Consulting

training.kalzumeus.com

How I got started consulting and the advice which changed my life.

[How I went from $100-an-hour programming to $X0,000-a-week consulting.](#)

Hiya guys! Patrick (patio11) here. You signed up for periodic emails from me about making and selling software.

My business is a motley collection of side projects. One of them: in my spare time, I run a high-end software marketing consultancy. It is modestly successful: I make my clients millions of dollars and charge mid five-figures a week.

Want the big secret to it? The big secret is that there is no big secret. Nobody ever shows up at your door and says "Welcome to the Illuminati. You can now charge $20,000 a week. Here's a list of clients." Assuming you have some valuable skill, like being able to program, turning it into a successful consultancy just requires excercising a bit of business accumen. Let's peak behind the curtain at some things which have worked for my business and those of my friends.

Clients Pay For Value, Not For Time

A few years ago, I was a much-put-upon grunt programmer at a large Japanese megacorp. I go home every Christmas to Chicago, so I was going to be in Chicago during December 2009.

I have an Internet buddy in Chicago named Thomas Ptacek. We met on Hacker News. He's the #1 poster by karma and I'm #2. Since we apparently share the same mental disease characterized by being totally unable to resist comment boxes, I decided to invite Thomas out to coffee. My agenda, such that it was, was to gossip about HN threads.

Thomas runs a very successful webapp security consultancy, Matasano. (Brief plug: they're hiring and if it weren't for this business thing I'd work there in a heartbeat: some of the smartest folks I know doing very, very interesting work which actually matters. If you can program they'll train you on the security stuff.)

Anyhow, after we got our coffee, Thomas invited me into their conference room. We

talked shop for three hours: Thomas and his VP wanted to hear what I'd do to market their products and services offering. I had been writing about how I marketed Bingo Card Creator for a while, and started applying some of the lessons learned to their content creation strategy.

(The actual contents of the conversion are not 100% germane to the story, but I [blogged a bit about it](#) and [Thomas posted his thoughts on HN](#). Long story short: programmers can do things which meaningfully affect marketing outcomes.)

At the end of the conversation, Thomas **said something which, no exaggeration, changed my life**.

> *Thomas: Some food for thought: If this hadn't been a coffee date, but rather a consulting engagement, I'd be writing you a check right now. Me: Three hours at $100 an hour or whatever an intermediate programmer is worth would only be $300. Why worry about that?*
>
> *Thomas: **I got at least $15,000 of value out of this conversation.***

You'll notice that I immediately thought the proposed transaction was time-for-money, but Thomas (the savvy business owner) saw the same conversation as an exchange of business-results-for-money. He correctly anticipated that Matasano would be able to take that advice and turn it into a multiple of $15,000. (They did, within two weeks, but that isn't my story to tell.)

This is, far and away, the most important lesson to learn as a consultant. People who are unsavvy about business, like me in 2009 or like most freelancers today, treat themselves like commodity providers of a well-understood service that is available in quantity and differentiated purely based on price. This is **stunningly not the case for programming**, due to how competitive the market for talent is right now, and it is even more acutely untrue for folks who *can* program but instead choose to offer the much-more-lucrative service "I solve business problems — occasionally a computer is involved."

So after this conversation, I stopped saying "I don't think I could do that" when companies asked me to work with them... and I also [stopped calling myself a programmer](#).

I could literally talk for several hours on properly pricing consulting services. Erm, strike that, I literally have. Ramit Sethi and I talked about it for a few hours ([here](#) and [here](#), with transcript). It was also the topic of a recent podcast episode with my cohost Keith Perhac and Brennan Dunn, again [available with transcript](#). **Read those.** People have told me that, in aggregate, they've made hundreds of thousands of dollars just by walking their rates up as a response to those interviews. (Brennan also writes a book about concrete strategies to do that. I bought a copy myself. It is [worth your time](#).)

I wish I could tell you that after speaking to Thomas I had a sudden burst of enlightenment on this topic, but that would be a lie. When I started consulting several months later, I went straight back to $100 an hour "to get my feet wet", but did learn one thing from the experience: I charged weekly rates.

Charging Weekly: It Makes Everything Automatically Better

What's the difference between $100 an hour and $4,000 a week? Aren't they mathematically equivalent? **No. Weekly billing strictly dominates hourly billing.**

- Weekly billing means you never waste time itemizing minute by minute invoices ("37 minutes: call with Bob about the new login page").
- Weekly billing means you have uninterrupted schedulable consulting availability in weekly blocks, and non-billable overhead like prospecting or contract negotiations happens between the blocks (when you weren't billable anyhow) rather than during the workday (when, as an hourly freelancer, you are in principle supposed to be billing).
- Weekly billing makes it easy to align units of work to quantifiable business goals, where those goals dwarf the rate charged.

Weekly billing also does wonderful things for pricing negotiations… because **you'll stop having them**. When I write a proposal for an engagement, I typically write a list of things we can do and my estimate for how many will fit into 1, 2, or 3 weeks. If clients don't have 3 weeks in the budget, we can compromise on scope rather than compromising on my rate.

If you quote hourly rates rather than weekly rates, that encourages clients to see you as expensive and encourages them to take a whack at your hourly just to see if it

sticks. Think of anything priced per hour. $100 an hour is more than that costs, right? So $100 per hour, even though it is not a market rate for e.g. intermediate Ruby on Rails programmers, suddenly sounds expensive. Your decisionmaker at the client probably does not make $100 an hour, and they know that. So they might say "Well, the economy is not great right now, we really can't do more than $90." That isn't objectively true, the negotiator just wants to get a $10 win... and yet it costs you 10% of your income.

When you're charging weekly rates, the conversation goes something more like this: "So you don't have $12,000 in the budget for 3 weeks? OK. What is the budget? $10,000? Alright, what do you want us to cut?" You can then give the negotiator something to hang his cost-cutting hat on while still preserving your ability to charge your full rate in this engagement and all future engagements. (Word to the wise: no client, anywhere, likes giving up discounts after they've been given them. I have ridiculously successful client relationships where I, stupidly, cut them a discount years ago and I'm *still* paying for that decision.)

So I bet you're wondering how I got from $4,000 a week to $X0,000 a week. Sadly, no silver bullet: I just climbed a ladder of project importance, gradually (over ~20 engagements and ~2 years) accumulating wins and using each win to get me to the next rung of the ladder. Let's look at how.

Getting Clients: The Importance Of Social Proof

One of my first consulting clients was Fog Creek Software. They got in touch with me after having read my blog and forum comments for a while. We've done the odd gig together over the years, beginning with a relaunch of their marketing site for FogBugz (writeup here) and continuing with a very fun project that will be written up on their blog Very Soon Now (TM).

People occasionally tell me that my strategy is not replicable because I'm, air quotes, "Internet famous." Back when Fog Creek got in touch with me, my blog had a few hundred readers on a very good day. I never demonstrated (and don't possess) untouchable genius unmatched by anyone before or since... I simply talked openly about things that worked.

I **always ask to follow a successful consulting engagement with a case study**. My pitch is "This is a mutual win: you get a bit more exposure and I get a

feather in my cap, for landing the next client." Case studies of successful projects with some of my higher profile consulting clients (like e.g. Fog Creek) helped me to get other desirable consulting clients. **Very few clients turn down free publicity**, particularly if you offer to do all the work in arranging it.

If you can't get a public case study (do all the work for them and just ask for their approval — this makes it very easy to say "Yes, go for it."), there are intermediate options. One is to ask for them to just OK a one or two sentence testimonial about working together. **Write this testimonial for them** and ask if they want to make any changes to it.

Here's a bad testimonial: "Patrick is smart. We enjoy working with him. — A Client" That testimonial does not resolve the #1 issue in your prospect's head: will this engagement make the business more than it costs?

Here's a much more persuasive testimonial:

> *Patrick's advice on starting a drip campaign for WPEngine was an epic win for us — it **permanently moved the needle on signups** after just a week of work. And it's easy to measure and therefore to improve.*
>
> *— Jason Cohen, Founder & CEO, WPEngine*

"Permanently moving the needle on signups" is one of the most persuasive things you could possibly offer a SaaS business. (It sure sounds better than "He wrote 8 emails. They were good!")

That testimonial is in Jason's own words, but I hinted in the direction of what would be useful for him to say. (The main change he did was striking a number that I had suggested.) Notice that this testimonial, even with that edit, remains **specific, focused on a business result, and highly credible**. In a discussion with a new prospect, their big worry is "Can Patrick do something valuable for me?" and "If I have him do it, will it generate a wildly positive result for the business?" Jason's testimonial assuages both worries.

If you can't get either a writeup or a public testimonial, at least ask for a private reference. I have a few from companies which I can't publicly mention as clients. They sometimes help sway people who are on-the-fence about hiring me,

particularly companies which are at the upper end of places where I could reasonably work. (I'm being oblique here. Sorry, nature of the business. I can reasonably work for Fog Creek / WPEngine, and in fact have. I couldn't reasonably be entrusted with strategic level projects at, say, Google. There exist a few order-of-magnitude jumps in between those two. Periodically, I try making one of those order-of-magnitude jumps. When I successfully manage one, I will — naturally — ask to do a public case study.)

I occasionally joke that every time I get a new case study, my weekly rate gets another zero, but that is directionally accurate. My first rate was $4,000 a week. It is now several multiples of that, all justified by "A client very similar to you paid my prevailing rate and had $THIS_FABULOUS_RESULT. If I could do that for you, what would it be worth?", with rates moving up every time I was comfortable doing so. (Both in terms of "My pipeline is sufficiently stuffed such that losing engagements over price shock is not a problem" and "Even at 25% more than what I charge right now, I'm still confident customers will have a successful outcome from working with me.")

Scaling A Consulting Business

I don't do consulting full-time because I enjoy running my own businesses more. That said, there are a couple of ways to take a consultancy and scale it. They're fascinating, since you can layer so many different business models onto the base money-for-time offering.

If you do well by your clients, you'll soon have too much work to handle. You've got two options at this point: raising your rates and turning away work in excess of your capacity to deliver, or raising your capacity, typically by hiring people. (Working more hours has serious scaling challenges... another reason to do weekly rates, by the way, as many folks working hourly succumb to the temptation to bill "just another hour" and end up with miserable work/life balance.)

Anyhow, you can either hire other people on a per-engagement basis (a freelancer managing freelancers, like my good buddy Keith Perhac) or you can have full-time employees who you charge out to customers, like Brennan Dunn's Rails consultancy. We talked in-depth about these models in our podcast recently.

The big surprise to me, when I investigated this as an option, was that you have to

be very careful to maintain your margins. I once had the bright idea, back when I was charging $5k a week, to bring in someone at $4k a week and pay them $80 a hour. $800 a week for doing nothing, right?

That's a catastrophically bad idea. In addition to having to cover your overhead, when you're sitting between a client and your employees' paychecks, you are absorbing significant risk of non-payment or delayed payment. Your payroll check needs to clear on Friday regardless of whether the customer has paid their invoice yet. There are occasionally *terrifying* cash flow swings in the consulting business... heck, that is practically the definition of the business.

Ooh, let's do story time. Earlier this year I got married. Weddings are expensive. You don't get any significant price break for buying two of them at once, which I did because my wife and I are from different countries and we wanted to include everybody. I recommend getting married, assuming you've found the right person, because how else are you going to have conversations like "You should pay $500 for a single balloon" and find yourself saying "That sounds perfectly reasonable, but only if we could pop the balloon immediately on receiving it."

One particular portion of the wedding cost $30k. I anticipated that $30k of receivables (work that had been done but hadn't been paid for yet) would turn into $30k of wire transfers prior to the wedding hall needing the money. They didn't. There was *nothing unusual at all* with this state of affairs: collecting invoices is like trying to dance the samba while simultaneously juggling, and I missed a ball or two. I ended up putting that on a credit card for a few weeks while waiting for my business to sort things out.

If you have two employees, you can easily have *a wedding worth of cash go out the door every two weeks*. Accordingly, you need to be better at cash flow management than I was. One way to do it is by charging your employees out at a rate substantially higher than you pay them, and then using the difference to build a cash-flow cushion. (Brennan recommends having $30k around *per employee* prior to scaling to your next hire. That strikes me as a great start.)

Concrete example: If you charge someone out at $4k a week, you should reasonably think of paying them on the order of $60 an hour (if they're freelancers).

Things get even more difficult if you hire full-time. You have to worry about

maintaining utilization of your team — i.e. always having work for them ready to go, because they get paid whether or not you've got them scheduled. In general, you should shoot for about 70% utilization, or them working 3 weeks a month. Let's say you hire an intermediate Rails developer at market rates: $8k per month. This costs you about $12k after you pay for taxes/health care/etc etc. (Employee benefits: welcome to being a business, please enjoy your stay.) You'll need to hire them out at in excess of $6k a week to accomplish that safely, at 70% utilization.

The economic returns to running a consultancy come largely from "leverage": being able to bill out your employees, as opposed to walking up your own rates as the principal. (Many principals find that their take-home goes down after hiring employees, at least for a while, while they iron out the kinks of cash flow and pipeline management.) After you're at 3 employees, even at a fairly healthy rate for yourself, you're probably making half your money on your billings and half on the margin between what your employees cost and what you successfully bill them for. This scales right on up, with the additional wrinkle that as one has more employees one spends more time on non-billable overhead (prospecting, HR, business administration, yadda yadda) and less on what you got into the business to do.

A necessary collorary to this: the principals of a technical consultancy do **very** well for themselves. I don't know if that is a secret but it certainly isn't well appreciated: nobody says Occupy Boutique Rails Consultancies, but the principals of them do end up in the 1%.

Hybridizing Consultancies With Product Businesses

Over the years I've seen a few people run product businesses concurrently with consultancies. (And I do it myself.) Some of the models are very interesting, but they're not obvious.

Probably the most common one is using the consulting revenues to underwrite product development. For example, 37signals and Fog Creek both started as web development consultancies. There are quite a few products produced along the same model these days: the principals hire themselves out at generous rates, only take modest salaries, use the difference to hire programmers at market rates, and have the programmers build the product they wish to sell. This has a lot to recommend it over e.g. bootstrapping the product from nothing (everyone gets a paycheck every two weeks) or e.g. taking investment (it is hard to close $20k in angel investment

but easy to close $20k in consulting contracts, and if the product takes off, you can stop consulting but you can't conveniently forget ever having been invested in). Consulting also helps develop generic business skills (which are very applicable to product businesses) and helps expose you to the problems of companies which pay money for solutions to problems, which makes customer development for your products much easier. (You can even share sales channels.)

The recent [course on lifecycle emails](#) which I made was a variant of this: it is a productized consulting offering. Basically, I took an engagement that I've delivered five times now and said "How could I refactor this engagement such that I can deliver much of the value for near-zero marginal time investment?" The answer was collecting my expertise and experience in a package that customers could use to self-direct themselves through implementing the advice. It seems to have worked out really well for many customers: instead of paying me $X0,000 for a week they can pay $500 to hear much of the same advice. It worked out very, very well for my business, too: I got approximately 1~2 weeks of consulting revenue without needing 1~2 weeks of on-site availability, and I now know that it will work if I want to repeat the experiment.

Another popular offering for consulting firms is to offer training workshops. For example, if you are an expert in a particular field, many companies are willing to pay for you to train up internal experts at their company. You can either convince one client to pay for a 1-, 2-, or 3-day event, or sell tickets to one which you put on for yourself. Typical rates I see for internal training range from $5k to $15k *per day* depending on what is taught and to whom. For public training, price points in the $500 to $1,500 a ticket range are common. Why is that a hybrid product offering as opposed to being straight consulting with a different charging model? After you have the curriculum/courseware for the engagement mapped out, you can then scale delivery of it almost arbitrarily: the cost of a trainer to deliver a day-long workshop in e.g. optimizing Postgres performance is much cheaper than someone who can actually optimize Postgres performance at an arbitrary company, and they're (relatively) common. You could take your single expert's knowledge and then package it up in several dozen workshops delivered in-person or online, with delivery performed by folks hired for the purpose. Amy Hoy and Thomas Fuchs used to run in-person and then online Javascript performance training workshops. It worked out very well but the rest of their businesses basically ate their availability. Sometimes billing $25,000 for a day just isn't worth the opportunity costs.

#firstworldproblems

Many consulting companies shut down their availability after successfully getting a product business going. That isn't a law of nature (Pivotal Labs' consulting arm survived Pivotal Tracker, for example), but after you've figured out scaling for a product business (highly non-trivial), they often scale so well that continuing with the consulting would be economically irrational.

Is This Interesting To You?

This is a bit off the beaten path for my writing, but a lot of you asked for it over email, and I seem to get strong responses when I post about it publicly. If you want to hear more like this, hit Reply and tell me what, specifically, you'd like to hear about. I tried to keep this fairly generic but if you want to hear a deep-dive into e.g. how to sell consulting clients on A/B testing or just want to hear what is in my bag-of-tricks I'm happy to oblige.

I also look forward to doing some more writing about software in the near future. If you've got a particular topic you want covered, drop me an email. I read them all and respond to most. (n.b. A belated Happy Halloween. Sorry for the few months of dead air — six weeks of international travel plus the busy season for my business coincided, and that cut down a bit on my available time to write these.)

Until next time.

Regards,

Patrick McKenzie

P.S. Quick plug: Brennan Dunn has a [podcast on consulting](). They're at six episodes now and they're all fantastic and, moreover, actionable, like on how you can use [content marketing as a lead generation tool](). I'd definitely suggest listening to them at the gym or whatever.

Contact us

Kalzumeus Software Takejima-cho 29 Ogaki City, Gifu Prefecture Japan, 503-0878
patrick@kalzumeus.com
www.kalzumeus.com

Feel Free To Share This

Anybody interested in seeing these emails on a regular basis should go to my site and give me their email address. If you'd like to share this link around, though, go right ahead! Copyright by Patrick McKenzie 2012, yadda yadda yadda, I trust you not to abuse it.

training.kalzumeus.com