

## Führen oder bluten?

**W**enn Sie Geld investieren wollen, stehen Ihnen viele Möglichkeiten offen. Sie könnten das Geld auf ein Sparkonto legen, aber die Zinsen würden wahrscheinlich von der Inflationsrate aufgefressen werden. Sie können es in Staatsanleihen investieren. Aber auch damit verdienen Sie nicht sehr viel Geld; und wer weiß, ob diese Anleihen in den heutigen Zeiten noch sicher sind.

Sie könnten Ihr Geld auch in ein kleines Startup-Unternehmen investieren, beispielsweise indem Sie mehrere Tausend Euro im Austausch für einen kleinen Geschäftsanteil hergeben. Wenn das Unternehmen auf einer soliden Geschäftsidee basiert und diese effizient in Markterfolg umsetzen kann, könnten Sie potenziell *viel* Geld verdienen. Andererseits haben Sie keine Garantie, dass Sie zumindest Ihre ursprüngliche Investition jemals zurückbekommen. Dieses Konzept ist nicht neu. Sie lernen es bereits als Kind beim Spielen kennen. *Wenn Sie direkt durch die Mitte laufen, überraschen Sie möglicherweise alle anderen, und niemand rennt Ihnen nach.* Auch im Alltag begegnet Ihnen das Konzept tagtäglich. Sie treffen eine Risiko-Nutzen-Abwägung, wenn Sie sich zu einem Meeting verspätet haben und versuchen, den richtigen Anfahrtsweg herauszufinden. Wenn die Straßen frei sind, können Sie durch eine bestimmte

Abkürzung 15 Minuten sparen; doch wenn sie verstopft sind, dann war's das.

Die Risiko-Nutzen-Abwägung ist ein wesentlicher Aspekt der bewussten Entscheidungsfindung, wenn es darum geht, Technologien oder Geschäftsbereiche für Investitionen auszuwählen. Vor etwa zwei Jahrzehnten war das Risiko gering, in COBOL programmieren zu lernen. Natürlich gab es damals auch viel mehr COBOL-Programmierer, mit denen man konkurrieren musste. Deshalb war das Durchschnittsgehalt eines COBOL-Programmierers nicht berauschend. Man konnte zwar leicht Arbeit finden, aber die Arbeit war nicht besonders lukrativ. Geringes Risiko. Geringe Belohnung. Hätte man sich dagegen für eine neue Sprache wie etwa Java von Sun Microsystems entschieden, wäre es wahrscheinlich eine Zeit lang schwer gewesen, überhaupt eine Anstellung bei einem Unternehmen zu finden, in dem Java eingesetzt wurde. Wer wusste damals schon, dass überhaupt jemand *jemals* etwas mit Java tun würde?

Doch wenn Sie sich damals die Branche so genau angeschaut hätten, wie es Sun offensichtlich getan hatte, hätten Sie möglicherweise etwas Besonderes in Java gesehen. Vielleicht hätten Sie das starke Gefühl entwickelt, etwas Großes vor sich zu haben. Früh in Java zu investieren, hätte aus Ihnen eine Leitfigur in einem großen kommenden technologischen Trend gemacht.

Natürlich hätten Sie in diesem Fall mit Java recht gehabt. Und wenn Sie Ihre Karten richtig ausgespielt hätten, hätte sich Ihre persönliche Investition in Java vielfach ausgezahlt. Hohes Risiko. Hohe Belohnung.

Nun stellen Sie sich vor, Sie hätten etwa vor 15 Jahren eine Demonstration des neuen Betriebssystems BeOS von Be gesehen. Es war damals unglaublich. Es wurde von Grund auf konzipiert, mit mehreren Prozessoren zu arbeiten. Seine Multimedia-Fähigkeiten waren einfach erstaunlich. Die Plattform fand damals ein erhebliches Medienecho; und die Meinungsmacher schwärmten in Erwar-

tung eines neuen Mitbewerbers im Markt der Betriebssysteme. Natürlich bedeutete eine neue Plattform auch neue Arten der Programmierung, neue APIs und neue Konzepte für die Benutzerschnittstelle. Es gab viel zu lernen, aber es schien die Mühe wirklich wert zu sein. Sie hätten viel Zeit darin investieren können, der erste Anbieter etwa eines FTP-Clients oder eines Personal Information Managers für BeOs zu werden. Als Be eine Intel-kompatible Version seines Betriebssystems veröffentlichte, zirkulierten Gerüchte, Apple wollte das Unternehmen aufkaufen, um dessen Technologie als Basis für die nächste Generation des Macintosh-Betriebssystems zu verwenden.

Apple kaufte Be nicht. Und es stellte sich bald heraus, dass Be nicht mal eine Nische des Betriebssystemmarkts besetzen konnte. Das Produkt kam einfach nicht an. Vielen Entwicklern, die sich die Programmierung für die BeOS-Umgebung angeeignet hatten, wurde langsam und schmerzlich bewusst, dass sich ihre Investition langfristig niemals auszahlen würde. Schließlich wurde Be von Palm gekauft, und das Betriebssystem wurde eingestellt. BeOS war eine riskante, aber attraktive Technologie-Investition, die für die Entwickler, die sich dafür entschieden hatten, keine langfristigen Erträge brachte. Hohes Risiko. Keine Belohnung.

Bis jetzt habe ich über den Unterschied zwischen der Wahl von Technologien philosophiert, die sich an der Vorderfront der Entwicklung befinden und die fest etabliert sind. Sich für eine stabile Technologie zu entscheiden, die bereits weltweit in Unternehmen produktiv eingesetzt wird, ist sicherer, aber potenziell weniger lohnend als eine spritzige neue Technologie, die noch von niemandem angewendet wird. Doch was ist mit Technologien, die ihren Höhepunkt überschritten haben? Die nur darauf warten, dass die letzten Nägel in ihre Särge eingeschlagen werden?

Wer schlägt diese Nägel ein? Vielleicht denken Sie beispielsweise an die letzten wenigen RPG-Programmierer, die grauhaarig die Stunden bis zu ihrer Pensionierung zählen, während die neue Genera-

tion von Programmierern noch *nicht einmal* von RPG *gehört hat*. Heute lernen alle Java und .NET. Man kann sich leicht vorstellen, dass sich die Karrieren der letzten verbliebenen Statthalter einer überalterten und sterbenden Technologie in derselben Todesspirale drehen wie die Technologie selbst.

Aber die alten Systeme sterben nicht einfach. Sie werden ersetzt. Darüber hinaus werden in den meisten Fällen firmenspezifische Systeme stufenweise ersetzt. Und auf diesen Stufen müssen die alten Systeme mit den neuen kommunizieren. Jemand muss wissen, wie man die alten Systeme überreden kann, mit den neuen zu reden, und umgekehrt. Normalerweise wissen die jungen Grünschnäbel nicht, wie man die alten Systeme dazu bringt, zuzuhören. (Meisten *wollen* sie dies gar nicht wissen.) Umgekehrt wissen die verkrusteten alten Voreigentümer ebenfalls nicht, wie man die neuen Systeme dazu überredet, mit den alten geliebten Kreaturen zu sprechen.

Deshalb erkennt der kühl rechnende Technologie eine Marktlücke oder Rolle, die gefüllt werden muss: das *Technologie-Hospiz*. Alten Systemen zu helfen, komfortabel und in Würde zu sterben, ist eine Aufgabe, die nicht unterschätzt werden sollte. Und natürlich werden die meisten Menschen einfach das sinkende Schiff vorher verlassen, entweder indem sie in Rente gehen oder auf einen anderen technologischen Bereich ausweichen. Wenn Sie der Letzte sind, der ein immer noch geschäftskritisches System betreuen kann, können Sie recht anspruchsvolle Bedingungen stellen. Es ist riskant. Denn wenn die Technologie *wirklich* verschwunden ist, sind Sie Experte für etwas, das nicht mehr existiert. Doch wenn Sie sich schnell genug bewegen, können Sie sich die nächste aussterbende Generation überalterter Systeme (engl. *legacy systems*) ausgucken und von Neuem anfangen.

---

### Merksatz

Beide Enden der Übernahmekurve neuer Technologien könnten lukrative Geschäftsmöglichkeiten bieten.

---

Die Übernahmekurve neuer Technologien hat zwei Enden. Wie weit außen wollen Sie auf diesen Enden agieren?

## Umsetzung

1. Erstellen Sie für den heutigen Markt eine Liste der Technologien, die sich in der Anfangs-, Mittel- und Endphase ihrer Übernahme befinden. Stellen Sie die Verläufe von links nach rechts grafisch dar; links soll sich die Vorderfront befinden; rechts sollen die Technologien stehen, die kurz vor dem Aussterben stehen. Bemühen Sie sich, in jedem Teil des Spektrums so viele Technologien wie möglich zu finden. Seien Sie so feinkörnig oder detailgenau wie möglich, um die Verhältnisse der Kurven zueinander möglichst korrekt darzustellen.

Wenn Sie alle Technologien dargestellt haben, die Ihnen einfallen, markieren Sie diejenigen, in denen Sie Ihre Stärken sehen. Markieren Sie dann (vielleicht in einer anderen Farbe) die Technologien, mit denen Sie einige Erfahrungen haben, aber die Sie nicht wirklich beherrschen. Wo befinden sich Ihre meisten Markierungen auf der Übernahmekurve? Bilden sie einen Klumpen? Sind sie gleichmäßig über das ganze Diagramm verteilt? Gibt es Technologien an den äußeren Enden, für die Sie sich besonders interessieren?