



ASP Solution versus Internal Solution
“Build” versus “Buy”
Part II (Strategic Review)

Overview

This is the second part of the “Build” versus “Buy” series of white papers.

Companies struggle with the idea of build versus buy all the time. Sometimes, the correct answer is abundantly obvious. Very few firms would choose to try and build their own phone systems, or perform their own electrical work. However when it comes to other types of systems, such as Commercial Real Estate Systems, some firms are less sure.

This white paper explores the strategic differences between going with an internally built system and an external ASP style solution.

Strategic Differences

Tested System versus Untested New Build:

Rockport has tens of thousands if not hundreds of thousands of person hours logged on the system. It has gone through a decade of testing and is maintained by a team of 15 people. It is used by best in class businesses which do not accept imperfect products. This battle testing has made the application rock solid.

The alternative is to try and stay with internal development and hope that this continued development will work. Staying internally is a major gamble as it means taking a little tested, unperfected product, and continuing to modify it significantly, while hoping that it works the first time.

Rockport on the other hand will be using 99%+ of its out of the box system, and then performing the simple custom configurations (screen and report design) which a client requests. Rockport has successfully done exactly this for many of other companies and has a track record of success.

Real Estate Domain Knowledge:

Rockport’s developers know both the technical as well as the real estate. This integrated knowledge tends to be much stronger than internal IT. No matter what the industry, someone who understands the end user and his/her daily concerns will develop a far more useable application than someone who does not have the same domain specific knowledge.

Dedication of Resources:

Internal IT typically is tasked with maintaining, improving and building several different applications for several different divisions. Rockport has one and only one focus – improving this application. This singular focus leads to faster and better results because of a lack of distractions, and a lack of fire drills from other departments. If internal IT is completely dedicated to the application then typically this single focus comes with significant costs which makes it far more costly than Rockport.

Time to Build versus to Buy:

No matter what the topic, and systems are no different- the time it takes to build is always significantly longer than buying the same product. Whether it is a house, a car, or a system it is always much faster and less thereby less risky to buy something than it is to try and build it. This is especially important to realize if time overruns would be damaging to the business enterprise. Nothing complex ever gets built quickly- especially with internal politics, distractions, and technical hurdles. If a system needs to be in place quickly and is nontrivial in nature, it is almost impossible to accomplish this with an internal build.

Mass Production versus Individualized Hand Production:

Assembly lines, Interchangeable parts, Economies of Scale, etc. all point to the same basic premise which has been proven time and time again- A company which dedicates its focus to one product will always build a better, cheaper and more sustainable product than a company that tries to do a one off development.

Long Term Sustainability:

Systems are never built and then walked away from. The financial industry (especially now) continues to change rapidly and systems need to react. Technologies, best practices and compliance also change. Much of this comes 'for free' for Rockport clients as Rockport evolves itself. If done internally every one of these evolutionary steps is paid for with internal work. If not, then the system stagnates. As a benchmark, Rockport puts ~\$2 million worth of new development into its standard system each year.

Intangibles:

Client support, testing and training all require significant time resources. Frequently on a system release these 'intangibles' can take as much or more time than the actual development itself. If the project is done internally all of these roles need to be filled by people who have cost associated with them. These costs may be actual costs (salaries of technical workers) or huge opportunity costs (if it pulls people off of other critical work – such as asset management or REO work).

Loss of System Expertise:

If an internal system is built by 1 person or a small team (the only way to do it with any cost consciousness involved), then the key person risk is huge.

If that person(s) leaves or is transferred to another division or project all their experience goes with them. There are many examples of this occurring with internal systems. Each time, the impact to the system ability to continue to evolve is devastating (the new people brought in have to relearn the system from scratch). This is why Rockport cross trains all of its employees such that it does not have the same key person risk. Additionally, by focusing on employee retention Rockport has an incredibly high retention rate.

Cost of System

Rockport is of the belief that our annual fees represent ~1% of the cost to build an equivalent system.

The cost of systems should be broken down into subcomponents. If cost is of relevance, internal should be asked at what price they can provide the following (all of which is required to deploy a Rockport type solution):

- System Technical Design Team (Business Users, Technical Design Team, Project Managers and other supporting roles)
- Actual System Development (Application Developers, Database Architects, Report Developers)
- Testing Team to assure quality of all deliverable (Application, Screens, Reports, Imports)
- Security Testing to ensure security of the new system
- Training Team to teach the new users how the system works
- Documentation Team to document the system features, and maintain it as the system evolves.
- Interfaces with other systems (also design, development, testing, etc)
- Infrastructure:
 - Network hardware and equipment sufficient to guarantee 99% uptime*
 - Storage space and servers for the housing of the application and data and documents*
 - Disaster Recovery protocols and offsite hot backup locations*

Please see Part I of this white paper series for a detailed review of the topic.

Detailed Functionality Examples

Most of the points below, despite sounding minor would take probably on average 6 person months to create **EACH**, all of these features exist in Rockport. However they rarely exist in internal systems:

Full System Audit Trail:

Rockport stores the user, time and value of every record changed in the system, maintains the entire history, and makes it accessible to the user upon demand. This is a big help in compliance and to understand why field's value is what it is.

Adhoc Reporting:

Rockport has incredibly advanced and user friendly adhoc reporting tool which is meant for the nontechnical user to be able to use. Ad hoc reporting is one of the most valuable features an enterprise system can have as it empowers the end user to do their job instead of always waiting on technical resources for the answer to their questions.

Prebuilt Reporting:

Extensive prebuilt reporting means that a client has to build less from scratch (e.g. the counterparty risk report which searches the entire system for a questionable counterparty and returns all instances of it with one click). Rockport has dozens of these critical reports which have taken thousands of hours (~20,000 hours). All of these would need to be built by internal system to be comparable in functionality.

Exportability of Reporting:

All reports are exportable to PDF, Excel, Word, CSV by the user.

Dashboards:

Rockport supports user dashboards.

Schedulable and Eailable Reports:

Reports can be scheduled and emailed from the system. This requires significant integration with mail servers, and permissioning which is difficult to replicate as an internal system without significantly adding to complexity.

Integrated Excel:

Rockport has spent years building a completely unique capacity to heavily integrate Excel with the Rockport application. This allows for the extension of the application into Excel for the functions which users would prefer to perform outside of a system. This feature alone is estimated to cost several million dollars.

Counterparty (legal entity) Records:

Rockport stores all counterparties to a loan (companies and individuals) as actual records in its database so that there is no duplicate entry of information and so that all information (e.g. past experiences with them) are readily available- for instance, if the counterparty went back on their word on the last workout or contracting job, it will change how you do business with them this time. Otherwise a user would never know unless they spoke with the loan officer who worked on the last deal with the specific counterparty.

Time Series:

All major sets of data which have time dimensions are entered into Rockport such that they can be stored on a time series basis (e.g. Appraised values change over time) and reported as such. This adds a significant complexity on the database architecture and therefore time to develop a comparable internal system.

Email Notifications of System Changes:

Rockport can send email notifications of changes to the system (e.g. Status Changes).

Advanced Portfolio Capacities:

Portfolios can be built either by use of filter criteria (e.g. all properties in a specific status, that are located in Michigan) or by selecting loans. Portfolios can also be made up of other portfolios. All these features are available on demand to the end user and are deployed with intuitive end user interfaces. These portfolios are then integrated into every report so that portfolio level reports can be run on any subset of data that a user prefers.

Chronologies:

The user can add read only notes to the different chronological records so that histories of that record can be maintained for both compliance concerns as well as a record of who said what when (lender liability, etc).

Geocoding of Properties:

Rockport geocodes properties which allows for mapping, as well as quality controls (for instance to make sure the same property isn't in the system twice).