

Extracted from:

Land the Tech Job You Love

This PDF file contains pages extracted from Land the Tech Job You Love, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printer versions; the content is otherwise identical.

Copyright © 2009 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The
Pragmatic
Bookshelf

PRAGMATIC LIFE

LAND THE TECH JOB YOU LOVE



ANDY LESTER



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic courses, workshops, and other products can help you and your team create better software and have more fun. For more information, as well as the latest Pragmatic titles, please visit us at

<http://www.pragprog.com>

Copyright © 2009 Andy Lester.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Printed in the United States of America.

ISBN-10: 1-934356-26-3

ISBN-13: 978-1-934356-26-5

Printed on acid-free paper.

P1.0 printing, May 2009

Version: 2009-6-18

OK to start using profanity. Not everyone you meet with may feel the same way. Stay professional.

8.4 A Walk Through the Interview

There are as many different interviews as there are companies that hire people. Your interview might be one-on-one, as part of a panel, over the phone, or at a coffee shop. Certain principles apply no matter what. Let's walk through the interview with these universal principles in mind.

Arriving at the Interview

Arrive at the interview at least ten minutes early, pen in hand. Chances are good that you'll fill out an application or other paperwork before you meet the hiring manager. Allow time for this paperwork before the scheduled start time.

More important, being late for an interview is just about an unforgivable sin. The interviewer's time is valuable, and arriving late is an insult to the person who's there on time. You're saying "My time is more valuable than yours." Arrive early to make sure this doesn't happen.

Of course, "showing up" means "showing up prepared, both mentally and in what you bring with you," as discussed in Section 7.7, *Prepare What You're Going to Bring with You*, on page 151.

The Receptionist

As you walk in, get a feel for your surroundings. Is there expensive art on the walls in a well-appointed lobby? Is it a startup in a cheap office storefront with beanbag chairs on the floor?

Find the receptionist. If there's not a dedicated person performing the role of a receptionist, the first person you see will effectively be your receptionist. Note that I said the first *person*, not the first *woman*. If you bypass a group of men to speak to a woman, you'll be remembered but not positively.

You must be unfailingly pleasant and polite to this person, because there are no second chances at a first impression.

Chances are the receptionist will have feedback about you to pass on to whomever is involved with hiring you.

After every interview, after the candidate has left the building, I immediately ask the company receptionist for her thoughts, as well as opinions from anyone else he may have met.

Initial Introductions

After a while in the lobby waiting patiently, it's time to meet the interviewer. This is where the real work begins.

When you first meet the interviewer, introduce yourself and shake hands if offered. There may be a bit of small talk, and you should find out what you're going to be doing at the interview and who you'll be meeting with.

Interviewer: *Hi, Steve? I'm Todd Trainer, and I'm one of the recruiting specialists in HR. I trust you found us OK?*

You: *Hi, Todd, I'm glad to meet you. Thanks for taking the time to meet with me today. The directions on the website were great, and I checked Google Maps as backup. I take it I'll be meeting with Mr. Weston a little later?*

Interviewer: *Yes, the way we like to do things is that you and I will be getting some paperwork done, and I'll explain a little about Conglomco. Then I'll take you to meet with some of Bob's staff, and at 11 you'll be meeting with Bob in his office.*

Note that everything is fine, no matter what. Do not give into the temptation to gripe to say, "Yeah, the directions were fine; I'm just glad I got here through all the traffic!" Nobody wants a griper, even in tiny doses.

As you're being told about the plans for the interview, don't be afraid to write them down. You're likely to forget something, and it's better to be able to refer to "Mr. Trainer"⁴ instead of "the man who met me out in the lobby, from HR I think?" Keeping notes is a good sign, because it says you're actually interested in what's happening and doing a good job at it.

4. Always refer to each person as Mr. or Ms. Lastname unless the person specifically says to call him or her by another name.

On the other hand, writing everything down can come across as antisocial or more concerned with the notes than with the conversation. You want to take notes, not court testimony. Also, it's polite to ask whether anyone objects to your taking notes.

Opening the Interview

The interviewer will probably open with a little background about the company, explain what they do, and maybe discuss the industry if it's new to you. He'll talk about the department's role and responsibilities in the company. Listen. Listen very carefully, because this is what you must address. Make regular notes so that you don't forget what it is you wanted to discuss or ask about and so that you don't spend all your brain cells remembering what you wanted to say, instead of using them to listen to what is being said.

The interviewer will probably give you a chance to discuss yourself, your history, and your skills in an introductory way. He may say "How's the job search going?" or "You've got a lot of different experience here," or even leave it wide open with "So, tell me about yourself." This is where you lay out a brief history of yourself, highlighting the most important skills you offer, as *they relate to the company*.

Interviewer: *...so with the offshore competition, management is watching costs like crazy, but the merger still leaves us almost doubling the number of users we have to support. But enough about us, tell me about you.*

You: *Well, after I graduated from Northwestern with a degree in computer science and a minor in business, I spent a few years working various system administration jobs, getting my feet wet, and getting exposed to as many different operating environments as I could. Those are the bottom three jobs on my résumé. The last four years I've been working for Parsleytronic, working on integrating Windows and Solaris environments. Integrating the sales and engineering sides of Parsleytronic, with only a minor increase in the IT budget, sounds a lot like what you're going through with your merger. How has the merging of the companies' IT departments been going?*

Note how you're answering the question and showing your past experience with something similar to the job you're applying for,

if not exactly the same. You've recapped the high points of your résumé, even though they're on the résumé already. Maybe he never noticed that minor in business before. Maybe your résumé wasn't clear enough that you were integrating Windows and Solaris. The recap helps make sure your points are made.

Somehow you want to tie yourself to the needs of the company and the hiring manager, even if only to just show interest in what's going on. Maybe your background isn't as good a fit as in the previous example, but you're not out of luck.

You: *...and then I've been at ShmooCo since 2004 working on engineering applications for brake subsystems. I've certainly had my share of challenges in my past, but your merging of the departments sounds like a big one. I'd like to hear more about how that's going.*

I'm sure there's a skeptic reading this saying "Boy, that would make you sound like a total suck-up." It would if it were insincere, but it won't be. You're going to be very interested on your first day of paid employment, right? So, be interested right now at your first meeting with your potential new boss. If you're *not* interested in the hiring manager's company, his problems, or his concerns, then you're in the wrong interview.

We'll delve further into how to open the interview in Section 9.4, *Tell Me About Yourself*, on page 181.

8.5 Sell Yourself by Telling Stories

Getting the job offer requires selling yourself. It's the active part of "get a job offer," and it's what can make the difference between getting the job you want and losing out to someone who took the time to do the right kind of selling.

Selling yourself isn't evil, it isn't creepy, and you don't have to feel like a used-car salesman. It's just putting forth the information about yourself that the interviewer may not know. If the interviewer never asks about your hot skills in automated testing, project estimation, or administering web server clusters, it's your responsibility to bring them up.

Bring up skills that show you in your best light.

Don't assume that because a given skill is listed on your résumé that the interviewer will delve into it in the interview.

The interviewer has probably not memorized the entire thing, no matter how interested she is in you. In fact, the things that stick out in her mind are the items that she sees as important, and you may be able to raise awareness of the other good aspects of your background.

The core of selling yourself is telling stories, preferably specific, quantifiable stories that don't fit on your résumé. Instead of answering questions as simple, matter-of-fact yes-no exchanges, like so:

Interviewer: *Do you know Perl's testing framework?*

Weak answer: *Sure, we use it all the time.*

turn the question into a jumping point for you to tell a story about the topic:

Strong answer: *Yes, very well. I started out using Test::More when I was writing my own CPAN modules, just testing the module I'd written. From there I wrote my own testing add-on, Test::Wango. Now, we use the Test::More framework and the TAP protocol for testing about one third of our internal applications, and the plan is to be at 100 percent by the end of the year.*

The latter answer shows the depth of your knowledge and opens up a world of discussion. If nothing else, it gives you the opportunity to teach the interviewer something he didn't know himself. Maybe he'd never heard of TAP or never realized that you could test more than modules with Test::More.

He wouldn't have asked, "Do you know the TAP protocol?" because he didn't even know it existed. You may have just shown the interviewer that you are even better at a given area than he or his current staff are, and that's a great position to be in.

The latter answer also gives you a chance to show your enthusiasm for the topic at hand. A good manager loves to see passion and excitement for a topic, and that first bland answer just doesn't expose any. Selling yourself is about answering more

Don't Play the RTFM Game with Your Résumé

RTFM means “Read the Freaking Manual,” often said to someone online who has asked a question where an answer to that question happens to exist elsewhere. It’s an attitude that will crush your chances at getting a good job, as well as getting along with others.

Tech people often think “If someone asks me about what has already been written, it’s an imposition or even an insult.” They’re not afraid to let that attitude show. The thing is, people will see you as a jerk, which is the last thing you want.

Maybe an interviewer will ask “You know Oracle, right?” If your response is “Yeah, it’s right there on my résumé,” you’re saying “RTFM.” RTFM assumes that the person is stupid or lazy. Your interviewer is neither.

Maybe he just missed the Oracle part. Maybe he barely skimmed your résumé. Maybe you forgot to put it in there yourself. Putting him on the defensive by pointing out that he missed it serves no one.

You were probably asked “You know Oracle, right?” as an open-ended entry into a conversation about your Oracle skills. Assume this is the case. Come back with “Yes, I’ve worked with Oracle back at Yoyodyne. We had quite a data warehouse going, although we had some startup problems. Boy, I could tell some horror stories, but I learned a lot. Are you looking to start working with Oracle?”

It’s not an annoyance; it’s an opportunity to sell yourself.

than the simple question with the information that your interviewer really wants to know, even if he doesn’t know it at the time he asks.

Ask Questions

Not only does your boss-to-be need to know whether you’re a fit for the company, but it’s just as important that you get to know the company and your boss-to-be so that you can decide whether the job is a fit for you. You should ask questions of

your interviewer throughout the interview process, when it feels natural in the course of conversation. The questions you ask should also help show your interest in the job.

- “You’ve said you’re moving to all Perl from PHP for your web applications. Why is that, and how is that conversion going?”
- “It seemed pretty crazy out in the programmer office when you showed me around. Is this common? Is there a big project wrapping up?”
- “I notice some books on Oracle on your shelves, but you’re a PostgreSQL shop. Are you looking to convert?”

Of course, you’ll have a list of questions that you wrote before the interview, as discussed in Section 7.5, *Prepare Your Questions to Ask*, on page 145. If you can work them in conversationally, then mark them off your list as you go. Don’t be embarrassed about having a list.

Do not bring up salary or benefits in an interview. That is for the job offer stage and will be discussed at that point. If the interviewer brings up salary or benefits, then follow his lead, but don’t probe yourself. Leave discussion of these crucial matters for the offer stage. Your job is to get the offer, not discuss what an offer might contain.



I can’t emphasize enough how important it is to leave money out of your interview process. I know it can be aggravating and your patience may be thin and you really want to know how much you’re going to make, but don’t give in. Don’t try to say, “I don’t want to waste anyone’s time here, so how much is this job paying?” It won’t work. At the very least, you’ll sound like the money is more important than the work.

Handle Technical Questions

The technical questions will probably be the easiest part of the interview, because they’re the most easily answered. Either you know the answer or you don’t. When asked about a technical issue, you usually don’t want to answer with a simple “yes” or “no,” because that’s probably not what the interviewer is looking for. You also want to take the time to try to sell yourself a bit.

Interviewer: *Do you know any Ruby?*

Weak answer: *Yeah, some.*

Strong answer: *Some. Back at Peekax Research, we had a little project to decide on a new web framework. I was on the team looking into Rails and got pretty well immersed into some of the dark corners of Ruby. We went with J2EE, but I'd do more Ruby work in a heartbeat.*

You can also see how more detailed answers help eliminate any confusion. The simple answer without evidence is open to misinterpretation. Chances are that what you consider knowing Ruby “some” will not be what the interviewer thinks. Even if the question is about a skill that isn’t core to your job, which is likely the case for such a simple question, you don’t want to have the problem in the future of misconceptions about your knowledge: “What do you mean you don’t know how yield works? You said you knew Ruby!”

As you tell the story about the work, take responsibility for what you did. Don’t be vague and mushy about the work that you did and the roles you played. You are selling yourself specifically, not the place that you worked for.

Interviewer: *What work have you done in Ruby?*

Weak answer: *At my current job, we’ve developed four basic internal CRUD applications in Ruby on Rails, and we even published a package of math functions to RubyForge.*

Strong answer: *I was on a team of three developers who created four CRUD applications in Rails. We’d been using .NET, but I convinced my boss to let us give Ruby a try. I also went through some hoops to get management to let us release one of our math libraries as a gem on RubyForge. It’s called Arithmegoo, and it’s had a few hundred downloads so far.*

Note the strength of taking responsibility, of saying “I did this.” Even as part of a team, you are not an amorphous blob of workers. There are actions that you have taken.

Note also the difference between giving a vague overview and telling a story with details. Details make the story come alive. Details give authority to your words. Details engage the listener.

Be sure to listen to and pay attention to the interviewer. She may not be looking for a long answer. If your tales of how many gems you've published on RubyForge are making the interviewer get that faraway look as she loses interest, cut them short. Of course, this is also true of the entire interview process.

Sometimes the answer to the question is a “no,” and you need to basically say that. You never want to pretend you know things that you don't, but you don't have to just say, “No, I don't know about that.” Here are three responses that are better than “no,” in order of preference.

1. Discuss something you've done similar. “I haven't used LDAP, but back in 2006 at Yoyodyne, I set up and administered Active Directory for a 2,500-person company.” Make sure you don't tack on something like “... and LDAP's pretty much the same as Active Directory,” with the implication “If you can do LDAP, you can do Active Directory.” Maybe the interviewer sees LDAP as vastly inferior to Active Directory. Just provide the facts, not the value judgments.
2. Show that you're at least familiar with the name. “No, I haven't. Are you doing some sort of enterprise-wide directory integration?” You're showing that you have some understanding of how it's used, and you're getting more information. It may turn out that you have a different, similar experience and can turn this into answer 1.
3. Ask what it is and how it's used. “I'm sorry, no, I haven't even heard the term. What is LDAP, and how are you using it?” You'll show interest in learning more, and may find out that it's similar to something you've done before and can upgrade your answer to answer 2.

Please don't use the cliché “No, I don't, but I'm a quick learner!” It's good to try to turn a negative into a positive, but “I'm a quick learner” says nothing.⁵ Use one of these three answers.



How to Use Your Portfolio

If you've created a portfolio of work, as discussed in Section 7.4, *Prepare a Relevant Portfolio*, on page 140, use it throughout the

5. See Appendix A, on page 248, for more of these clunkers.

Don't Panic!

Don't think of each question as a question that you're doomed for not knowing, to be flung from the bridge for not knowing the airspeed velocity of an unladen swallow. I once asked a candidate, out of the blue, if he knew anything about LDAP. I thought he was going to have a heart attack as he stammered out his "Uh, uh, no, but, uh, I can learn pretty quick!"

Chances are if you've been called in for an interview, you've got the minimal skill set that they're looking for.

interview but carefully. It's a tool to help explain and illustrate but not the focus. You and your skills are the focus.

Remember that *you are going to leave the portfolio behind* at the interview. If she wants to mark up or make notes on your portfolio, that's fine, because it's hers to keep. It's well worth it to be able to have this exchange:

Interviewer: *Now, I see here [pointing with pen] that you're...oh, wait, can I write on this?*

You: *Absolutely, I brought it for you to keep.*

The best way to use your portfolio is as an illustrative tool to help show details about projects and technologies you've worked with.

Interviewer: *Have you had to maintain combined Windows and Linux networks?*

You: *Absolutely. Most of the time it's been working on existing networks, helping get them in to shape, checking for security holes, and so on. Once I had the luxury of designing a homogeneous network [flipping to "Network diagrams" in your portfolio], and I took the lessons from the previous years' work and designed this [showing the diagram to the interviewer].*

As you show your portfolio to the interviewer, keep a running checklist, on paper or in your head, of which areas you've had to show. Chances are you won't have an opportunity to show off everything in the portfolio, so you'll want to know what else you

definitely want to draw attention to. If it has to wait until the end of the interview, that's OK. You can work them into your closing questions and comments.

You: *A couple items we didn't discuss in my portfolio that might be of interest are this database abstraction framework I created and some of the feedback forms I got from a training class I taught. You mentioned training problems briefly, and I thought that might interest you. I brought this copy of the portfolio to leave here.*

Wrapping up is covered in more detail in a moment.

8.6 Sell Yourself by Doing the Job

The best way to make an impression in an interview is to actually do the job that needs to be done and show that you can do it well. It's not always possible, but when you can, the payoff is huge.

Phil Morrison tells of an interview where he clearly demonstrated his expertise and landed the job.

The Whiteboard Solution

by Phil Morrison, system administrator, Milton, Florida

I'd been approached by a recruiter for a position, and the recruiter came to the interview with me. There were six people in the interview, asking me very specific questions on how I would handle merging another company's Active Directory. It was obvious that they were about to undergo such a task and did not know where to begin. I had just recently been involved in a massive AD migration and asked them whether they would like for me to draw out a possible design on the whiteboard.

I drew out how I would set up the AD along with child domains, site connectors, and so on. This brought more questions, and I answered and added more detail to the drawing, explaining why this was the best practice, positioning them for future expansion.

They asked whether I had any experience in delivering software via GPO to the desktops. I smiled and outlined the three tiers of software. I explained the concept of "thirty minutes fix or reload," the motto at my last job. I showed how they could build images that could rebuild a desktop along with all the software, detailed out roaming profiles and redirected document shares so that

The Pragmatic Bookshelf

The Pragmatic Bookshelf features books written by developers for developers. The titles continue the well-known Pragmatic Programmer style and continue to garner awards and rave reviews. As development gets more and more difficult, the Pragmatic Programmers will be there with more titles and products to help you stay on top of your game.

Visit Us Online

Land the Tech Job You Love's Home Page

<http://pragprog.com/titles/algh>

Source code from this book, errata, and other resources. Come give us feedback, too!

Register for Updates

<http://pragprog.com/updates>

Be notified when updates and new books become available.

Join the Community

<http://pragprog.com/community>

Read our weblogs, join our online discussions, participate in our mailing list, interact with our wiki, and benefit from the experience of other Pragmatic Programmers.

New and Noteworthy

<http://pragprog.com/news>

Check out the latest pragmatic developments, new titles and other offerings.

Buy the Book

If you liked this eBook, perhaps you'd like to have a paper copy of the book. It's available for purchase at our store: pragprog.com/titles/algh.

Contact Us

Online Orders:	www.pragprog.com/catalog
Customer Service:	support@pragprog.com
Non-English Versions:	translations@pragprog.com
Pragmatic Teaching:	academic@pragprog.com
Author Proposals:	proposals@pragprog.com
Contact us:	1-800-699-PROG (+1 919 847 3884)