Starting Software Company: Taking the Plunge

Paper #310 – HPWorld August, 2001
Robert Lund – President of Lund Performance Solutions
Andrew Herbert – Vice President of Lund Performance Solutions

Abstract

Robert Lund, the founder and President, and Andrew Herbert, the Vice President, of Lund Performance Solutions will, for the first time, tell how to start and run a successful software company. This talk is aimed primarily at those who wish to start their own companies and provide greater service to the IT community as well as reap more financial benefit. Unique pitfalls and opportunities will be described and discussed. A special feature of this presentation is how to utilize offshore contractors to compete in the fast track of software development. Anyone who is thinking of taking off the yoke of "employee" ought to attend this seminar. Existing software companies may benefit as well.

Three Fundamental, Non-Optionals to Succeed in Life

Proven, Not Just Spoken Integrity – *Ethos*
In all facets of life people are *always* asking the following question, *"Can I trust you?"* The question might be sub-conscious, but it is always asked. Our customers, and others whom we serve, must know that we will do what we say otherwise they will go elsewhere.

Genuine Compassion - *Pathos*
At the heart of service, in general, and customer service, in specific, people want to know that the people serving really care. Not surprisingly, many companies have adopted slogans like, "We really care," or "Customer Care is Our Main Product." All too often these phrases are just that, slogans. Humans must tangibly know that other humans genuinely care about them, apart from just gaining something. Companies that invest in people who have been raised with a strong respect and love for others will find that these people are a major asset to the company. When combined with integrity, true care for people will make your company (and life, for that matter!) nearly unstoppable.

Relevant Technology – *Logos*
Needless to say, what customers want are relevant, affordable, and easy to use products. A software company must stay on top of operating system changes and new platforms. Bugs and new features present a never-ending river of challenge. But in order to succeed we must be sure that our software and services stay current and keep up with newer technologies and customer wishes.

These three principles form the foundation of the best and most truly successful companies.

Software Company Pitfalls

Technology and the tools to develop it are moving nearly at the speed of light. This puts tremendous pressure on you to keep up a frenetic pace of doing more and better than your competitors. Additionally, the global presence of the web has blurred market boundaries. It has also allowed entrepreneurs in *every* country on the planet to develop products and deliver them effortlessly to customers in *any* country on the planet. This is both terrifying and exciting. Terrifying for those who have enjoyed near-monopolic access to markets due to geographic and cultural walls, and the accompanying lucrative lifestyle. The web has caused virtual e-cultures to emerge. Historically, mainframe and mid-range customers have purchased software and consulting from companies with a local presence. House calls were normal – and expected. But now, an Eastern European software engineer, who, by the way, can live like a king on $1,000.00 US, is able to develop a program, advertise it on the web, take a credit card order, download it with doc, FAQ lists, and email support, *very* quickly. This is exciting for such folks (not to mention the customers), but has demolished traditional means of software development and delivery. Over time, this will have *huge* economic impact, and equalization in some social sectors. Even now, some of our engineers in Eastern Europe are making many multiples more than their fellow countryman. Computer science professors in Eastern lands are being lured away to contract for Western companies. Wouldn't you if your monthly salary as a university professor was $150.00 US and you could work similar hours, with potentially less pressure and make $1,500.00 US? Not a difficult call.

Such is the nature of the software business. Nearly all other industries are bound by traditional constraints such as commodities-laden cost of goods, warehouse real estate and Fed-Ex delivery.

Other pitfalls include many of the normal perils of doing business.

Software Company Promises

While the pitfalls look ominous, the promises can be well worth making your way through them. Here are a few:

- Software is cheap to re-produce (but remember to give some good thought to the issue of licensing and tier-pricing – these issues are not as easy to address as they look!)

- Software can create an entrance for a plethora of other services such as consulting, support, customized enhancements and more software.

- Under the right circumstances, developing your own software can be the perfect "moon-lighting" venture. You still have the benefit of the steady paycheck at your day job, while performing all of the other tasks in the evening. Be sure to make plenty of time for family members and friends! Also, for integrity sake, be sure that your software company does not compete with your current employer.

- If you are able to find a niche, particularly if you invent a new algorithm, then you have a short window to enjoy your time in the sun. Combined with a timely marketing presence you may reap abundantly.

Four Things We've Done Wrong

Over the years we have learned some things about successfully running a software/service company. Unfortunately, a few have occurred the hard way. Thankfully, we have developed a value of learning from our mistakes and taking remedial action. Here are four things that we have had to learn about the hard way:

Being more impressed with our technology than our customers were

One danger of technologists is to become too infatuated with esoteric technology. "Tech-heads" may be impressed, but many administrators and managers are simply intimidated. We have learned that most IT professionals are generally overworked and have a difficult time keeping up with new technology. Consequently, they like things to be as simple and non-esoteric as possible. Self-explaining, ease of use, on-line documentation, and quick access to knowledgeable support staff, and relevant GUIs are all key phrases here.

Keeping less-than-optimal people too long

Sometimes a heavy dose of pathos causes management to hang onto people who are a wrong fit for the enterprise a bit too long. 'nuff said!

Maintaining too low of a salesperson to engineer ratio

I remember one time, long ago, when Vladimir Volokh said to me in his wonderful, Russian-accented English, "Bob, you've done this all wrong; you have 1 salesperson and 3 engineers; I have 25 salespersons and 1 engineer." I retorted, "Yea, but you have Eugene!" We have seen other companies, including ourselves, fall into the trap of not having enough sales and marketing support. Indeed, it is almost universal for a one-man-band to over-emphasize the technical and under-emphasize the propagation of that technology.

Taking too long to bring a product to market

It takes vision to conceive of a new product or service. But all too often a visionary person does not possess strengths in executing all the necessary details required to successfully take such products from the womb into maturity. Market windows do not stay open forever, and we've missed a few!

Four Things We've Done Right

Emphasizing excellent relationships between employees, along with a value of respecting their off-hours

One benefit than can be realized by startup and small companies is that they are able to create more of an extended family environment. But it must be intentional. Unless there are people who intentionally inject such into the business, it will not usually happen on its own. At our company, we have emphasized that people generally work to live, not live to work.

And unless they have enough time outside of work to pursue life in the fashion that fulfills them, they will not be productive at work.

Successfully utilizing offshore labor

A number of years ago, we decided to test the concept of using contract engineers in countries other than the U.S. This was largely due to a shortage of large dollars and engineers in our region. We had developed relationships through helping people in an Eastern European country. We discovered after giving a lot of resources to needful persons that it was better in the long run to give them jobs, while still helping those who truly could not work, etc. This has worked well for us, and well for them.

Choosing to prosper the old-fashioned way

A few years ago we visited a company that had just gone public. LPS was going to sign a significant contract with this company. They were housed in a high-rise with first-class, gold-plated everything, complete with *new* office partitions and office equipment. I reminisced to a time in our young company when a couple of us looked like the Beverly-Hillbillies as we drove back from Seattle with a most embarrassing load of desks, chairs and other office appurtenances. I'm pretty sure that there were no two desks or chairs that looked alike. I mentioned to the team, "Wow, wouldn't it be cool if our place could look like this high-rise company?" That company enjoyed a trip to the moon due to a huge influx of stock dollars. Sadly, in less than twelve months this company's stock essentially divided by 15 or so. About 250 people were jobless. The company somewhat stabilized and is now trying to grow at a more prudent pace. I suppose that one question would be does one create better software more quickly with brand new, $5,000-per-seat office furniture, versus a $200 garage sale special? To be sure, if you choose to succeed in the old-fashioned way, you may not be able to afford the best equipment, but you should have what you need to do the job. It is a fact that a few will be able to attract fast track, investment money, and they, due to unique offerings, market conditions, etc., will do very well. The stakes are higher, the hours are longer, the gray hair comes faster, but you might roll boxcars. So, which way will you go?

Creating an open and empowering environment for *every* employee

One of our greatest thrills has been to create a productive, but non-oppressive environment for each employee. This means that managers must empower, yet not control. This has worked best by starting off with employees who have excellent ethos and pathos to begin with. We also perform a personality profile on each new prospective employee. This tells us if we are truly fitting a round peg into a round hole.

A typical lifecycle of a software company

Phase I – The Dream

In this phase, there is ample vision. Sometimes the idea for a new product comes from a utility that the founder created to solve problems. Since this product is the founder's "baby," it goes without saying that he is proud of his baby. This "pride of parenthood" will do at least two things. First, it will cause the parent to give all for the growth and welfare of the new

baby. Second, it will often blind the parent to defects that the product possesses. Nevertheless, the mere idea of this new baby will cause the parent to dream big dreams and rightly so. This dreaming is what causes world-changing products to come to market.

Phase II – Concentrated development with sacrifice

This is the phase where the founders pour in an inordinate amount of resources, and with good reason. If there is an idea, opportunity, time, and finances, why wait? So the reasoning goes. But such projects invariably take more resources than one originally estimates. Time, energy, passion, and money are poured in often in an inordinate fashion. Everyone senses and sees the vision, and therefore is more willing to constrain their normal tendencies to be slothful and undisciplined. Pride and hope drive the developers on towards the finish line.

Phase III – Birth pains, birth and the birth announcement

It is in this stage where the product hits beta. Everyone is excited. The beta sites provide vital feedback. Documentation and advertising ought to be in full swing, but often are not. One of the frustrating dilemmas that are faced at this point is timing for advertising. Most printed advertising requires a number of months of advance booking. At this sensitive juncture, the founders have to make a decision. Do we book ad space three months ahead of time and hope that the product is tested, documented and deliverable at that time? Or do we wait until the product is perfect and ready to go, thereby losing an important three-month window in the marketplace. Always a tough call.

Also at this point, funds and nerves are thin. If the project has taken longer than projected, and software always does, then it is tempting especially for a self-funded entrepreneur to borrow money to keep going. Sometimes this works, and sometimes it doesn't, but in either event, it places a risk liability on the back of the operation. The pressure relieved has been exchanged for a longer-term, milder pressure. In times of plenty the debt obligation is almost negligible. When cash flow is lean, then the payments are oppressive, and if things don't improve the longevity of the operation is compromised.

One writer has pointed out that one root of the word *debt* is a cousin to the root of the word *death*. These are wise words for a small start-up. Debt can send you to the moon… or to the grave. It needs to be carefully deployed, if at all.

Another alternative to a private or bank loan is venture capital (VC). Often less oppressive than a loan, VC still causes the owner to give up much. For some reason, it is easier to spend VC money than borrowed funds. This may account for some of the many dot-com failures observed throughout 2000 and 2001. Once the money is gone, and if there isn't the hope of profit, the big wheel stops turning, at times rather abruptly.

Here are a few notes about marketing and advertising. A good idea-person (ad creator) is worth their weight in gold. At this stage, hopefully you have good ads, a super web site (in this day and age, there is no excuse for a mediocre site), a marketing campaign with a strong theme, and excellent follow-up and sales support. Without these you are doomed.

Phase IV – Slow, steady growth, boom or bust

Here, the owner(s) are faced with orders and growth. Software must be continually debugged, improved and promoted. The engineer(s) are weary and could use help. At this point, if not sooner, the thought of hiring employees enters into the thought process. Going from a small consortium of partners or a single proprietor to hiring employees is a much larger risk step than many realize. Whole new jurisdictions enter into the life of the company. Just the thought of worker's compensation, a potential wide-variety of employee-related litigation, training, bad attitudes (remember an employee is usually not an equity holder in many cases and often does not share the abandoned enthusiasm of the founder, sick days, and benefit plans, all send chills up the spine of any self-respecting employer.

But an ancient Proverb says, "Where there is no oxen, the stall is clean, but much increase comes from the strength of the ox." No employees, no hassle, but no growth or low growth are also synonymous.

The decision to expand beyond a mom-and-pop shop depends greatly on the ability to attract, develop, and keep quality people. It is interesting to note that Hewlett-Packard, at least in times past (as I haven't seen the statistics in recent years) was number one in this arena (Datapro surveys). Prudently acquiring good people, even if you have to spend a bit more, usually is worth it. Saving some salary money for a person who does not have exactly what you want will cost dearly in the long run, especially when compared to more capable people. Our development manager just recently said, partly joking, but mostly serious, that he felt that this paper could be summed up by saying, "The success of a software company depends two-thirds on hiring quality people and one-third on not losing them!"

Also at this point, the new company is starting to acquire customers. Ah, the all-important customer. Are they a purchase order-wielding, soul-less consumer who simply exists to beef-up your bottom-line, or a friend and ally for your mutual long-term success? How you answer this question weighs heavily on your true success. We have learned that a happy customer will literally bring more customers. An unhappy one will drive people away; the worse part of it is that you will not know about it, nor will you be able to defend yourself. Employees with healthy doses, and a balance, of ethos, pathos, and logos will attract customers more rapidly than a huge advertising budget in the long run.

Phase V – Expansion, innovation, market-share

At this point the founder who stays a sole proprietor with zero or very few employees can be very successful if he has exceptional integrity, customer care and technology innovation. If the products and services offered become popular, it will be hard to stay small and still deliver good service. Growth will bring the normal pains of raising a company. There are many good books about business success, in general. Nearly every book about business, customer service, etc., pays for itself in huge multiples. Here's an important thing to never forget: even a small contingent of loyal customers will carry you a long distance; be *really* good to them. In this stage you have more competition. You must innovate and expand wisely; you will have to run faster and smarter, or get eaten alive!

What about You?

Given the above homily, how well do you think your company would do if you decide to take the plunge, and transition from an employee or contractor to a proprietor? Some good questions to help you answer the above question:

- Do I have at least 5 months of personal and business expenses covered after I take the plunge?
- To what degree can folks live with or without my product(s)?
- What features distinguish my product(s) from other competing products?
- Is my *pathos* outstanding?
- How does my marketing/advertising compare with what's out there?
- Does my website just inform or grab people and pull them in?
- Do I have nerves of steel?
- Can I be a good techno-geek along with business manager, marketeer, negotiator, pathos-oozing customer support representative, conference speaker, *ad naseum*?
- Am I committed to staying current on all facets of my technical niche?
- How will I maintain a profitable edge over my competitors who have truckloads of VC to dump at will?
- Do I have full support from my family and friends? Do they understand the cost to them, due to the cost to me, and that they will probably have to kiss me goodbye for the next 9 to 18 months?
- Am I willing to "bridge" my vocational life for a time; i.e., work a normal job and write code into the wee hours of the morning?
- Can I handle the potential increase in wealth if I succeed? Do I have enough ethos to handle fame and fortune? (Gosh, when was the last time you heard that question in our land?)
- Do I have tested models for handling all areas of a software company? How will I respond to bug and enhancement requests? In-house accounting? My own "back of the envelope" marketing, or a professional?

Conclusion

In short, the software business has been extremely rewarding for many, but this industry is fraught with pitfalls. Many software and e-tech related businesses have shot up like rockets, but have flamed out shortly thereafter. Sometimes the best answer is to partner with a firm that has a good track record of support, products, marketing, and industry reputation. By grappling with the above issues and honestly answering the sample list of questions, you will have a better chance coming out a winner.