

Learning Management Systems: Buy, Build, or Both?

In “Learning Management Systems: Buy, Build, or Both?” Cem Erdem compares two approaches lifelong learning programs use to obtain management software: building and buying. Erdem reveals the advantages and disadvantages of both, adding that both approaches are difficult, as each entails a set of sacrifices. However, Erdem believes that recent developments in technology will allow learning programs to fully reap the advantages of each approach.

Abstract

In my last article I profiled the differences between in-house and hosted solutions for learning management systems. In this article I will show the differences, the advantages, and disadvantages that come with building your own management system, compared to buying a system from a software vendor.

Cem (gem) Erdem,
CEO and President,
Augusoft, Inc.

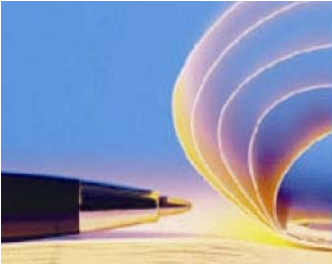
When you buy a learning management system, what you are really paying for is knowledge and experience. Each system is a product of years of development, involving the efforts of hundreds of people. This is the main advantage of a system that you buy, as opposed to build: you have size, power, and knowledge on your side.

Systems that are developed for the lifelong learning market at large will have the resources of several customers that have helped to fund the development. Therefore, buying a system is generally cheaper, since research and development costs are spread out amongst all customers of the software. In order to attract customers, a learning management system has to appeal to as wide of a base of customers as possible, in order to recoup investment in development. This means that systems that you buy from software vendors have a solid set of core functionality which covers and creates efficiencies in the most important areas of learning management.

In addition, since all software vendors will need to create a better product to attract new customers, you can expect such a product to grow and incorporate the newest technologies. You do not have to put forth the extra effort to guarantee your software stays maintained and up to date. Upgrades and buy fixes are generally included in any license that you buy to use a learning management system.

Unfortunately, the size of a learning management systems' customer base or development team can create a crippling set of disadvantages.

When you buy a system, you will have to fit your organization and business



practices into that application. When the software was first built, one direction was chosen at the expense of others. Once the software started on that path, there was no turning back. More often than not, with bought systems, you do not get a good fit, and since you have outlaid your resources towards buying the application, you will have to comply with the demands the software puts on you, instead of the other way around.

This is similar to the way people used to buy cars. In the 1920s you had a few models to choose from, and no choice of options. You could not customize the color, nor could you get a custom leather interior. Even five years ago, cars were not built to an individual consumer's demands. You had to go to the dealership and hope they had what you wanted on hand. What was on the lot was what you got.

Most software is similar: that which you end up buying often does not work perfectly for you. In fact, you often end up working for the software. When you do receive upgrades or maintenance for the software, there is no guarantee that it will include what you need from it. Many software companies will have a development model that doesn't respond to customer needs well, or have so many customers that they cannot respond to individual customer demands. In these situations, customers are often divorced from the development of the software. They cannot choose the details that will make their system a perfect fit.

When you buy a pre-made system, it doesn't give you a competitive advantage. Chances are, your competitor has the same functionality and capability that you have. It does not harness the full potential of technology. A learning management system, if designed well and used properly, can give your organization a very effective competitive advantage.

Additionally, with any software system that is not built specifically for your program, you face the threat of an unwelcome change in the software. What happens to your program if the software develops in a direction that you don't care about, or that is totally wrong for your program? Do you find a new system? Or do you just forego upgrades in the future?

As in a car purchase, when you buy a software product, you don't automatically get a share in its development. You won't always get a say in where the product goes, and when you do, you will have to compete with the demands of all other customers of the system. However, when you build a software system from scratch, you get absolute say over how the product is developed, so that it fits your organization perfectly.

Therefore, the biggest advantage of building your own software system is the flexibility you gain from dictating the creation and development of the system. You can tailor the system to fit the unique needs of your program.

That being said, there are several drawbacks to building your own system. The foremost drawback is the expense. If you have built a car, or known someone who has built a car, you know that (depending on the builder) the end product is beautiful, and a perfect fit for the driver. But building the car from the ground up takes a heavy toll in time and money. Parts have to be ordered from all over, and the car generally cannot be assembled until all the parts arrive.

In the same way, building your own software system is extremely expensive and can sap the resources of even a large organization. The tradeoff then is either to build a software product that does everything you want, but costs an incredible amount of money, or to scale back on building the software, and build a product that may not offer as many of the features you need, but will address the key issues you need to address, at a lower price. This defeats the purpose of building a software system: nobody spends their own time and resources building a Model T.

When you build your own management solution, you are responsible for the upkeep of your software. There is no team to concentrate solely on the maintenance and upgrade of your software. To upgrade or maintain your system saps even more of your valuable resources. The hobby car builder cannot get a maintenance plan from a dealership, nor can the program that builds its own software.

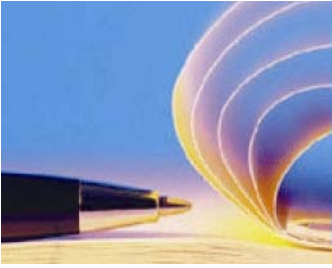
Perhaps the most damaging drawback of building your own learning management system is that you effectively isolate yourself from the rest of the lifelong learning market. When you create your own software solution, you rely solely on your own knowledge of the lifelong learning market. With only your own program as reference, you do not have access to the best practices in lifelong learning. You only have your own input, instead of learning from thousands of others from across the country. You are stuck reinventing the wheel – or the Model T, as it were.

Obviously, both buying a software system and building a software system have their pluses and minuses. Deciding how to balance your priorities, whether you need a more cost-effective or a more flexible solution can be difficult. However, there is middle ground in learning management solutions, one that incorporate the advantages of both built and bought systems.

The best of both worlds is a software management system that harnesses new trends in technology that allow them to be flexible enough to create customized, perfect-fit solutions for their customers, while still maintaining core functionalities that are defined by the shared knowledge of the learning market at large. This type of systems would consist of 80% core functionality (the engine behind the system.) such as profile management, robust security, infrastructure, the familiar shopping cart, standard reports, and online catalog creation functions. All of these are infrastructure related components that any company would need for successful business management.

You can buy these core components from software vendors that subscribe to the hybrid buy/build model, and then customize the 20% of features that will make the system a perfect fit for your organization. This is a new development model that became possible because of the Internet and the web. This represents the best of both worlds in an application:

1. Your money goes towards the continued development, maintenance and upgrades, freeing you from the responsibility of maintaining the infrastructure of the system. Additionally, you share the cost of the development with other programs.
2. You can guide the customization of your own management system, without having to take on the full burden of creating the customization.



3. You can share knowledge with other learning organizations, while still maintaining business processes that are unique to your program. This gives you a competitive edge.

In a way, both software vendors and car companies are updating the same time-tested trend: listening to their customers, and giving them what they want. With the advent of robust information technologies, it is getting much easier to give customers exactly what they want, by involving them in the decision. Now, with the development of the Internet, you can choose the extras on your car – CD player, leather seats, and a forest green paint job by the click of a button. You get to choose the extra 20% that makes the difference.

With enough time and money you can build your own car, but most people buy one because it makes life easier for them – the development, the maintenance, the support of the company all come included.

The same is true with learning management. Most lifelong learning programs could build their own system, but they know doing so would take up time and money which is better spent elsewhere. Also, buying an off-the-shelf product will not suit all your organization's unique needs. Many management systems will help you manage 80% your program, but few can give you the extra 20% that makes all the difference.