is reasonably well done and easy to use. However, the maker must maintain two separate inventories of products: one in-world and one on XStreetSL. If a new version is produced, then two storefronts must be updated. While this may seem to be a minor matter, it becomes larger when multiple objects are involved. For example, a maker might produce a line of clothes with separately purchasable colored versions. There might be two dozen such items. But the same scale of work to place them on sale in-world must be repeated to place all of the same objects for sale on XStreetSL. Because the linkage between XStreetSL and the Second Life grid is tenuous, twice as much work is required of the maker.

Potential Solutions could include:

- Direct mapping of properties between in-world for sale items and the external service
- Exposure of in-world for-sale items' properties via XML, which could be consumed by multiple external web-based sales services

Tracking & Feedback

Problem: Accounting & Transactions

Any business owner should monitor the progress of sales in an effort to weed out poorly performing products and produce more of the best-selling items. To do so, the maker must inspect the transaction log, which is available in HTML, XLS or XML formats.

Unfortunately, the transaction log is actually a mix of store-sold items and personal activities. While the data is easily obtained, it must be manually inspected to remove non-store transactions that would obscure true store activities. Some makers solve this issue by creating a separate avatar that owns all items, thus isolating the accounting information. In some cases, that solution may not be viable because the avatar's reputation may be required for marketing reasons and must appear directly associated with the products.

Potential solutions could include:

- If items are sold by a store entity and not an avatar, then the store's transaction log would be immediately more useful for analysis
- Tags applied to objects could be echoed in the transaction log, providing a means to identify not only sales items, but categorize them as well

Problem: Product Version Tracking

Products are often improved over time, resulting in multiple versions of a salable object. Responsible makers attempt to track the many versions by appending a version code to the object's name or description, but this process is entirely manual and achievable only by the most disciplined makers. Novice makers typically have no notion of the need for versions and run into complications later.

Potential solutions could include:

- A "version" property for objects separate from the name or description fields
- A simplified versioning service similar to those used for managing software components

Beyond Daily Business Operation

Problem: Selling a business

Just as products can be sold, so can an entire business. But two problems exist. First, the objects and property comprising the business are mixed with other items in the inventory of a specific avatar. Selling the business means these items have to be specifically identified and transferred to the purchasing avatar. In some cases there can be a great many items, including those not currently for sale, and the transfer exercise becomes burdensome.

Secondly, two or more avatars in partnership own many businesses. In these situations, sale of a business becomes a complex matter of inventory searching, multiple object transfers and several manual money transfers, all of which offer plenty of opportunity for errors.

This issue increases the friction encountered by business sales, and perhaps is one of the factors preventing larger virtual businesses to emerge: a virtual business cannot easily grow by acquisition.

Potential Solutions include:

- Establishment of a business as an entity that can own objects. If this were so, then the business and all of its property and objects can more easily be sold
- Tagging of inventory items could enable quick identification of business-related objects

Vendors as Symptom

Vendors in a virtual world are scripted objects that present multiple objects for sale. An avatar may load a vendor with a series of salable items, which then may be viewed successively by shoppers. Should the shopper wish to purchase, scripted buttons on the vendor can complete the transaction.

Virtual businesses often use vendors to overcome prim limits on their parcel. Should a business owner wish to sell 500 products but have only 200 free prims available, there are few options other than resorting to vendors. The vendors enable sales to occur without incurring use of scarce parcel prims.

However, there are issues with vendors from a customer point of view, some of which are a serious detriment to effective sales:

- Vendor user controls are quite varied and unfamiliar controls mean a lower probability of sales
- Vendors often require shoppers to repeatedly click through long sequences of products in order to (possibly) find their desired item. However, each click sequence is typically quite slow to complete, most often due to the time required to load the product's image texture. While some vendors intelligently pre-load upcoming textures, the process is still awkward and many impatient customers simply give up and move on
- Products are often shown only as an image, whereas rezzed objects can be viewed fully in three dimensions, perhaps with animated actions. Certain types of products cannot be properly inspected by image only, and thus sales are less likely

It would appear that vendors should only be used as a last resort, but in spite of the issues above there is another reason business owners often choose vendors. The vendor object's script provides a layer in which missing business functions can be provided. The most sophisticated vendors provide advanced functions, well beyond the standard click-to-buy virtual business protocol. Advanced features might include an ability to rapidly deploy new products across multiple sales locations or tracking product sales in a manner independent from the confusion of the avatar transaction log.

Some virtual business owners have come to depend on such advanced sales systems, and therefore the existence of such vendor systems is a symptom of the deficiency of standard protocols within the Second Life economic platform. It is likely that additional ideas for protocol extensions may be found by inspecting the most popular features provided by sophisticated vendor systems.

Conclusion

It is clear that development of large-scale virtual businesses is discouraged by these factors. If large-scale virtual business activity is desired, then these issues and others must be addressed.

Several common themes repeatedly occur within the potential solutions above, suggesting these may be priority items:

- Enhancing the object permissions to be more flexible
- Establishing a business entity capable of holding salable objects and property, and being jointly owned by several avatars
- Tagging of objects
- Exposure of for-sale object properties via XML

While a significant amount of virtual business activity currently takes place, it appears that business efficiencies could be raised through the introduction of additional standards and features for the virtual world. Further research in this area should take place, ultimately resulting in an implementation with Second Life or similar virtual worlds that enables virtual business to thrive.