The SaveMe Software Company

"I need your help. The only condition is that you tell us what to do in thirty days, and then help us take the medicine in the next thirty. I need this done fast. I’ve got to fix my company."

So spoke the CEO of a small, rapidly growing software company focused on storage management software -- systems that help DP managers manage storage on different types of devices on large enterprise networks. Over the past several years, revenues had grown from nothing to about $3 million in the current quarter. The management team was driving for $5 million in the next quarter and $8 in the one after that.

While sales were climbing upward, the company was having trouble making new products on time. The company’s essential process --- sell to any customer in any industry and develop anything they want in order to get the sale – would not scale. It’s last two developments had not gone well. A new revision to the core product, scheduled as a 9-month effort, had taken 12 months and would not really be working for another 3 months. A second bolder effort, to both improve the underlying "platform" and create a new release of a complementary application had fared much worse. After 18 months of work by dozens of engineers, the entire project had ground to halt.

The CEO stated his case:

I know our product development is broken, but I do not know how to fix it. We have great people and we pay them as well as anyone. If they had given me the product they said they would last quarter, our revenues would be twice their size today. We are stuck in our tracks, though. We used to develop product in 9 months; then stretched to 12 months; and our last go around was over 18 months before I pulled the plug. That is why I called you guys in.

---

1 This case is totally fictitious. It was prepared for teaching purposes only by Marc Meyer and John Friar. November 2000.
My top engineers tell me the software has become so complicated that it is impossible to do even small things fast. The big things we need to do, well those never seem to get done. I sense that morale is low inside the company and that we are finally about to get some genuine competitors.

He went on to explain:

Everything was so beautiful for the first three years. Plenty of venture money. Strong software engineers. I landed a series of solid accounts, and hired some good sales staff to follow through. We started to grow. Went from a dozen people, to two dozen, and then all of a sudden, we had a hundred. Now we have 150.

When we were a couple dozen people, we were all on one floor, sharing telephones, coffee cups, having a few drinks, having fun and building a business. When we hit 50 people, we created real departments for engineering and sales, and put each one on a different floor of this building. Then, when we grew past 75, I had to get more space, so we grabbed a couple floors across the street in that building outside my window. I put all of engineering and the testing departments over there. Ever since, that parking lot has been our Berlin Wall. A sales or marketing person walks over there to provide some customer input into product development – well, it is as if they speak a foreign language. Engineering has not been over here except for formal meetings in 18 months.

This CEO rose from his desk, looked outside his window and reflected on the individuals on the other side of the parking lot.

When I call up the high priests across the parking lot to say I want to talk, I know they are peaking out the windows to see how upset I look. If I look hot, I know they'll hide. I cannot seem to find anyone. If I do not look mad, they'll talk to me. Then, it usually comes down to the fact that “you don’t understand code” and that we just need to give it more time.
Well maybe I never programmed much in my career, but I know that we cannot survive with an 18-month development cycle, or even a 12-month cycle for that matter. I cannot IPO this way. Sure we are selling what we made yesterday well, but we are going to need a lot more functionality darn soon. The high priests need to listen to our sales and market folks, or they are going to build things that customers won’t need, and miss the boat on things that customers need a lot.

You organize a meeting that day with the CEO, the COO, the VP of Sales, the VP of Marketing, and the VP of Engineering. Bombs are being thrown all over the place.

- The VP of Marketing declares that the company serves "a market segment of one," offering a broad horizontal application that everyone in a large organization needs. He declares "there is no difference marketing to a financial firm versus a telecom versus a government agency for our software. The problem is with engineering. We just need to kick their butts."

- The VP of Engineering, a very thoughtful guy, speaks next. He admits that Engineering has indeed tried to do too much and that the more problematic a project is, the less likely his own management team is willing to be up front about that with the rest of the company. He wants to scale down the scope and duration of projects. He also suggests that part of the problem is that there is no consistent, structured process for determining market requirements and passing them to engineering. Rather, the CEO and COO go on sales calls, close new sales contingent on "this little feature or that little feature" and leave us "holding the bag." This phenomenon has come to be known as the CEO doing a "swoop and poop" on the development organization.

- The VP of Product Management, a rather introspective fellow, has great market instincts but no systematic methods for confirming these insights. He has six staff working for him, and
they all sit together next to Sales.

- The VP of Services, perhaps the hardest working manager of the lot, speaks next. "We have pretty good user training, but our professional services are pretty thin. We answer telephones, help the sales guys put together demo's, and send out patches and sample code. However, if the customer needs an integration of our software with SAP, for example, they are going to have to do it themselves or get a systems integrator to do it. I just don't have the people. Unfortunately, all of our active users have had professional services performed on our standard product."

- The VP of Sales then speaks. "We should just push product as fast as we can. I don’t pay my guys to make the stuff work, just to sell it. That's your problem Mr. VP of Services. And buddy, your going to have lots of problems because I plan on selling 50% more product from quarter to quarter from here on out."

"Now you see my problem, right?" says the CEO. "We are shooting ourselves. I'll pay your fee, but I need a plan fast, and I need it to stick."
Case Questions:

1. How would you segment the target markets for SaveMe Software?

2. What is the difference between a tool, a product, and a solution for SaveMe Software? What should be the implications of that for product packaging and pricing?

3. Organizationally, how might you go about structuring and then coordinating groups focusing on tools, versus products, versus solutions in SaveMe Software?

Then, think about YOUR COMPANY:

1. How would you segment Your Company's target markets?

2. What is the difference between tools, products, and solutions for Your Company? How does that differ, if at all, for traditional market segments versus new, emerging ones?

3. How does Your Company organize the development of tools versus products versus solutions, and how well does it coordinate developments between them?

4. If you had to focus on one or two key areas to improve product and service development/delivery at Your Company, what would they be?
Supplemental Information

Products

The company has two products:

- The first product is called MStore, a repository that creates a massive database of files, size, access permissions, backup status, and so forth.

- The second product is MStore-DB, a storage management system focusing on database management systems and their files. Much of the growth in data storage is based on large Ecommerce and Ebusiness applications storing voluminous amounts of data in Oracle.

These two products were approached as separate projects. They do not share the same architecture or an underlying database schema for storing information. The GUI's are all different, too. So are the installation processes. None are based on or offer an API.

An API might be used, for example, by a systems integrator to hook up the storage management system with a user’s administration system to charge different departments for their storage use. Another application might be to better manage access to files at various organizational levels. There is no clear delineation between the platform that could be, and these two products. The VP of Engineering has stated that much could be shared. He also believes that a stream of value added applications, such as billing back departments for storage use, can also be defined and developed.

The first product (MStore) has had multiple annual releases. The second product (MStore-DB) is in prototype form only, even though twenty engineers have been working on it for a year. Most agree that the problem was that they were trying to greatly increase the scalability and security in the underlying engine, as well as trying to create an entirely new application, all in the same project. Most of them want to "start over" but feel that too much "face" will be lost if they ever admit it. Your product plan should seek to separate "platform" from "product," and reduce the riskiness of the company's current approach.
Markets

- The company’s best sales person has been the CEO. He has sold the company’s software to banks, brokerage houses, insurance companies, manufacturers, telecommunications firms, and Internet service providers. He and his management team like to say that they exist in a “target rich” environment. He walks in from every trip with lots of targets, many of which become development efforts. With greatest traction in financial services companies, managers believe that Internet Service Providers comprise another emerging segment.

- There is a difference between being sales driven and market driven. Market driven companies do not necessarily sell to every customer that “fogs up the mirror.” Sales driven companies do. This company has never said no to a prospective customer.

- For each market segment, there are logical strategic marketing partners as well as logical technology partners. The company does not have a clear definition of how segments, marketing partners, and technology partners fit together.

- Over the past two years, 80% of the company’s sales have been to banks and brokerage houses. These are large institutions.

- Customers have been using systems integration consulting firms to provide such services for the company’s software. The company itself has no formal 3rd party developer’s program, nor a programming interface for 3rd parties.

- The company has sold systems to about 300 companies. The software is actually used in about 50 of these customers. 100% of active user companies have procured professional services for custom programming or systems integration on top of the software sale. Some engineers believe that integration products (productized APIs) need to be created for 3rd parties.
Sales

• Europe produces about 50% of the total company’s sales, but does so with six salespersons as opposed to a dozen in the United States.

• Your team calls the European sales manager. You learn that he has hired his own professional services manager and two additional services staff. They are teamed up with salespersons on nearly every sales call. They then either perform the custom programming and integration themselves, or subcontract the work and actively manage it.

• The US sales teams gets paid twice as much for product sales as for service contracts. You learn that to most, discussions of professional services rarely come up in sales calls.

Organization

• The company has approximately 150 employees.

• It has a number of functional departments, including: Sales, Marketing, Product Management (customer requirements for product development and marketing programs), Engineering, QA, Product Maintenance, and Customer Support. There is no formalized Professional Services group.

• The company actually has a number of “go get ‘em” mid-level engineering managers who, with prior companies, demonstrated considerable project management skills. A review of personnel with the CEO showed that in fact there eight such individuals. The problem is that in the current system they really weren’t managing anything of great importance.

• Communications between these departments is strained. Being a good communicator has never been a promotion requirement to engineering manager.
Processes

• The standard operating procedure is that problems between functional groups tend to be resolved only through confrontation between VPs of the respective functions. They do not wait for meetings to do this, either.

• Product management staff do not tend to work elbow to elbow with engineering leads. When they do visit, the reception is often not very warm. "Back up your requirements," is the typical response.

• To surmount the “death march” mentality, management decided last year to send its middle managers to an advanced project management training course, offered by best selling management guru. Developed in the manufacturing and operations world, this process focuses on building slack or buffers into project goals and schedules. Some cynics have observed that its best for the company to avoid “death march” projects in the first place, and that smaller, more tightly focused initiatives are needed. In fact, they feel that the guru’s processes might give everyone just enough confidence to repeat past sins.

Finances

• Given current staffing and financial projections, the company breaks even on a cashflow basis at $8 million per quarter in sales. It hopes to get there in two quarters.

• There is enough venture money in the bank to last another 12 months at the current burn.