Transform how you hire software professionals

Agile Hiring

artima

Sean Landis

Agile Hiring

Agile Hiring

Sean Landis

artima

Artima Press
Walnut Creek, California

Agile Hiring
First Edition

Sean Landis, a software architect, has retooled hiring practices at three companies, leading to significant improvements in the quality and quantity of new hires.

Artima Press is an imprint of Artima, Inc.
P.O. Box 305, Walnut Creek, California 94597

PrePrint™ Edition first published 2010
First edition published 2011
Build date of this impression March 1, 2011
Produced in the United States of America

Cartoon drawing of Sean Landis by Chet Haase.
Cover photo of agile skier by Sean Landis.

This book is dedicated to my four lovely daughters,
Katie, Amie, Debbie, and Christi,
who supported me throughout this effort.

Overview

Contents

Foreword

I've been the Technology Senior Vice-President (CTO) at Overstock.com for six years. During that time my top priority has become hiring, or as I now call it, talent acquisition and management. From the smallest tech teams to the behemoths (Apple, Google, Microsoft), at the end of the day, the innovation and implementation of technology only happens with the right people power. If you hire the wrong individuals, or can't attract the talent you need, your company will fall short of its potential.

I'm assuming if you've picked up this book, hiring has become a priority to you and/or your organization. This is really the first critical step. Unfortunately it is the easiest step. My experience has taught me that just because I decided hiring was an organizational priority, it was not something we could solve just by throwing more person-hours at the problem. Clearly you need to put in the hours towards hiring efforts, but this is wasted if you don't have the right skills, strategy, and process in place.

Along my path of improving talent acquisition, I met up with a very good partner in crime who has a greater passion than I for improving our talent acquisition process. When Sean Landis joined Overstock.com, we already had a solid hiring plan and process in place or I doubt we would have been able to attract him to the company. As I began working with Sean on the hiring front he had incredible insights and new ways to look at improving what we were doing. Sean was one of the drivers as we pushed everyone to continue to focus and drive our talent acquisition. Let's face it: reviewing resumes is not that sexy, and we'd all rather be writing code.

The advice found in this book has been particularly valuable at Overstock.com. By re-engineering how we do phone interviews, we have improved our hiring efficiency while making an excellent first impression; candidates want to work for us. We have gotten much better at discerning good from bad resumes, which has avoided much wasted time. We have also improved the quality of our on-site interviews, nearly eliminating bad hires.

I look at this book in two ways. First, this is an excellent guide to put you on a solid path to better hiring. Second, even if you have a mature hiring process and are having success it is a great reference to reflect on what you might look to improve.

At Overstock.com we have built an incredible technology department. We've been able to bring in some great talent. The team is such that we are all able to learn from each other and collaborate to build great solutions. This has been one of the keys to our company's success. I feel incredibly lucky to work in such a great environment, and I give the majority of the credit to what we've been able to accomplish in improving our talent acquisition.

Sam Peterson—SVP Technology, Merchandising
Salt Lake City, Utah
October 3, 2010

Acknowledgments

Thanks to my friends and mentors from the early 1990s at a large software company. My manager, Ned Plasson and I, worked out several of the ideas I describe in this book. Also thanks to Rick Plautz for trusting us with hiring and Matthew Smith who gave an Aussie perspective.

Thanks to Robert Cooper, Heather Mardis and Scott Swarts, who partnered with me to hire developers to work on state-of-the-art software in a startup environment. It was there I learned how to hire outside my skill area when I was asked to hire a technical writer. Our software, by all accounts, was devilishly difficult to document and required a top-notch writer. We found one in Ruth Stento who taught me much about writing. Our developers were some of the smartest people I've ever met.

Later, in a corporate research environment, our leaders, Ken Zdunek and Mark Gannon, had the courage to break away from the standard hiring process because it was not serving our needs. Thank you for trusting me with my role in hiring; I learned about hiring the best researchers and how to perform the entire hiring process for a large company. Thanks to my manager, Venu Vasudevan, who always had great ideas.

Thanks to Michael Ogg and my colleagues at Valaran, a startup, where we had very informal hiring processes. This was another complex product line that required great developers.

Great thanks to my managers at Overstock.com, Ian Robertson, Sam Peterson, and Peter Maughan. It was here that I codified my experience and put it into practice to help build the best hiring system I am aware of.

Special thanks to Alistair Cockburn and the Salt Lake Agile Roundtable group. Alistair's writings, presentations, training, and conversations have provided much of the *Agile* in *Agile Hiring*.

Thanks to all the companies I've interviewed with, good and bad, that gave me the opportunity to learn from the other side of the hiring equation.

There are many people who have supported this effort with advice, ideas, and reviews. Thanks to Mark Griffin, Neil Hartner, Kevin Steffensen, Allen Loew, Cam Roe, Chris Mason, Mike Finger, Lauren Campbell, Lisa Crispin, Brendan McNichols, and Chris Maki.

Finally, thanks to my four wonderful daughters, Katie, Amie, Debbie, and Christi. Throughout this process, they have been a source of encouragement and support. They mean everything to me.

Introduction

Hiring software professionals is hard. This book will help you do it better.

I know from experience how difficult it can be build software teams through hiring. In this book I describe principles and techniques I developed in over twenty years of real-world experience hiring in the software industry. These principles and techniques can help you build a great hiring system that will enable you to hire the best software professionals.

Who should read this book

This book is an indispensable resource for everyone involved in the hiring process. That includes managers, human resources (HR) personnel, and most of all, the front line hiring folks. If you review résumés, conduct phone or on-site interviews, or if you are responsible for making offers to candidates, this book is especially valuable to you.

If you are responsible for putting together or improving a hiring system, you should read this book. You will learn what all the pieces are and how they fit together to form a cohesive whole.

An entirely different reader is the software professional who is seeking a job. Although I write to those who are hiring, the job seeker will find this book an great resource for understanding what employers look for. You will also learn what to look for in a great company. Companies that hire well tend to be great employers too.

Why I wrote the book

I wrote this book because few existing books focus on hiring software professionals. I'm not sure what is special about this field, but in my experience, the accepted hiring wisdom is not very effective when applied to software professionals. Thus much in this book is likely to be new to anyone familiar only with the traditional hiring wisdom.

The main premise of this book is that very few companies hire well. The skill level of the technical staff is not where it should be. There are too many slackers, poisonous personalities, and people unqualified for the work they do. Quality suffers; the code base deteriorates; and the bad design decisions of the past have made the products too difficult to improve. Unfortunately, bad hiring takes its toll long after the offer was accepted. At that point it's too late. This book will help you dig out of a bad hiring rut; through great hiring, you will be able to transform your organization.

A second premise is that individuals with deep domain knowledge are in the best position to perform great hiring. A non-expert in your organization is likely performing many of the tasks that I will suggest should be performed by software professionals. My contention is that hiring is one of the most important things an organization does, and therefore it is worth investing your valuable resources toward the act of excellent hiring. One of the key failures I have observed is that companies have the wrong people in the wrong places in the process of hiring. This book will suggest ways to improve that in your company.

Another premise of this book is that companies often focus on the wrong candidates. I have observed this failure in many companies. I recommend hiring experienced, full-time employees that you intend to retain for the long term. This book will help you do that in your company.

What has worked for me is a principled approach and techniques with feedback loops that encourage self-improvement and learning. The principles provide a solid foundation upon which the team can work safely, while individuals, through feedback and reflection, can develop skills and intuitions that support discernment in hiring. The techniques provide your group with the ability to confidently discern the candidates you want to hire.

How the book is organized

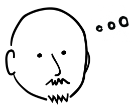
This book focuses first on the foundations that underpin an entire approach to hiring. These values and principles cut across and motivate the techniques that follow. Next I discuss the basics of defining the target of the hiring process; if you don't know what candidates you want, you will probably make bad hiring decisions. The goal is to know who you want, be able to recognize them, and then hire them.

The next three chapters focus on technique. First comes résumé reviewing, an area that is surprisingly fruitful and usually the place organizations fall down hardest. Phone interviews are discussed next, another area where skill makes a big difference. Then on-site interviews are covered. The treatment here is a bit more abstract because this is so broad a topic. I discuss my preferences to give a feel to how on-site interviews can go. By this point in the book, the reader should have the tools to know how to customize what I present for their needs.

> **A Note on the Examples**
> Most of the examples in this book are expressed in technologies I am most familiar with. Therefore you will see references to Java, Hibernate, SQL, and the like. There is nothing technology-specific about the principles and techniques in this book; they are applicable whether you are hiring Java developers, C# developers, or Objective-C developers, *etc.*

The book concludes with closing the deal: getting the desired candidate hired. Finally, I leave you with a few parting thoughts on the main themes of this book.

The principles and techniques in this book hang together and should be used holistically, but they are not fragile. If you understand the foundations and have worked the techniques, the experience you gain will lead you to profitable and safe customizations.



My cartoon face occasionally calls out useful nuggets of information. This is valuable when you are skimming the book for ideas and mental cues.

What’s so *agile* about hiring?

I debated whether to use the title of *Agile Hiring*. “Agile” has become a buzzword and I don’t want to appear to be riding that wave. But as the book took shape, I realized just how much Agile ideas informed my approach.

The starting point for agile software development is a position statement called the *Agile Manifesto*[1]:

> We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
>
> **Individuals and interactions** over processes and tools
> **Working software** over comprehensive documentation
> **Customer collaboration** over contract negotiation
> **Responding to change** over following a plan
>
> That is, while there is value in the items on the right, we value the items on the left more.

My approach to hiring is all about the people. Process is the backbone, but people are the flesh, passion, and smarts. Doing what works is more important than dotting every i and crossing every t. Planning is important but plans are unreliable. Every interview is different, so adapting to the situation is essential.

The manifesto is just the starting point. Agile development recommends other ideas such as reaching a goal incrementally and iteratively, building a team's expertise broadly and deeply, and reflecting to improve execution. Those familiar with agile development will see its fingerprints throughout this book.

What's not covered

A high-quality hiring system is like a living organism that spans upward and outward within and across organizations. An attempt to cover all facets of hiring—even if limited to software professionals—would result in a much larger and less wieldy publication.

I do not discuss how to source candidates or advertise positions. I do not cover how to hire new college graduates. The argument against hiring out of college is similar to that described in *Money Ball* against recruiting high school baseball players.[2] These players have a huge amount of development

[1]"Manifesto for Agile Software Development" [Sig]
[2]Lewis, *Moneyball: The Art of Winning an Unfair Game* [Lew04]

Why a ski racer?

I chose a ski racer for the cover because of the similarities between ski racing and agile hiring. The ski racer reviews the course before the race and develops a plan of attack. She knows that once she kicks out of the start house, her plan could change. Every race is different: the location of the gates, pitch of the slope, snow conditions, weather—all make for a unique experience. One surprising turn can slow her down and cause her to revise her plan. Instead of a steady, measured attack, now she must make up time, she must go all-out. But the principles and techniques of skiing remain the same. She must adapt those principles and techniques each time she runs a course. After the race, she reflects on her performance. She reviews video, and she trains and trains and trains. She's passionate about being the best.

When I am hiring, I review candidates. For the good ones, I develop a plan on how to interview them, but I know that I will have to adapt, in real time, to what the interview throws at me. Interviews aren't scripted because every one is different. Yet there are principles and techniques that I can always rely on. I must adapt them for every candidate. After every interview, I reflect with the other interviewers on what went well, what didn't, and what we could try in the future. We are passionate about hiring because it is one of the most important things we do.

in front of them and very few ever make the major leagues. There is simply too much risk to make this approach fruitful. "Rookies" place a tremendous load on an organization. In baseball, teams have sophisticated minor league systems to develop raw talent. Some software organizations are structured to safely and efficiently develop junior software developers. If you aren't sure about yours, then you probably don't work in one. Prefer seasoned professionals over *fresh-outs*.

I don't discuss hiring contractors. If your organization makes significant use of contractors or outsourcing, either software development is not a core value of the company, or the company's software strategy is misguided. The techniques herein can be used to evaluate individual contractors and outsource teams, but in an organization that depends on external resources to develop its core software assets, I wouldn't spend much effort trying to hire well. That effort is better spent looking for gainful employment at a company that values software development as a core competence.

EBook features

The eBook is not simply a printable version of the paper version of the book. While the content is the same as in the paper version, the eBook has been carefully prepared for reading on a computer screen.

The first thing to notice is that most references within the book are hyperlinked. If you select a reference to a chapter, figure, or glossary entry, your browser should take you immediately to the selected item so that you do not have to flip around to find it.

Additionally, at the bottom of each page are a number of navigation links. The "Cover," "Overview," and "Contents" links take you to major portions of the book. The "Glossary" and "Index" links take you to reference parts of the book. Finally, the "Discuss" link takes you to an online forum where you discuss questions with other readers, the author, and the larger software hiring manager community. If you find a typo, or something you think could be explained better, please click on the "Suggest" link, which will take you to an online Web application with which you can give the author feedback.

Although the same pages appear in the eBook as the printed book, blank pages are removed and the remaining pages renumbered. The pages are numbered differently so that it is easier for you to determine PDF page numbers when printing only a portion of the eBook. The pages in the eBook are, therefore, numbered exactly as your PDF reader will number them.

Content overview

- Chapter 1, "Foundations," introduces the basic concepts that are used throughout the rest of the book.
- Chapter 2, "Define the Target," helps you establish the requirements and other criteria against which you will measure candidates. If you don't know what you want, any candidate will do.
- Chapter 3, "Reviewing Résumés," provides you with what you need to become expert at analyzing résumés. There is a lot more information in a résumé than you could imagine.

- Chapter 4, “Phone Interviews,” teaches you how to master the most challenging aspect of hiring: the phone interview. Good phone interviews help you filter out unsuitable candidates that would otherwise waste your time and money in a costly on-site interview.

- Chapter 5, “On-site Interviews,” explains the basics and nuances of face-to-face interviewing, including many things you probably won’t read anywhere else.

- Chapter 6, “Closing the Deal,” arms you with ideas for successfully completing the interview process with an accepted offer. I will debunk some long-held beliefs.

- Chapter 7, “Parting Thoughts,” summarized what you have learned, and encourages you to continue the process of becoming great at hiring.

- Appendix A, “Behaviors,” describes valuable behaviors to look for in an interview candidate.

- Appendix B, “Hiring Principles,” is a functional list of principles that I feel are foundational to great hiring.

- Appendix C, “Résumé Review Checklist,” provides a reference for those new to reviewing resumes. Use this list until you begin to go beyond its limitations.

Agile Hiring

Chapter 1

Foundations

This chapter presents the foundations upon which the rest of the book is built. The first thing to understand is why you should invest effort improving your hiring skills. Next, I will explain how hiring principles create a sustaining fabric of consistency across an organization.

I present a simple hiring process and name some hiring roles, so that I can describe the hiring activities in a consistent fashion. I will cover a few "perspective building" topics such as cost management, values, behavioral traits, and a fun little metaphor, that underpin my approach to hiring. Finally, I will show how tracking candidates, and your process, provide support for continuous improvement.

1.1 Hiring as a core competence

A key premise of this book is that hiring is one of the most important things a software organization does. Every good hire accelerates your organization; every poor hire is a drag on your organization.

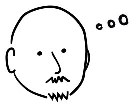
People underrate the negative effect of poor hiring. Conventional wisdom says that a good team can absorb a weak member and still perform well. But recent research shows that this is not true; one bad apple *can* ruin the whole barrel.[1]

If you are only able to hire one person, you'd better make it count. If you are hiring a bunch, you can bring down an entire company with bad hiring—or totally renew a weak organization with great hiring.

---

[1] Felps, Mitchell, and Byington, "How, when, and why bad apples spoil the barrel: Negative group members and dysfunctional groups." [Fel06]

Have you ever tried to hire into an organization that is less than stellar? You probably found it difficult to hire top talent. One possible reason is that the mature developer has a sixth sense in detecting mediocrity. Experienced professionals can size up an environment and detect good or bad smells.



Every hire adjusts the future of your organization. So hire well.

Strive to hire the best

Every new employee puts a drag on the organization when they join. The best developers will get up to speed quickly and flip the drag into a pull. They don't require nearly as much mentoring, training, or handholding as less experienced developers.

The best developers have a range of experience different from those of the team members they join. The diversity they bring will expand the capabilities of the team. Often the best developers have interesting hobbies or interests that will enrich the social environment they work in.

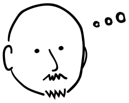
The best developers will require some mentoring, but before long, they will be mentoring others. The best developers have leadership skills, whether it is pure people leadership or some other form of technical leadership.

The best developers understand the importance of filling roles and respecting boundaries. They have the maturity to follow when appropriate, and the courage to lead when necessary.

The best developers learn quickly and are able to adapt to new situations while maintaining high productivity. They have a can-do attitude that is tempered by a clear sense of reality. They can evaluate the forces in a situation and use good judgment in making sometimes-difficult decisions.

The best developers are strong communicators and highly collaborative. Software development today demands that people work together and communicate intentions, expectations, and concerns. It is no longer sufficient to just be a *propeller head*.

The best developers will be creating the leadership supply your company will rely on in the future. They will bring a network you can tap into for information and future hires. The best developers are interested in long-term employment and continuous improvement.



You want to hire the best developers.

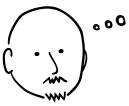
Prefer full-time employees

If your company views software as a core competence, then full-time employees are preferable. If your organization is planning on its software (or itself) being around for any length of time, you probably should prefer full-time hires to contractors or outsourcing.

Every person who works on your code carries around critical knowledge in his or her head that isn't written down anywhere, or if it is, it's probably out of date or isn't being read. The longer that person works with your software and systems, the more value you have locked in their head.

As your software matures, the developers who matured with it are the ones who will be able to fix it, refactor it, keep it running smoothly, and replace it when necessary. When a new hire joins the team, the veterans will be able to help convert the newcomer into a seasoned player.

When a developer leaves your company, you lose all that valuable information. Sure, you may be able to do a last ditch "information transfer" but mind-meld technology blew up with the planet Vulcan so you won't likely get good results.



Your best bet is to hire full-time employees and do everything reasonable to keep them as long as possible.

1.2 Promoting consistency with hiring principles

When many people are participating in the hiring process, it is difficult to maintain consistency in hiring. An organization should do everything it can to keep participants on the same page.

One fundamental way to promote consistency is to develop and communicate a set of hiring principles. When people are unsure how to handle a situation, they can turn to the principles for guidance.

Every organization should develop its own hiring principles, but here is a set that is an excellent start:

Always be respectful of the candidate. How a company treats its candidates is an indicator of what the work environment is going to be like. How you treat a candidate is a reflection on you; it could be awkward to later work with the mistreated candidate. One day that candidate may be your boss! By showing respect, you put your employer and yourself in the best light.

Always be on time. Being on time is a reflection of the company culture and is another way to respect the candidate. Being late will make the candidate feel unimportant and will trigger doubt about why someone is so busy they can't even be on time for something as important as an interview. Sure, the interview is important to the candidate, but it is also one of the most important things you do for your company. If you are late, apologize with an explanation and an offer to reschedule if the length of the interview is compromised. Empathize with the candidate; this interview is very important to them.

Be prepared. Interviewers must be prepared to conduct a high-quality interview. The interviewer should know the job descriptions. She should have reviewed the résumé, other candidate information, and all interviewer comments, and have developed a personal interview plan. Preparation significantly improves the chances of a good hire, avoids wasted time, and leaves a positive impression with the candidate.

Value the candidate's behavioral qualifications. Behavioral traits are as important as technical ability. The scope of projects today demands strong communication and collaboration skills; the days of the lone geek in the dark office are long gone. Obnoxious tech superstars need not be tolerated; mousey, shoe-staring introverts have no place in a collaborative environment. Hire for quality behaviors that align with your organizational values. This will pay off richly in the long run.

Always consider and value leadership potential. Every employee should be a leader. There are many leadership opportunities beyond running a team or organization. Even with an eye toward leadership, few will rise up to take the reins of a team or group of teams. Hire those who demonstrate leadership to build up your future leadership supply. The faster your organization is growing, the more important this becomes.

Can too many cooks spoil the broth?

Will too many people aspiring to leadership create conflict and lead to attrition? The fear is that too many people vying for a small number of leadership positions will leave your best people disaffected. So how much leadership supply do you need?

There are several important factors. The first is the rate at which your organization generates new leadership positions through growth (if new positions are being generated through attrition, your problem isn't hiring, it's retention). The second factor is the size and shape of your potential leadership pool. You ought to always know who is aspiring for leadership and where they are in their development.

Thirdly, you will always need more leaders than you have. Too few will turn out to be good people leaders. Those who aspire for leadership but are unable to lead people well may need to settle for other types of leadership roles. A good work environment provides many leadership opportunities. Your aspiring leaders will be willing to wait for opportunities if your organization is a satisfying place to work.

Persistently and politely pursue answers from the candidate. When interviewing, do not settle for vague answers from candidates. If the answer to the question is important, politely pursue a more complete answer. You are not trying to get the answer you want to hear. You are trying to get enough information to understand whether the candidate will be a good fit.

It is surprising how evasive a candidate can be, but you deserve the answer so do not settle. The candidate may be avoiding telling the truth because they were untruthful on their résumé. The candidate may simply be afraid to admit ignorance. In either case, not answering the question—even if the answer should be "I don't know"—is unacceptable. Often when your team is discussing the interview later, someone will say, "I don't know the answer because the candidate was vague." That means someone didn't pursue the answer persistently.

Do developers make bad leaders?

Being a software professional has nothing to do with ability to lead. Two things make good leaders: behaviors and skills. If you focus on behaviors in your hiring of developers, they will be predisposed for leadership success. The hired candidate may walk in the door with the skills necessary to lead or not. If not, skills are easy to acquire through training and mentoring. People can acquire or modify behaviors, but it is much harder than skill development. Hire for behaviors and train the leadership skills.

Every interview is different; be creative. Every interview truly is different. Most will pose no major challenges, but many will. Be creative when dealing with strange situations. Hiring is not a predictable process; it demands adaptation and quick thinking as you keep the principles in mind. Don't be afraid to ask for help from your peers.

Take ownership of your needs, opinions, and decisions. Hiring is a collaborative process. The hiring participants have different needs, opinions, and perspectives. It is easy to be swayed by others in the process, so try to form your own opinions based on sound observations. If you feel compelled to change your mind after hearing what others think, question your motives before you question your opinions. Hold true to your observations, but not stubbornly so. Honest re-evaluation of your decisions is fine, but foolish supplication to the hiring guru's opinion is not. The hiring guru might have a point; just be sure you agree before changing your mind.

Respect the opinions of others. Likewise, respect the opinions of others. It is not your job to convince them of your views. Instead, state your opinion and give your reasons. If they are compelling, people might agree. If they don't, let it go. It is better to lose a candidate than to ruin a relationship with a workmate.

Leave the candidate with a positive impression of your company. Whether you make an offer or not, treat every candidate as if you intend to. You have succeeded if the candidate leaves wanting a job. The person you pass on today may return with more qualifications in the future. That person has a network, and the impression your company leaves can have a surprising influence on your ability to hire. Word gets around fast in the development community. Your company could develop a great reputation or a poor reputation even to the point where recruiters won't work with you.

Wait for good candidates. Be patient when hiring; you don't need to hire the first person that comes through the door. If you are not seeing many qualified candidates, rather than settle, focus on getting more candidates. There are always good people to hire, so wait for them.

Hire for the long-term. The average tenure of a software developer is only a few years, but good companies are able to keep their talented employees for much longer. Always hire with the intention that you want this candidate as an employee for a long time. A short-term view may lead to cynicism or compromise in the hiring process.

1.3 Hiring is like dating

Although it may a bit sound strange, a dating metaphor works well for hiring. But before getting all mushy, I must first construct a simple hiring process. Then I will discuss cost, values, and behavioral traits—these are important in dating too!

Hiring process overview

Although this book does not cover the entire hiring process, having a process skeleton in mind enables easier presentation of other hiring concepts. Think of the hiring process as a pipeline where candidate résumés come in and new hires come out. Along the way, you filter undesirable candidates out of the pipeline until only the best are left. Here are the steps of a basic process:

1. Accept résumés and log them for tracking and distribution.

2. Have individuals review résumés and recommend whether to phone screen or not.

3. Close out candidates who are not to be phone screened and schedule phone screens for the desired candidates.

4. Conduct phone screens; recommend whether or not to invite on-site.
5. Close out candidates who are not to be brought on site and inform them.
6. Schedule on-site interviews for desired candidates, making travel arrangements if necessary.
7. Conduct on-site interviews; recommend whether or not to make an offer.
8. Close out candidates to whom an offer will not be made and inform them.
9. Make offers to desired candidates; negotiate terms if necessary.
10. Complete the company hiring processes, including visa and relocation, necessary to bring the new hires on-board.

This simple process is sufficient for most organizations and it is easy to augment with additional details.

Cost management

One benefit of multi-phased interviews (review résumé, phone interview, on-site interview) is cost management. By eliminating unqualified candidates as early as possible, you save the wasted expense of later process phases.

As an example, consider a situation in which five teams have openings. A résumé review that takes about ten minutes for each of the hiring interviewers and a technical interviewer will have a direct cost in salary and lost productivity of about $100 per candidate, plus a small indirect cost for the time spent by others who depend on process participants. A one-hour phone interview with three interviewers will have a direct cost of about $200 and an indirect cost of around $100. A typical on-site interview involving the CTO, a pair of architects, the five hiring interviewers, and lunch could easily exceed $1000 in direct cost and several hundred in indirect cost.

Clearly, the earlier in the process an undesirable candidate can be detected and eliminated, the better. This is why it is so important to become a good résumé reviewer. There is even greater value in cash and productivity by limiting on-site interviews to only those candidates you are likely to hire.

Other cost factors should also be considered, including visa processing, relocation, and recruiting costs. Keep in mind that if you spent $10,000—or even $20,000—on these additional costs, this is still only about 10 to 25 percent of one year's salary. If your strategy is to keep employees a long time, then this cost will amortize and not be significant compared to the value of having a productive employee.

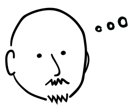
Organizational values

Hire people who exhibit values and behaviors consistent with your organizational values. Misfits will weaken the organization and be uncomfortable. It is the hiring company that is responsible for the fit, not the candidate, so value match should be a core part of your hiring process. Although the candidate is looking for a fit too, you represent your organization and have the knowledge to judge whether the candidate will mesh with your organizational values; the candidate does not.

At this point, you may be thinking, "Gee, I have no idea what our organizational values are." Chances are good that if you don't know your organizational or company values, there aren't any. If there are, any executive should be able to guide you to them, so ask around. If not, then your company has a great opportunity to establish values, set staff expectations, and begin to measure new hires against those values.

Organizational values need not reflect the way things are; they can be the values that leaders hope to realize. An organization will not realize those values unless they are defined and communicated.

From a hiring perspective, every new employee that exemplifies organizational values brings the organization, as a whole, closer to the desired value system. You can think of hiring, therefore, as a way to shape your organizational values!



Hire to become the organization you want to be.

Behavioral traits

I mentioned previously that communication, collaboration, and leadership are key behaviors to seek in candidates. Like values, behaviors should be established by the organization as most important.

Make sure everyone in the hiring process understands what behaviors to look for during the interview process. I have included a sample list of valuable behaviors in Appendix A.

The dating metaphor

If you combine the notions of the pipeline, values, and behaviors, you can imagine that hiring is like dating with the eventual intent of "marrying" candidates (putting aside polygamy in the metaphor). As you date, you start with a little information (the résumé) and slowly escalate the relationship: If the person interests you, you starting going to dinner, hiking, and movies (the phone screen).

As the relationship escalates further, you spend more time together. You meet the folks, talk politics and religion, and discuss the possibility of a future together (the on-site). If things are still good, a proposal is made (the offer) and hopefully accepted.

This analogy is helpful not only in reminding everyone that it takes time to get to know someone, but also that you are looking for a long-term relationship. You don't want to waste time and money on someone who isn't going to work out, so you eliminate them as soon as possible. Only those who really mesh with your organization are worthy of an offer.

1.4 Roles

For the purposes of this book and the lightweight process described in the previous section, it is helpful to define a set of participant roles. The roles are used in the descriptions throughout the book. Often an individual will take on multiple roles in the hiring process.

Process coordinator Coordinates most phases of the interview process: makes appointments, sets schedules, distributes résumés, invites attendees, and follows up with the candidates or recruiters.

Résumé reviewer Reviews résumés to determine if any candidates should be invited for a phone interview based on the job openings and the stated qualifications of the candidate. Identifies areas of interest, concern, and potential questioning paths to follow in other phases of the interview process.

Lead phone interviewer Leads the phone interview by doing most of the speaking and coordinating with the other phone interviewers. The goal of the lead phone interviewer is to conduct an orderly interview by promoting the general principles.

Hiring interviewer Has one or more openings to fill and participates in a phone or on-site interview to fill them. Responsible for identifying the degree to which a candidate qualifies for the openings.

Technical interviewer Participates in phone or on-site interviews and is specifically focused on evaluating the candidate's technical skills as they pertain to specific openings. All interviewers participate in this role to some degree, but it may be helpful to identify a technically focused role.

Behavioral interviewer Responsible for evaluating the candidate's behavioral strengths and weaknesses. All interviewers should fulfill this role, but it may be helpful to have a role focused on this area.

Opener Contact point for the candidate arriving on-site. The opener is responsible for bringing the candidate into the on-site interview process, which may involve meeting the candidate in the lobby and escorting the candidate to the first interview session.

Closer The final interviewer, who fields any remaining questions. The closer discusses salary expectations, benefits, *etc.* The closer finishes the process and escorts the candidate off the premises. If the candidate is from out of town, the closer ensures transportation is ready and the candidate has time to catch the return flight.

Other roles can be defined as they make sense, but it is important the right people fill the roles. Although the process coordinator, opener, and closer can be non-technical, all other roles are technical.

It is best to have the technical roles performed by leaders. Certainly the team leads fill the Hiring Interviewer role. It is a good idea to have directors involved, and even more senior management. The key players, though, are the team leads, because they have the greatest vested interest in hiring the right people.

Middle and senior management should respect the decisions of the team and overrule only rarely and judiciously. Your strongest technologists should be in the role of Technical Interviewer, but only if they are also strong interviewers. Technical interviews are very stressful on most candidates. It requires skill to get a good read on the candidate despite their stress level.

The Closer is often a senior manager who can discuss salary matters as well as company-related questions the leads may not be able to answer. Often having a more senior person do the close will leave a positive impression on the candidate.

1.5 Tracking

It is important that all phases of the process be recorded in a tracking system. Even more so if your company is growing or planning on hiring more than a few people. Process management systems like Jira[2] are ideal for this purpose. The tracking system becomes the focal point for the control, information, and comments pertaining to the candidate.

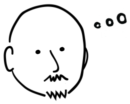
You should track how the candidate entered the pipeline (recruiter, job board, referral, *etc.*) It is helpful to have a comments section where notes on the candidate can be provided such as location, visa status, cover letter, or anything else of interest that isn't in the résumé.

The résumé and any other documents should be linked to the entry in the tracking system. As reviewers and interviewers encounter the candidate, they comment in the tracking system. The system can be used as the focal point for voting on whether to reject the candidate or move him or her to the next phase of the process.

Think carefully before adding salary information to the tracking system. You may not wish all the participants—people that need access to the tracking system—to see salary information. This information is valuable to track but you may need to have an additional means for tracking it.

[2] `www.atlassian.com/software/jira`

The process coordinator can be responsible for setting up the entries in the tracking system, moving the entry through the process phases, and closing the entry. It is a good idea to limit accessibility of a tracking entry after a candidate is hired to avoid awkward situations in which an employee can view the comments of their hiring process. Most tracking systems provide a means of specifying visibility either on a group or individual basis.



Tracking provides a foundation for continuous improvement.

Tracking is invaluable. Even a quick hiring process takes a few weeks and busy participants are likely to forget a lot about a candidate in that period of time. With a large number of candidates moving through the system, it doesn't take long before participants forget vital candidate information.

The tracking system provides a way to accumulate knowledge for each phase of the interview so ideas and concerns can be passed, for example, from the résumé review phase to the phone interview phase. These hints are vital to a successful process that leverages the power of many minds.

The tracking system also provides a way to monitor how interviewers are doing. Set up your tracking system to send email to all the participants. Encourage people to read the comments. Have a few senior interviewers review and give feedback on the comments to help the staff improve.

It is surprising how many candidates reapply, sometimes years later. It is so valuable to have the history available to see what happened before. Sometimes it is harmless: the candidate took another offer. Sometimes it is serious: the candidate failed a background check; the candidate had a dangerous personality, *etc.* Sometimes the candidate was too junior and specific concerns were raised: with history you have a yardstick to see how far the candidate has progressed, which is invaluable in hiring the right people.

Beyond comment tracking, it is also very useful to track the hiring pipeline performance. If you have a process coordinator, this person is in a position to track the data. Table 1.1 is a variant of a tracking table shown by Johanna Rothman.[3]

Report on this data weekly or monthly, depending on the speed of your process and throughput. The faster you are, the finer the reporting resolution.

[3]Rothman, *Hiring the Best Knowledge Workers, Techies & Nerds* [Rot04]

Table 1.1 · Track the hiring pipeline performance

Recruiting Strategy	# Résumés Received	# Phone Screens	# of On-sites	# Offers	# of Hires
Recruiter 1					
Monster					
Web Site					

The primary goal is efficiency. Here are some things to look for:

Evaluate data from end-to-end to measure strategy effectiveness. The bottom line is the number of hires a strategy is providing, so this is a key measure. Non-producing strategies ought to be considered for elimination if they have a significant cost associated with them.

Evaluate data from step-to-step to understand process efficiency. Each step (column) tells you something about a part of your hiring process in regard to the particular strategy. For example, a particular recruiter may be sending a huge number of résumés and few are phone screened; this is bad.

Recruiting strategy signal-to-noise ratio. Every résumé you receive has a cost so you want to minimize the noise (the number of candidates that do not get phone screens). If dealing with a recruiter or human resources (HR) department, manage expectations to improve the fidelity of job match. Remove noisy, expensive strategies that have low effectiveness from the process if you cannot improve the noise problem.

Establish targets for the signal-to-noise ratio. A 45 to 50 percent ratio of phone screens to résumés received would be very good for a managed process. By managed process, I mean that the input end of the strategy is doing something to improve fidelity. For example, a recruiter should be performing filtering for you. A less than 25 percent ratio ought to be investigated. In an unmanaged process, these ratio targets would be much lower.

Watch the cost. Recall the discussion about costs of the process steps on page 29. Actively manage expensive strategies to drive down costs.

Phone screen to on-site ratio. If you are doing a good job at résumé reviews, you should target around a 35 to 45 percent ratio here. If you drop below 25 percent for a couple of months, investigate your staff's ability to perform résumé reviews. Too many unjustified phone screens are being scheduled.

Ratio of offers to on-site visits. You should average close to 50 percent if you do very good phone screens. If you are dropping below 25 percent for a couple of months, your phone screens are ineffective.

Offer to hire ratio. Target about 70 percent. If you are dropping below 50 percent for a couple of months, either you are not selling your opportunity effectively in the on-site, or your offer is weak (think total package). Ask candidates (or recruiters) for feedback on why they turn you down.

The feedback here is critical. You measure the effectiveness of the pipeline and let the hiring staff see how they are doing. Encourage the team or even intervene to correct obvious problems. This simple tracking will provide you with all you need to monitor the health of your hiring system.

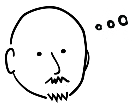
1.6 Nothing is certain—and that's OK

Hiring is a very fuzzy process. You can rarely be certain that you are making the right decision. When you reject a résumé, you cannot know if you are throwing away an opportunity to hire the person that will transform your company. It's OK.

The techniques described in this book are probabilistic and mastered through practice. You will learn as you practice how to "see" things that you couldn't see before. Your vision will improve but you will always continue to see mirages. Sometimes you will act on those mirages and make mistakes. It's OK.

The goal is to learn to make good choices most of the time without being inefficient. Theoretically, you could phone interview every candidate that gives you a résumé, and if you do good phone interviews, you will probably have a high probability of inviting the right people for an on-site interview.

In practice, you will not have the time to run quality phone interviews on all the candidates so you will have to weed out a significant number of résumés to save time and money. If you learn to do this well, your success rate will approach the all-in rate but with much less wasted time. Remember that your day job is to develop software!



Hiring is an imprecise art. You will make some mistakes; it's OK.

Alistair Cockburn draws an analogy with Ancient Japanese warrior training where three phases of learning are named *Shu*, *Ha*, and *Ri*.[4] At Shu level, the person requires a set of rules to apply, thus allowing a baseline of effectiveness. At the Ha level, the person seeks alternative rules, tries bending rules, or even eliminating some, thus optimizing. At the Ri level, the person is a master of situation and skill. The right techniques are applied in real time, backed by experience and reflection, with little or no thought of rules; new techniques are created as needed. It might help to think of Shu, Ha, and Ri as Learn, Collect, and Invent/Blend.[5]

With regards to hiring, a person at the Shu level will find value in the checklists, guidelines, *etc.*, found in this book, and will refer back to them often. At the Ha level, the person realizes the limitation of checklists and the like and collects other techniques, bends the techniques, discards some rules where they don't apply. She reflects on the underlying principles of the existing wisdom and can apply it at will with useful variations. At the Ri level, the person has internalized the wisdom and principles to such a degree as to wield them instantaneously in any situation. New situations are not a problem; the Ri level person merely adjusts and succeeds.

People new to these techniques desire certainty. They want rules. They often misinterpret the techniques as predictive rather than empirical. It is human nature when approaching a process to desire a set of functional rules to use right away. But for empirical processes, rules never work well. Now that you know this, your goal is to move through the learning phases as efficiently as possible, and avoid the temptation to grasp at certainties.

Since most readers are at the Shu level, I have presented information so that it can be used as guidelines. For example, I provide a résumé review checklist in Appendix C that will be valuable to the Shu. The Ha-level person

[4] Cockburn, *Agile Software Development, The Cooperative Game* [Coc06]

[5] "Shuhari" [Shu]

will begin to feel the checklist is unwieldy and incomplete. She will devise some new guidelines based on her observations. As she begins to transcend the guidelines, she will see higher order patterns and relationships that begin to tap into the real power of these techniques. As new people come into the hiring process, they will not understand how the Ri-level people arrive at some of their conclusions, and that's OK.

1.7 Conclusion

Hiring well is critical to your organization, but it is imprecise. You can radically improve the quality of your hiring with some foundational support. Develop hiring as a core competence based on hiring principles that promote quality and consistency. Understand the roles involved in hiring. Establish a tracking system to help manage the hiring process, to communicate about candidates, and to provide a feedback mechanism for staff improvement.

One more thing is required before you start evaluating candidates: defining the target. In the next chapter, I'll discuss how to define the job descriptions you are going to hire for.

Chapter 2

Defining the Target

If you don't know what skills, values, and behaviors you want in the candidate, you aren't going to hit the target. Establish ahead of time how you stand on the specifics of the position you are hiring for, including the job description, salary range, relocation, visa constraints, and any job specific values and behaviors.

Although defining job openings is not a core focus of this book, a basic discussion is fundamental to the rest of the material.

2.1 Writing the job description

The job description is your primary outreach tool into the candidate pool. To a significant extent, it will determine the candidates you receive, so it is worth the effort to get it right.

Spark the candidate's imagination. A good job description makes the reader curious about the opportunity; it makes them imagine your company is a place they might like to work. Say something about your company's mission and culture that is going to resonate with your target audience.

Use professional language. I've seen job descriptions that attempt to appeal to young people by self-selecting into cultural categories like outdoor enthusiasts or gamers. These will attract young people but maybe not the kinds of developers you are looking for if you value expertise. Avoid being cute and trying to appeal to the "What's in it for me?" crowd. Your company is a business, not a daycare.

Differentiate between what is desired versus required. You are unlikely to find a candidate that fits all your requirements and desired attributes. By differentiating, you make clear what is critical and what isn't. This gives you flexibility in hiring. You may be looking for experience in one technology but you might consider listing closely related technologies, especially for the desired but not required skills.

Be as accurate as possible in describing the position. Human nature dictates that candidates will apply who are both under- and over-qualified. Establish clear criteria for dealing with these groups. For example, an under-qualified candidate might be acceptable if they have a clear upside long-term benefit. Maybe they exhibit the values and show a strong growth path. An over-qualified candidate might be shifted toward another job opening, or might be willing to take a lower position with the expectation of promotion if performance dictates.

State the specific values and behaviors you care about. By doing so, you are doing the candidate and yourself a favor. The candidate learns something very important about your organization and whether it might be a good fit. You get the advantage of another filter to strain out unwanted candidates.

2.2 Setting the salary range

Know the salary range for the position before interviewing. Deciding late could muddy offer negotiations and could make it more difficult to work with recruiters.

Keep abreast of going rates in your local market for similar positions. Likewise, understand how the work-life experience at your company stacks up to the local competition. Keep in mind that the smart candidates realize an offer has many components to be considered holistically. Be competitive and have an answer when the inevitable comparisons arise. Take comfort that the bad places to work usually pay more.

Keep in mind the salaries of existing employees. It is wise to maintain parity in staff salaries; in a growing organization, this is a continual challenge. Instead of offering uncompetitive salaries for parity sake, consider truing up staff salaries occasionally. Always take value into account. Not all "Senior Developers" are equal. Parity takes relative value into account.

Avoid stating salary on the job description. All it does is set unrealistic expectations and restrict your flexibility. If you share salary information with recruiters, do so in a range and make it clear the salary range only applies to qualified candidates (in case you choose to hire under-qualified candidates into a lower position).

> **If the candidate wants too much money**
>
> Candidate salary might be cause for elimination up front, but I have found that what the candidate says before the interview and after are often very different. Sometimes the candidate doesn't understand the relative costs of living in your locale compared to their current residence.
>
> Other times, the candidate accepts a much lower salary because the full offer package offsets the salary deficit. Sometimes—for whatever reason—the candidate simply has unrealistic expectations and eventually comes around.
>
> Ideally, your company and its hiring process make such a positive impression on the candidate that he or she is compelled to accept the offer! So if the candidate's stated salary requirements are "in the ballpark," and the candidate appears promising, I prefer not to eliminate on grounds of salary.

2.3 Relocation and visa constraints

Establish in advance what your relocation policy will be for the position. If you are not considering out-of-state positions, say so on the job description. If you are considering out-of-state candidates, state whether you will provide relocation assistance.

Many companies have relocation policies; you can refer to your HR organization for details. How you package relocation to candidates can have a big impact on your workload and new hire satisfaction. I've seen packages that are difficult to implement and explain. Be sure to set clear expectations with your candidates and new hires to avoid frustration.

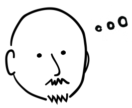
In today's job market, demand outstrips supply for software developers, so a large number of candidates will be immigrants. If you are unwilling to hire employees who need immigration assistance, you will be drastically reducing your hiring pool.

Set guidelines on how you plan to handle the various work permit situations (H1B, Green Card, *etc.*). Decide what categories you are willing to sponsor and what processing and legal expenses you will cover.

These costs can be significant, so it is common practice to include a service length commitment from the employee. There is often a waiting period after the hiring date, before the company will begin pursuing work permit processing. Consult your HR department or legal counsel for details.

2.4 Getting on the same page

It is fundamentally important that the hiring staff is on the same page. Everyone must understand what are the most important things to look for in a candidate. There are many tools at your disposal for aligning the hiring staff but the job description is the target and, therefore, is most fundamental. Every time a position opens or changes, use the job description as one tool to keep the hiring staff on the same page.



Everyone involved in the hiring process must know what to look for in the candidates.

As hiring progresses, it is common to make mid-course corrections. Set up a mailing list for the participants in the hiring process and send updates to keep everyone on the same page. For example, if hiring demand outstrips the supply of candidates, the organization may decide to *lower the bar* a little for new hires. Conversely, you may be trying to lift the organization by hiring more carefully, so you *raise the bar*. It is imperative you have a way to communicate this change to all the participants. Be clear what is being relaxed and what is not. For example, you can hire people with less experience who are still intelligent, and possess the behaviors your organization values.

If there are any reasons for automatic elimination of a candidate, make them clear to the hiring staff. For example, you may have a policy that all developers and testers must work on-site—no telecommuting. Or, the job may involve travel or have special dress codes. If a candidate cannot comply, this is grounds for elimination.

In a growing organization, your hiring staff is likely to grow as well. Put a training program in place to bring new participants into the process. Consider setting up workshops to disseminate best practices and answer the inevitable questions that arise among the hiring team. Maintaining team alignment is one of the best things you can do to keep the hiring machine operating at maximum efficiency.

2.5 Conclusion

You've created the job description and you have established your sourcing strategy.[1] You've established a salary range, relocation and visa constraints, and you've communicated this information to your hiring staff. The résumés will be arriving soon. In the next chapter, you'll learn how to evaluate them.

[1] As mentioned in the introduction, sourcing is beyond the scope of this book.

Chapter 3

Reviewing Résumés

A résumé is usually the first point of contact a candidate has with a company. Therefore it is the candidate's most important sales tool. With this in mind, expect excellent résumés and be very suspicious of poor ones.

A candidate should craft a résumé that shines the very best light while trying to remain truthful. This balancing act often leads to a degree of exaggeration or a little buzzword bingo.

Few candidates lie. Many stretch the truth. Most choose to list every technology ever encountered in their career regardless of meaningful experience. I've come to expect all this, and it is easy to spot. What may be surprising is that many candidates have a sometimes wildly inaccurate understanding of their own abilities. Occasionally, a candidate will underestimate his or her abilities, but the majority of candidates with unrealistic self-understanding will overestimate their skills.

In hiring, the somewhat competent, grandiose ones are those you must watch out for most. Since these people have some skill and are overconfident, they often present themselves well and can fool a recruiter or less experienced interviewer.

As you practice hiring, you will improve your ability to discern the difference between a candidate's abilities and his self-perception, but you will never achieve certainty. Résumé reviewing is an imprecise art that takes time to develop. Don't be paralyzed by that. Do your best, and expect good résumés. Follow the advice here until you begin to adapt and transcend. There are plenty of candidates out there, so don't be afraid of losing a few through the cracks. If a candidate's résumé is questionable, chances are good that person is not someone you want to hire.

The Dunning-Kruger effect

There is little danger of hiring someone who underestimates his or her abilities, but one who grossly overestimates can be quite dangerous. They tend to be difficult to work with, because they believe they know more than their peers. The peers become aware of the problem, grow resentful, and are unable to collaborate. The grandiose employee tends to make decisions he or she is unqualified for, bad decisions that can have drastic long-term impact.

Research has borne this out in what is called the Dunning-Kruger effect.[a] The effect is a case of cognitive bias in which "...people reach erroneous conclusions and make unfortunate choices, but their incompetence robs them of the metacognitive ability to realize it." This is an old and well known idea. The authors quote Charles Darwin who wrote, "Ignorance more frequently begets confidence than does knowledge."

The authors hypothesized that for any given skill:

1. Incompetent individuals tend to overestimate their own level of skill.
2. Incompetent individuals fail to recognize genuine skill in others.
3. Incompetent individuals fail to recognize the extremity of their inadequacy.
4. If trained to improve their own skill level, these individuals can recognize their previous lack of skill.

[a] Kruger and Dunning, "Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments" [Kru99]

3.1 Résumé review processes

Determine how to organize résumé reviews. The processes used should match the résumé backlog and the hiring needs at the time. If there is only one hiring group or one hiring manager, the process can be very lightweight. If there are multiple hiring groups or many open positions, the process may need to be more sophisticated.

Over time, certain members of your staff will demonstrate greater skill at résumé reviewing. Try to ensure that every résumé is seen by at least one skilled reviewer. This not only helps you avoid accepting poor résumés, but also provides a teaching opportunity for the less-skilled reviewers.

Consider having a skilled reviewer pre-screen all incoming résumés to eliminate the obvious mismatches. This adds another step in the process but might save a lot of time and money. Rotate the position if possible. The downside is that the quality of your process hinges on the ability of one individual. Consider this approach if you have no other vetting methods in place. For example, if you get the bulk of your résumés through job boards or direct submission, this approach might be helpful. On the other hand, if recruiters who are filtering well for you submit the bulk of the résumés, then pre-screening is probably a wasted effort.

Here I describe some common types of résumé reviews and highlight their benefits and weaknesses. In all cases, the reviewers should enter their observations and opinions into a tracking system after each résumé review.

Lone reviews

The main advantage of lone reviews is that you maximize your résumé review throughput. The disadvantages are that you put all discretion into the hands of one person, and that person's workload might be dominated by résumé reviews at the expense of other useful work. Also, your results will be only as good as the skill of the individual reviewer. Be sensitive to this fact. If the percentage of phone screen rejections is too high or too low, it may be due to weak lone résumé reviews.

Lone reviews are useful in several situations, assuming strong reviewers are performing those reviews. Lone reviews can help work through a backlog of résumés quickly (though you should also reflect on why such a backlog exists in the first place).

Lone reviews can also be useful if the organization has a strategy of hiring contractors or contract-to-hire candidates. The distinguishing characteristic here is that because the candidate can be let go if things don't work out, you might be able to afford less rigorous reviews.[1]

[1] My experience is that a contract-to-hire approach is more expensive in lost productivity than involving more people in hiring full-time employees. The cost of getting any employee up to speed is approximately fixed and is spent regardless of whether the contractor is let go or not. This strategy is also demoralizing to the staff as failed candidates are released. I feel

It is best to involve hiring interviewers that have openings to fill, but sometimes this is impractical. By having a manager perform lone reviews, hiring interviewers can devote their precious time to later phases of the hiring process. Of course, if the quality of the lone reviews is poor, there will be no benefit to the hiring interviewers, because they will waste time interviewing weak candidates.

Pair reviews

In pair reviews two reviewers examine each résumé together. Pairing has several benefits over lone reviews. One additional opinion significantly increases the fidelity of the review process by reducing the number of fruitless phone screens. Pair reviews allow the sharing of information between reviewers, which can lead to a synergistic expansion of information. Pair reviews are also excellent for training and mentoring less experienced reviewers. I have already stated that the additional resource consumption is worth the effort, so I don't count that as a disadvantage.

One negative to pair reviews is the possibility for conflict; with no third person as a "tie breaker," differences of opinion can remain unresolved. Consider establishing a resolution process for these cases.

Pairing requires two people setting aside the same time block, which might compromise the individuals' schedules. Scheduling that time might create delays that could lead to a backlog of résumés and missed hiring opportunities.

Try to involve at least one hiring interviewer when pairing, because it is always best to include those with vested interest in the hiring process.

Distributed résumé reviews

Distributed résumé reviews are an efficient approach when there are a sufficient number of qualified reviewers. In this approach, the process coordinator notifies reviewers of new résumés once they have been entered into the tracking system. Reviewers can examine the résumés on their own schedule, allowing them to manage their time appropriately.

The main advantage of this approach is that you can involve a large number of people easily and get the benefit of consensus. It is easy to include

---

it is nearly always better to hire full-time.

hiring interviewers in this process. There are no significant disadvantages, but there are coordination challenges.

The first challenge is how to ensure timely reviews. Keep track of the age of each résumé and communicate expectations about how long a résumé can be outstanding before a decision is made on conducting a phone interview. It is helpful to email a list of outstanding résumés daily with the status of each: age, number of comments, needs more comments, *etc.*

The second challenge is determining when to move the résumé out of the review stage to closure or phone interview. Set a policy on how to determine résumé review completion. For example, you could base it on a minimum number of reviews, minimum positive or negative reviews, or a combination.

Third, there may be a tendency by reviewers to be wishy-washy. Insist on decisiveness: either yes or no. It is fine to express a degree of uncertainty but a choice is necessary. This process requires commitment by the participants and some degree of maturity in résumé reviewing.

There are many details and variations to address in your local context. Be creative and focus on quality and efficiency.

Résumé round tables

Résumé round tables combine some of the benefits of pairing and distributed reviews. Once a significant number of résumés have accumulated, some reviewers (three to five as a guide) are invited to a résumé round table meeting. A reasonable minimum number of résumés to review is six to ten. Each reviewer is given a copy of the same résumé to read and should write their observations and comments on it. After everyone has done this, observations are expressed either in turn or in open discussion. A vote is made on whether to schedule a phone interview. Lack of consensus may lead to more discussion. Establish requirements for determining résumé resolution: number of yes votes, majority, *etc.* Have a computer handy to enter brief comments into the tracking system immediately. Move to the next résumé.

Once all résumés are reviewed, consider having each reviewer enter specific comments into the tracking system. The process coordinator can then schedule phone interviews for the good candidates.

This process has at least four benefits:

- It promotes a timely review and response completion.
- It provides an orderly process.

- It develops skills in résumé reviewing/filtering through real-time mentoring.
- It is a good way to eliminate résumé backlog without compromising quality.

These four benefits should be weighed against the cost of inviting a possibly large number of people to a potentially long meeting. Their schedules may not be compatible, which can lead to delays. There is a tendency to rush through the résumés so a mature reviewer should monitor that each résumé gets appropriate attention. Additionally some résumés may sit idle waiting for a sufficient number of reviewers to become available.

Experiment with the number of reviewers that works best for you. More than five is probably going to be unwieldy. Also experiment with different ways to get the reviewers' comments into the tracking system.

Résumé filtering

If you have a large number of rejected résumés during reviews, it probably means there is insufficient filtering at the early stages of the hiring pipeline. You can detect this if you use the tracking technique I suggested earlier.

You could choose to improve the up-front filtering, or you could do your own filtering. An example of improving up-front filtering is working closely with recruiters or HR to help them understand how better to eliminate obviously unqualified candidates.

If you must do filtering at the hiring organization level, then you can use any of the previously mentioned types of review processes. The goal is to eliminate the obviously unqualified candidates as quickly as possible. It is probably easiest to add a filtering phase where you sweep through the résumés and divide them into two categories: unqualified and needs detailed review. Résumé roundtables are excellent for filtering.

A challenge when filtering is to strike the right balance between unintentionally eliminating qualified candidates and minimizing the filtering effort. How aggressively you filter out résumés will be dictated by the rate of résumés you receive, and the number and quality of positions you must fill.

Filtering demands the ability to quickly get a sense of the quality of the candidate, so you want people who have strong review skills. The reviewers also need to understand the positions available that they are filtering against.

Later in Section 3.4, I will cover review techniques in more detail and point out how to approach filtering a résumé there.

3.2 Parts of a résumé and their value

Every résumé is different, but most résumés tend to follow the same basic format. The order and layout varies but the content is about the same. Résumé 1 on page 51 shows an example with most of the parts I discuss.

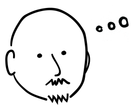
Objective or summary

The objective section will make a brief statement about the candidate's career goal, typically focusing on the near future (*e.g.*, the next job). Sometimes the candidate will provide a career summary instead that encapsulates their career in a few paragraphs.

Objective statements usually aren't of much value. They tend to be too general to provide information for the job opening at hand. Occasionally an objective appears to have been written specifically for some other position or for some earlier time in the person's career. In these cases, the objective is more confusing than helpful.

Summary statements, on the other hand, can provide some useful information. If the candidate provides a career synopsis it can shed added light on the career history, tying the various jobs together in a common theme. Often the candidate will use this section to highlight self-perceived, meta-level strengths such as value focused, energetic, quick learner, collaborative, or "can do."

I like to perform a *consistency check* between the summary and the job descriptions. The job descriptions should support the summary, but if they don't, then that is a concern.



Consistency between the objective or summary and the rest of the résumé can be an indicator of truthfulness.

Since summaries, and even objectives, reveal a bit of the candidate's self-perception, they can help evaluate the accuracy of their perception. I have found that the accuracy of a candidate's self-perception is an important indicator of future performance. This is where the *Dunning-Kruger Effect*

John Doe
Architect/Lead Developer

SUMMARY

- Expert in software architecture on Java platforms
- Deep experience in data modeling and RDBS
- In-depth knowledge of design patterns, and object oriented design
- Clear communicator and experienced team leader

EXPERIENCE
February, 2006 – Present *Hotel California, Inc., Hollywood, CA*
Architect/Lead developer

- Design and development of the business rule engine and supporting database used by hotel booking systems. Spring-based application residing on Tomcat. Client requests to Web Services using XML-RPC and HttpInvoker gathers additional data from several systems, applies business rules and returns the response.

Technologies: Java, Visual Rules, Spring, Ibatis, Apache, Tomcat and MySql.
March, 2005 – February, 2006 *Credible Health Inc., Frozen North, Canada*
Senior Software Developer

- *etc.*

SKILLS

- Java,J2EE, Spring, Hibernate, Ibatis
- RDBMS (Oracle (9i,10g), MySQL, SQL, PL/SQL)
- SOA, Web services (Xfire, Axis, Oracle web services), BPEL
- Web/Application Servers (Apache, Tomcat, WebSphere, Weblogic)
- Source Control Software (VisualSourcesafe, Clearcase, CVS)

EDUCATION
May 2001 University of Oz Over-the-Rainbow Master of Science in Wizardry
Dec. 1998 University of Oz Over-the-Rainbow Bachelor of Science in Computer Science

Resume 1 · Example showing typical parts of a résumé.

may come into play. A summary or objective that is inconsistent with the rest of the résumé is an indication the candidate has an unrealistic self-perception. If it is consistent with the rest of the résumé, the candidate may be more grounded, more realistic, and more mature.

Skills

The skills section usually serves two purposes. The first is to convey information to a human; the second is to convey information to a computer program. In either case, the typical skills section will be stuffed with a list of buzzwords that add very little value.

Usually, the skills section will include every technology the candidate has been in proximity with, regardless of whether it was actually used. Occasionally, the candidate will only list technologies used and this is evidenced by the relative shortness of the list. In these rare cases something might be gleaned from the skills section.

More often, the skills section is written for the purpose of getting the résumé through filtering software. Recruiters and HR departments often run résumés through filtering software that attempts to match résumés with specific job requirements. This eliminates much of the tedium of hiring at the expense of fidelity. From the candidate's perspective, the best strategy is to list everything they've ever come in contact with so as to not get screened out by a lousy filter.

> **Truth stretching or dishonesty?**
>
> People will sometimes go to great lengths to get a job; some will even be dishonest. The skills section is an area where people stretch the truth and so the information often cannot be trusted.
>
> Try to be empathetic when evaluating the truthfulness of résumés. Is the information stretched for benign purposes such as filter passing, or is it more serious? Sections like education and job descriptions must be accurate; have no tolerance for inaccuracy in those sections. On the other hand, if you keep in mind the dual purpose of the skills section, you should not be troubled by exaggeration there. Someone who is inaccurate or dishonest in the key sections of a résumé is likely to behave poorly as an employee.

Education

The education section is where the candidate lists significant education and training they have had. Software professionals, typically have a college or university level education included on the résumé. Candidates will sometimes list other trainings or certifications here or elsewhere.

This section is important depending on the education your targeted position requires. I will discuss education in more detail later in this chapter, starting on page 72.

What the candidate chooses to provide (or leave out) can say a lot. You will see a wide range of schools and degrees. Understand what the education requirements of the position demand and do your research regarding institutions a candidate lists. If you have not heard of an institution, do some research to ensure it exists and whether it is any good. Review the curriculum the candidate followed to get a sense of what she ought to have learned.

Employment history

In the employment history section, the candidate lists the positions held and gives a synopsis of each. Expect to see the employer, city, state, country, job title, and dates of employment to be listed for each position. Often a contextual description of the company and projects is included. The most important part of each position description is the list of responsibilities the candidate claims to have fulfilled.

You should be concerned if any of the above information is absent. This is the most important section by far and provides the richest set of information, including possibly information the candidate did not intend to provide.[2]

Certifications

Candidates may list certifications in a dedicated section or in another section such as Education. Certifications are difficult to interpret because there is a culture and motivation around each kind.

For the hiring company, the primary value of a certification is that it gives tangible evidence of a level of mastery of the subject. Unfortunately, passing a certification test typically does not indicate a high level of mastery. To the contrary, in areas I have expertise, I have observed that possession of a certification rarely indicates mastery.

If a company that is hiring a Java developer has no one with sufficient expertise to evaluate candidates, then a certification may provide solace. My

[2] You can harvest a wealth of information from a résumé that the candidate did not intend to provide.

Certifications: good or bad?

I have expertise in two areas where certifications are popular: Java programming and agile methods. In hiring senior Java developers, a solid majority were not given offers. Passing an online test has weak correlation with job success. Crib notes are available for all the popular tests, so neither practicum nor experience is required.

The agile world is even worse. Until recently, one could attain Scrum-Master certification by merely attending a two-day course.[a] I have attended two such trainings, one for my education, and one to validate a course my organization hoped to use. The instructors were world-class and yet, the results were highly variable. I received two certifications just by staying awake!

Some people get certified because their employer requires it or is willing to pay for it. The employer either wants to improve competence or, in the case of outsourcing, uses it as a sales tool to assure their customers that only "certified developers" will be working on the project. I think outsourcing is a bad option for companies who consider software a core asset; companies who seek outsourcing are likely to be impressed by certifications.

Some candidates believe a certification carries currency. After all, the candidate made note of the certification, possibly even pasting the certification icon at the top of the résumé. I think the idea that certification carries currency is misguided in most cases.

Unless you have good reason to act otherwise, treat certifications with skepticism.

[a] The Scrum Alliance has improved this situation by creating a test that must be passed before attaining the certification.

conviction, however, is that software professionals are best suited to hiring other professionals and that certifications are a poor indicator of skills.

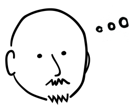
Publications

This is where a candidate lists published work they authored or co-authored. Often a candidate will also list presentations, committees, and other technical leadership roles.

Unless the position for which you are hiring is related to research, you are not likely to see many candidates who have been published. Publications and presentations, *etc.*, may indicate intelligence, communication skill, the ability to innovate, technical leadership, and the ability to become a subject matter expert.

If publications are listed, evaluate them to determine what they say about the candidate. The first consideration is the publications' relevance to the positions you are hiring for. Discovering a candidate that happens to be an expert in an area of interest is truly a bonus. This is pretty rare so—everything else being satisfactory—pursue such a candidate aggressively.

Publications and other public work that are not specifically related to the hiring position may nevertheless be very important. For example, attaining mastery of a subject is not a common skill. People who have achieved mastery in one area usually can reproduce mastery in another area. The key, then, is to determine whether the candidate's achievement was mastery in that area.



A person who has achieved mastery in the past is likely to be able to repeat the process in other areas.

Today, just about everything someone writes and posts on the Internet is syndicated; it is trivial to get "published." The upside to the Internet phenomenon is that you can usually track down and read published works of all kinds. It is a good idea to review listed publications, because what someone writes speaks volumes.

There are many things to consider when evaluating published works and presentations. The first is whether the candidate is the primary or lone author. Typically the primary author is listed first. Be mindful of research publications: Often the candidate is not the primary researcher but an assistant. College research professors, for example, put their minions to work assisting in trivial (and sometimes not so trivial) tasks, and list their names on papers in return. Typically the more names listed, the less influential the secondary authors are. Explore other publications from the primary author and the candidate to tease out the candidate's actual relevance in the area.

Has the candidate been published multiple times for the same or similar topics? If so, this indicates devotion to a topic that might indicate mastery. On the other hand, numerous publications on unrelated topics usually indicate a problem: a person with strong opinions and a strong desire to be heard, but a lack of depth. This person may be under the Dunning-Kruger effect.

If the candidate has a secondary degree, they often have published a thesis. A thesis is an original work requiring significant effort; it is highly revealing. Unfortunately, the vast majority of thesis work is neither innovative nor relevant. It is extremely unlikely a thesis will have any relevance to your interests, so look beyond the topic and toward the style, communication, competence, quality of research, and mastery of the subject matter.

If a candidate lists blog posts as publications, the only way you can make a judgment is to read the posts. Likewise, there are many online journals and magazines that are of little consequence. They accept most submissions with a minimal amount of review. Mere publication is of no value; you have to read the publication and judge for yourself.

The best publications and conferences are those that are peer reviewed and have a low acceptance rate. IEEE and ACM publications and conferences are examples at the high end of the spectrum. There are many publications and conferences that are not research-oriented but still have a low acceptance rate. You can inquire with the publisher or organizer to discover their criteria and acceptance rates. The candidate who has passed the scrutiny of discerning publishers or conference organizers is more likely to be a desirable employee.

Publications are great because they give you another good tool for understanding the candidate. They also supply fodder for direct discussion during a phone or on-site interview. Do your homework and familiarize yourself with their work. If you have in-house experts in the areas of the candidate's publications, get them involved in the interview. Additionally, if the candidate is a real winner, you are likely to gain their respect if you can speak intelligently about their work. This becomes a powerful selling tool.

Conversely, a candidate who cites several weak publications can be a danger. Keep an eye out for negative traits like truth stretching, deception, arrogance, and entitlement. Related behaviors tend to come packaged together. Look for correlations like this as you gain expertise in hiring.

Interests

Some candidates include a section where they list non-professional interests. In some cases, the candidate has interests worth considering, but often, the content is superfluous or even unprofessional.

Interests that demonstrate deep skill, high creativity, leadership, or altruism might be considered positive. As with everything the candidate puts on a résumé, judge it as evidence of whether the candidate will bring values and behaviors your organization desires.

3.3 What to look for: perspectives

I find it most useful to evaluate résumés from a set of important perspectives. I present a set that works well, but as you move from Shu to Ha or Ri, I suggest you revisit this list and add to it. Try to use all the perspectives on every résumé review. Here's my list:

- Position Match
- Culture/Environment Match
- Experience
- Performance
- Growth
- Leadership
- Impact
- Potential
- Education
- Problem Detection

Reviewing a résumé with perspectives in mind makes the review more purposeful and quantitative. When you comment on the résumé, you can use the vocabulary of the perspectives to express yourself.

You need not limit yourself to this list of perspectives. If there are other broad areas that make sense in your context, define and use them. The perspectives in my list are described in the following sections.

Position match

Position match is the most basic perspective. Use it to determine how well the candidate's résumé matches the skills and experience required for the position in question. Before looking at a single résumé, realistically establish

Feeling judgmental?

Most people are uncomfortable judging others; especially if the judgments seem based on sketchy information. You wouldn't judge your friends or family—or even strangers—this way, but hiring is different.

Your job is to hire the best candidates, and that requires sometimes-uncomfortable evaluations and judgments. You must be critical, not in the sense of being "inclined to find fault or judge with severity, often too readily," but rather, "involving skillful judgment as to truth, merit, *etc.*"[a]

Being uncomfortable with judgment is healthy; use it as a reminder of the importance of what you are doing and the unavoidable imperfection of your perceptions and judgments.

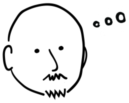
Hiring is an important activity for the candidate, your employer, and you. For the candidate, getting a job is hugely important and personal. Your employer desires the very best and has asked you to deliver. You may eventually work with this candidate; your future success may even depend on him. Don't lose sight of just how important hiring is.

Be mindful that the information you are analyzing is vague and inaccurate and therefore will lead to imperfect judgments. As a candidate goes through the hiring process, you accumulate information that incrementally paints a picture of the person. New information will either reinforce the picture or force you to change the picture.

A difficult skill to master is the ability to let go of judgments when the data suggests they are inaccurate. For all sorts of reasons, people tend to hold on too tightly to their initial judgments. Develop the ability to let go of your early judgments and repaint your picture whenever the data suggests you should.

[a] Definitions from www.dictionary.com.

your expectation of finding a candidate that matches the position requirements. Use the job description as a guide. Prioritize the requirements and categorize them as *must-haves* and *nice-to-haves* so that you can make decisions about how well a candidate qualifies for a position.



Focus on how well the candidate matches the job description.

If the résumé lists skills, take note of the ones you are interested in, but do not make any judgments on mastery by virtue of name-dropping. As previously mentioned, the technologies are often listed to get through a filtering program or process.

Focus on the details of the candidate's employment history to find evidence of actual technology use or skill application, if not mastery. Look for a good degree of correspondence between the buzzwords (in the skills section) and the experience stories in the job descriptions. Look for any clues that this candidate has significant skills and experience in the areas of interest.

Consider how long ago the candidate had the relevant experience. Work done with a technology of interest too far in the past has much less value than work done recently. Take into account whether that technology has changed significantly in the time period and how much you value the latest trends in that technology.

Leadership has a long shelf life; technology doesn't

The value of experience with a specific technology degrades with time. As the technology changes, the candidate must stay current to maintain relevance. If the candidate has not used an important technology for a while, she will have to catch up. In cases like this, you must determine how much effort (and lost productivity) will be required for the hired candidate to catch up. Keep in mind that a person who has attained mastery in the past is likely to be able to repeat the process.

Leadership, teamwork, and communication skills, on the other hand, mature with time and transcend trends. Furthermore, these skills are not as easy to acquire as technical skills. Place a high value on candidates that have demonstrated these soft skills. You will have more success hiring for soft skills and teaching technology than the other way around. Ideally, hire for both, but if one of them has to slip a little, let it be technology.

If there does not seem to be much evidence that a candidate has the skills you are looking for, take this as a significant detractor. Do not assume that just because the candidate has mentioned a skill or technology that she has actual experience using it. Expect to see detailed mention of it in job descriptions or project descriptions where the candidate played an important role. For example, consider the work history segment from a fictitious résumé in Résumé 2.

SomeCompany, Someplace March 1996 – Aug 2009

- Led 10 member software development team to deliver new JAVA application that dispatchers use to manage couriers.
- Designed and implemented interface to improve management of customer data residing in an ORACLE database.
- Defined support processes to reduce down time and improve responsiveness when incidents occurred in software system.
- Successfully transitioned team to JAVA and WEBLOGIC.
- Mentored junior developers to improve quality of software.

Resume 2 · Example work history segment.

In this example, I observe potentially significant Java experience based on the first and fourth bullets. I weigh that against the fact the person played a leadership role in the first bullet and may not have done much coding, and that the candidate lists technologies in all caps (probably for emphasis, or maybe due to ignorance).[3] Although Oracle is mentioned, there is no indication of actual usage. There are several signs of leadership, including team leadership in bullet one for new development, process improvement in bullet three, possible training in bullet four, and mentoring in bullet five. There is no indication of the difficulty of the work or the development tasks undertaken by the candidate. Bullet two is unclear about what the interface (UI, object, *etc.*) is and what the candidate is talking about. I would take note of the lack of clarity and detail in this job description. If a phone interview is conducted, these vague areas must be explored.

In the absence of mitigating circumstances, eliminate candidates who do not have the must-have skills and experiences. There is no point wasting any more time if the person is not going to meet the requirements.

[3] Misuse, misspelling, incorrect capitalization, *etc.*, is a concern. If the candidate misspells or misuses the name of a technology, I wonder just how much he knows about it.

Overqualified?

What do you do if someone seems to have all the requisites and a lot more? This can be tricky to evaluate. It is easy to confuse over qualification with something else. For example, someone who has moved into management may be rusty, or they may just want to be technical again. This isn't overqualification per se since management typically isn't a technical role. The candidate may have enjoyed a higher salary and more responsibility as a manager, but probably didn't grow technically.

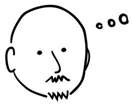
Members of the hiring team might make assumptions that lead them to eliminate the candidate that has, in the areas you desire, all the requisite skills; more years of experience than you require; and maybe technical leadership in the form of publications. They might assume the candidate is interested in a more challenging position or a higher salary than you can offer. If you believe a candidate exceeds the qualifications of the job, don't eliminate the person without facts. Call her and resolve the concerns before scheduling the phone interview.

Sometimes salary is a problem. If the candidate came to you through a recruiter, salary concerns should have already been addressed. Communicate this to the hiring staff so they don't eliminate these candidates for salary reasons. If the candidate came in unfiltered (say from the company Web site), you are more likely to experience mismatches of all kinds. A preliminary call becomes more valuable in these situations. Design your hiring process so that salary concerns are handled before the candidate is put into the review part of the process. Interviewers then need not worry about salary concerns and can focus on the candidate's qualifications.

Culture/environment match

Regardless of how well the candidate fulfills the position requirements, if he is not comfortable in your culture or environment, the candidate is not likely to remain long at the company. Cultural and environmental factors are major contributors to job satisfaction and, therefore, performance.

If your company places a high value on its culture and working environment, most quality people will adapt if they are leaving unhealthy or moderately lame cultures or environments. For example, a new employee coming from a company where the culture encouraged sixty-hour work weeks is probably not going to have trouble adapting to a forty- to forty-five-hour work week. On the other hand, a person with a history of working in companies where quality communication, collaboration, and transparency are not valued may experience a culture shock moving to a company that practices these behaviors.[4]



Seek signs of strong environmental and cultural match for better employee satisfaction and retention.

Consider the pace and style of work at your company. For example, if you build medical devices and implement the software for those devices, you probably have deliberate, safety-centered processes and work styles. Code is probably not delivered frequently and reviews and other mechanisms are probably in place to assure safety. Hiring a person who has only worked at small startups might not be the best decision; at least not without taking the time to explain what goes on in your environment. Even with an explanation, and an expression of understanding, candidates who need a job are unlikely to fully consider or even be capable of completely understanding the implications of drastic changes like this. In a phone or on-site interview you might be able to tease out a problem by deeply exploring what the candidate truly enjoys about the cultures she has experienced. If those values are inconsistent with your culture, you will probably have a problem in the future if you hire the candidate.

Some technologies have significant cultural implications. For example, people who avoid the use of open-source tools tend to bring a certain set of values that are quite different from the values of those who make strong use of open source tools and software. I find this a great indicator of fit.

As you read the employment history, take note of the companies. Does what you know (or what you can learn) about each company lead you to believe they are culturally similar or different from yours? What are the relative sizes of the companies? How old are the companies? Do the companies develop software at several locations? Do you know anyone who has worked at any of these companies? Are they working with you now, and if so, how successful has their transition been?

---

[4] I write these transition concerns as if the candidate is moving from less healthy to more healthy cultures and environments. I do this because I hope that if you are reading this book, your culture and environment must be good! If not, these concepts tend to go both ways.

Waterfall or agile

Candidates steeped in waterfall techniques will have an adjustment period if moving to an agile environment. In one company, we were pretty faithful to agile principles. In résumé reviews, we sought indications that candidates had worked in an agile environment. Most who have agile experience will mention it, especially since it was in our job description. We discovered that among those who said they worked in agile environments, few really understood what they were doing. Even so, moving from poorly practiced agile is less of a leap than moving from waterfall.

People who have only worked in waterfall environments tend to struggle with aspects of agile development. These include less detailed initial requirements, continuous customer involvement, changing requirements, frequency of releases, and the blurring and continuous nature of activities such as analysis, design, coding, integration, and testing.

I like to explore how happy the candidate was in their waterfall environment. Did they think it was effective? Were they curious about agile? What did they understand about agile? Usually people are eager to leave the waterfall methods behind, but from time to time, I encounter those who either do not believe agile methods can work or are quite pleased with waterfall. This could be a problem.

How can you tell if someone is agile or waterfall from a résumé? For agile, look for tools and methods that are commonly used: unit testing, continuous integration, frequent releases, *etc*. You can pick out waterfall when the candidate describes working in phases: "Approved requirements," "Participated in System Design," "Coordinated hand off of code to QA." These aren't certain indicators of waterfall, but you can often get a pretty good sense of the methods the candidate has used by how they phrase their accomplishments.

Change aversion is a powerful indicator that you can sometimes pick up from a résumé. I don't mean someone who jumps from job to job; that is usually a problem. Say someone has moved about the country a few times. That person is probably not change averse, whereas someone who never moved or has worked at the same company for ten years may struggle with change. There may be clues in the job descriptions as well. Has the candidate ad-

Countercultures

At one company, we noticed that candidates who had significant history working for non-profit or government organizations tended to find our environment threatening. Our environment was characterized by transparency, frequent releases, managed risk-taking, high communication and collaboration, rapid change, and high-energy.

We theorized that working in non-profit or government organizations was very different. These organizations tended to have infrequent releases, lack of urgency, high degree of process ceremony including complex approval requirements, and an aversion to risk and change. These characteristics were counter to what our culture valued most.

Whenever we reviewed résumés that implied the candidate was comfortable in these slow environments, such as through long, slow-growth service, we became skeptical. That said, some of our best employees had worked at non-profit and government jobs. What was the difference? They weren't comfortable there. From their résumés, there was no indication they were comfortable and, in some cases, their desire to escape was mentioned in the career objective or in a cover letter.

justed to changing technologies or roles? Has the candidate been successful in obviously different environments? Hopefully you can see that how a candidate handles change tends to correlate with other positive traits.

There are so many cultural and environmental considerations, it is impossible to list them all. Only some can be teased out of a résumé. Over time, you'll be able to predict which candidates are likely to work well and which are likely to struggle in your environment.

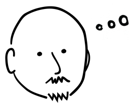
Experience

The experience perspective looks at the length of service in the industry and in the areas of interest to you. Decide in advance about how many years of experience you want for candidates. Don't be rigid, but have a notion of what you are looking for to provide a rough guide.

Consider breaking down years of experience on a per-skill basis. For example, if the job involves serious Java programming, you might want someone with at least four or five years of Java programming experience.

There is no need to specify years of experience for all the skills of interest. The more specific you get, the less likely you will find a match. Think of years of experience as a guide rather than a rule. The downside of years of experience is that it does not capture the candidate's skill in an area. Typically, a candidate will compute years of experience as the number of years from today back to the first time he used the technology. Candidates rarely account for the intensity of use during that period; there may even be large gaps during which the candidate did not use a technology at all.

An alternative that is more useful than years of experience is specifying a desired level of mastery in the skills you are seeking. This takes more work because your hiring team must develop a yardstick of specific skill levels to consistently measure candidates against.



Mastery is more important than years of experience.

When you count years of experience, do not count gaps where the candidate was not working. Check whether the candidate was attending school while working. If so, try to determine if the candidate was working part- or full-time.

Working part-time can sometimes be discerned by examining the responsibilities portion of the job description. Usually it is fairly obvious, because the nature of the work is light and low impact. I'd recommend discounting the value of part-time work by half. Full-time work while attending school is impressive and should be counted completely. Give credit for determination, hard work, and initiative.

I have already argued for hiring experience over potential. That said, in the search for experienced software professionals, an occasional gem of a person comes along whose potential hits you like a ton of bricks. Be flexible! There are many benefits to hiring an individual with great potential.

Employees with potential are unsullied by years of negative work experience. They are ready to learn, and you can train them in a fashion that will provide benefit to your organization. They bring high-energy and a fresh perspective. Some may be recent graduates who have learned new ideas in school that you can implement to give you a technology edge.

Detecting potential is much harder than analyzing experience. On the résumé, look for high levels of intelligence, achievement beyond years, rate of growth (or career trajectory, if there is enough of a career to analyze), good communication, and signs of "maturity beyond one's years." In later phases of the interview process, it is essential the high-potential candidate demonstrate a strong compatibility with your organizational values and behaviors. If they don't fit in, they will not develop as you hope.

When considering a candidate with potential, keep in mind the cost and risk of hiring him or her. The cost is in the mentoring and training, including the drag it will put on the mentors and trainers. The risks include possible failure of the candidate to reach potential, and the damage that can be caused by having to invest extra resources into the less experienced employee.

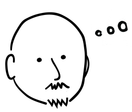
Keep in mind the organizational context. Do you have the organizational resources to support personnel development? Are teams already stressed to the point that the burden would bury them? Does the current distribution of staff experience argue for or against hiring less experienced people?

Performance

If you want to hire top-notch software developers, seek candidates that show a record of strong performance on their résumé.

Don't settle for people who merely make the grade. These résumés read like a series of inevitable promotions; responsibilities that tend to be somewhat repetitive and mundane for the years of experience. It is as if these people were a victim of their own career.

Mundane tasks are often the fate of the junior developer, so don't count mundane work early in a career against a candidate. In fact, you may find an otherwise good career blemished by a job that looks like nothing but tedious, low-skilled labor. As with nearly everything about hiring, there are exceptions. It could be this candidate was totally misappropriated by an employer. Maybe the candidate desperately needed a job in a difficult economy, or difficult personal situation. In these cases, you might see that the candidate was not in that position long and it was followed by a job with a much more impressive description.



Look for people who have shown initiative in taking their career somewhere.

Examine the career over time. Have the job titles tended to improve?

How rapidly have they improved? In each position, have the responsibilities expanded in difficulty and importance? If so, this person is growing and will continue to grow.

Seek evidence that the candidate has succeeded on complex projects or in challenging situations. Look for high-impact contributions that have improved the bottom line or have fostered important change.

Career trajectory

I refer to career change over time as *career trajectory*. The résumé of a candidate with significant experience will chart a trajectory. Think of each position as a point on a graph. Over time, the points are plotted and the next point will nearly always be plotted higher than the previous points. There may be an occasional dip because downturns happen, but it should be followed by a renewed upward trend.

If you draw a moving average line through the points, it will have an upward slope. Look for people with steep upward slopes, candidates whose careers are clearly on a strong upward trajectory.

As always, beware of news that appears too good to be true. You may find a candidate whose trajectory is radically steep. Look deeper. Career trajectory analysis should be coupled with an analysis of the length of the jobs among other things. Someone who has moved rapidly from job to job over a period of time may have accrued a very impressive set of titles. Look hard at the responsibilities and accomplishments, and keep in mind that someone who hasn't been in a position long probably hasn't demonstrated mastery at that level.

Be concerned about candidates who have spent too long doing essentially the same thing. This might take the form of being stuck in a role for too many years. It might take the form of doing the same kind of work across several jobs. Avoid candidates who appear to be doing the same thing in different jobs—their career is languishing.

In the first eight or so years of a career, about two years is enough time for a title change or a detectable change in responsibility. As the candidate achieves more responsibility, expect the role duration to lengthen. There are no hard rules, so just try to develop a sense of how good careers progress compared to mediocre ones.

It is common to see what appears to be a backward step in the trajectory of someone's career. "Two steps forward, one step back" may not be a bad thing, and really depends on the circumstances and what else you can glean from the résumé.

I have seen many résumés where a candidate achieved a leadership role only to be followed by a non-leadership position change. Many times the candidate was not ready for leadership or had decided they did not want leadership. Since I recommend hiring for leadership, this is something to explore, but it isn't a résumé deal breaker. If the candidate wasn't ready, but still has aspirations to lead, this can actually be an opportunity. I have found that these individuals do well on their second try. If the candidate has decided leadership is not in their future, you will have to decide whether that person will fit in your organization. There is nothing wrong with having strong developers who never take on a leadership position—especially if you have a strong technical career track. And, there are other kinds of leadership that the candidate may be eager to explore.

Another example is a situation in which the person has changed companies, and it appears from the job title that they have been demoted. It is unlikely you'll be able make anything of this from a résumé unless the candidate provides hints. The "demotion" might simply mean the new company had fewer job titles or a higher bar for that title than the preceding company. It might be that economic challenges forced the candidate to accept a lesser title. Or it could be that the lesser title was offset in the mind of the candidate by other benefits of the new role. It's usually impossible to draw conclusions from the résumé. If the answer might matter, and the candidate makes it to a phone screen, then this question should be entered into the tracking system and asked in the phone screen.

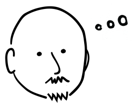
Growth

Growth is different from career trajectory. This perspective attempts to identify how the candidate approaches personal growth. Growth is a vital trait in a software professional, because our market and technologies change so rapidly. If you plan to keep employees around, you want them to be dedicated to keeping relevant and useful.[5]

[5] The organization that encourages growth in partnership with employees is more likely to achieve long-term retention. This includes training, providing books and other educational materials, mentoring, and appropriate job challenges.

Look for any signs that demonstrate the candidate can learn and grow. Does the candidate have a history of adopting new languages and technologies? Has the candidate used different development methodologies? Have job changes required the candidate to learn new domains? Have new roles required the candidate to improve his collaborative skills?

Look for signs of deep growth or understanding in some areas. We can't be deep in everything, but most strong candidates develop depth in a few areas. Absence of depth is a big concern because the person may be proficient at quickly learning just enough of something to wreak havoc in its use. On the other hand, one who has depth of skill is more likely to wield that skill with wisdom and restraint.

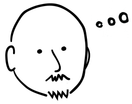


You want to hire people who have the capability of becoming really good at something.

Try to figure out if the person is into languages, algorithms, design, process, methodology, communication, *etc.* If you can decode a candidate's growth interests, consider how these match your needs. Whether they match is not a litmus test; think of it as a bonus if they do.

Be mindful that people's growth interests change over time. It is common to see people's interests out of school aligned with more academic topics like languages, AI, or algorithms. Over time, the forces of their work life tend to influence their interests more. Someone who shows changing interests over time—accompanied with depth and practical application—are a great find, because this indicates she can repeat the process of growth.

Look for the methods the candidate has applied in an effort to grow. Does the person seem to be an independent learner or require a learning regimen? Just because the candidate has a lot of formal education and training does not mean he cannot learn independently. What you are likely to discover on a résumé is probably incomplete and inferred; it is evidentiary data that can prime the questioning on a phone screen. At some point, judge how well the candidate's growth style matches the way the people in your organization grow. This inidicates how successful the candidate might be if hired.



A person who takes the initiative to grow is a person who values adaptability. This is a value trait that you want to hire for.

Leadership

If you are hiring candidates that you intend to retain as long as possible, hire with leadership in mind. Even if you are not trying to fill a leadership position now, seek candidates that demonstrate leadership ability. Because a long-term employee will eventually be faced with a leadership role, leadership capability should be a core concern.

You want to hire people who will grow to lead others, but keep in mind that leadership takes many forms beyond just leading people. In the résumé, look at the experience stories for explicit or implicit evidence of leadership. Mentoring is an important early leadership role. Look for participation in standards bodies, expert groups, open source projects, improvement efforts, community efforts, scholastic societies, clubs, *etc*.

If a résumé lists staff leadership, such as team or group management, or claims leadership as a skill, that candidate probably desires to be a leader. Look out for the *two steps forward, one step back* syndrome. Many candidates have become a leader only to later move back into technical roles. This is neither bad nor good, but must be explored if the candidate makes it to the next phase of the hiring process. People drop out of leadership for many reasons: not enjoying leadership, preferring to remain purely technical, having a bad experience, moving into a new situation where leadership is not an option, realizing they are not ready or not yet good leaders.

Often people who have dropped out of leadership want to try again. Unless you ask, you won't get the whole picture. These discussions tend to provide deep insight into the candidate, so seek them out. Highlight these observations in the tracking system so that they can be discussed should the candidate be phone interviewed.

A candidate of around five years of experience ought to show some signs of leadership, although not necessarily people leadership. After eight to ten years, most software professionals have led people on some sort of project. If you don't see some evidence of leadership in these cases, be concerned. The candidate may be averse to leadership, or worse, may not be equipped with the skills and behaviors of a leader. Many of these skills and behaviors are those you seek in a developer: communication, collaboration, judgment, and initiative, to name a few.

There may be cases where leadership is secondary to immediate need, but be careful. Every person hired who is unable to lead will place a long-term burden on the organization. It will become harder to provide a place for employees who cannot take on some type of leadership responsibility.

Impact

The impact perspective seeks evidence that the candidate has made a mark in past jobs. In the experience stories, look for tangible evidence that the candidate has made a significant positive impact.

Does the candidate claim she made contributions to the bottom line or to major cost savings? Has the candidate made significant process improvements, or championed organizational or technological change? Does the candidate seem to participate in successful projects? Are the projects relevant? Does the candidate have a central role? Do good things seem to happen around this candidate?

Conversely, learn to recognize ineffectual candidates. These tend to list a lot of minor tasks and few results. You will not build excellence with ineffectual employees.

Potential

Potential is a very important, but difficult perspective to evaluate. For the candidate with several years or more of experience, you can get a feel of the trajectory of the career and extrapolate. For the more junior candidates, potential is very difficult to determine, but for this class of candidate, a major part of what you are banking on.

For those with a few years of experience, look for completion of tasks that seem beyond the candidate's years. Look for early successes that made a positive impact. The good news should have started in school and continued right into the career. Look for awards and other recognitions. Theses, publications, and open-source projects can demonstrate potential.

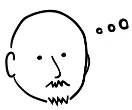
Nothing beats a proven, experienced candidate, but if there are mitigating circumstances (weak market, salary constraints, *etc.*), consider stepping down expectations and focusing on potential.

When hiring the best and most experienced, be opportunistic when it comes to potential. If a résumé of a candidate with less experience comes along that shows impressive potential, consider moving it through to a phone screen. Consider what your organization can bear regarding less senior candidates before doing so.

> **Seeing in scales**
>
> The opposite of potential is lack of promise. This often shows up on the resume as a career of stagnation and mediocrity. For every positive you learn to recognize, learn also to recognize the opposite. People are complicated; their skills and behaviors fall on a scale of possibilities. Hone your skills so that you can picture perspectives and attributes on a scale for different perspectives.

Education

The experience and education perspectives have an inverse relationship: with more experience, education becomes less important; with less experience, education becomes more important. Take care in determining the value to be placed on education for each candidate. Err on the side of better education if you are unsure, but understand that experience washes away the past. A strong educational foundation is an indication the candidate has a solid base (although it is not a guarantee).



Education is important, but quality experience is more important.

Research the education institutions mentioned and understand what they offer. Although many quality universities exist, including "name brand" universities we all recognize, many others exist that range from mediocre to downright unscrupulous. If you are unfamiliar with a school, look it up on the Internet; go to its Web site; browse around. Check out the curriculum and the faculty. You might discover the candidate received a degree online[6]

[6]There are an increasing number of quality online programs being rolled out by traditional institutions. When in doubt, research the school.

or from a trade school. These candidates won't have depth in their craft and may not even have more than cursory skills.

Foreign institutions can be difficult to evaluate. Again, use the Internet, but also speak with colleagues or friends from the country in question. Most countries do not enjoy a large number of good universities. India, for example, has Indian Institutes of Technology (IIT) and a few others. There are many Indian education institutions, but the vast majority offering computer-related degrees are like junior colleges or trade schools.

Get a sense of the quality of the technology programs at different institutions. Annual rankings for university programs provide this information for US schools. Look at the research efforts, industry collaboration, *etc.*

Pay attention to the type of degree earned. Computer science (CS) degrees from accredited universities tend to be most valuable; the better the program, the more valuable. Be wary of information technology (IT) degrees, these are usually either business related or are offered by institutions that cannot muster a full CS program. Also, be wary of things like application development degrees—a sign of a trade program.

Look for candidates with advanced degrees in computer science. Advanced degrees indicate a significant commitment to the profession. Many candidates will grind through a bachelor degree just to get a job, but few will work through secondary degrees. As before, be conscious of the institution.

If the candidate does have an IT degree, or some other non-CS computer-related degree, research the curriculum. Often these degrees are very easy to get, especially outside the U.S. On the other hand, they may provide the specific skills you need, especially coupled with relevant work experience.

Decide how to deal with candidates with non-CS degrees. This should not be a showstopper if coupled with strong experience. Consider the type of degree—some are closely related to CS. Electrical engineering and other engineering degrees develop a mathematical and reasoning basis similar to CS, likewise with mathematics and meteorology. Surprisingly, I know of industry leaders with sociology and English degrees. Lean heavily on candidate experience in these cases.

Look for honors and extracurricular activities. Candidates rarely provide grade information, but will state that they were Magna Cum Laude if they were, or say nothing if they weren't. The value of such honors rises with the quality of the institution. Look for technical- and science-related activities and societies. This demonstrates leadership and an interest in the craft, which is extremely valuable in a candidate.

Problem detection

This perspective explores possible problems in a résumé. Think of problems as red flags. As you review the résumé, tally the red flags. Too many red flags may be sufficient to eliminate a candidate. A few red flags may not be that serious if they are counterbalanced by positives in the résumé.

As you explore a résumé with the other perspectives in mind, you will inevitably find and note problems. The difference with this perspective is that you are actively seeking specific problems.

Keep in mind that certain problems can compound each other by building evidence of a pattern of behavior. As you gain more experience, you will begin to recognize themes, both positive and negative. As you become more accomplished, rather than mindfully examining each perspective individually, you'll begin to intuitively assemble and adjust a mental model of the candidate. Be patient though; this Ri-level ability takes time to develop.

If it turns out a phone interview is in order, certainly explore every potential problem to determine if it is an issue. The remainder of this section focuses on some key problem areas that are often easy to detect in a résumé.

Job hopping

Job hopping is a tendency to move from job to job, not staying at one place very long. Job hopping demonstrates an unwillingness (or inability) to remain in a position for any length of time. If you are hiring for the long term, you are unlikely to be able to retain a job hopper. When the job hopper walks out the door, you lose all the local knowledge he has accumulated as well as the significant investment getting him up to speed.

As always, be careful about labeling a candidate a job hopper without considering other factors. The candidate may have been a contractor for long periods of time, or had other demands such as attending school while trying to earn a living. The candidate may have been subject to a series of difficult situations beyond her control such as layoffs or business failures. The candidate may even have had a strategic reason for job hopping, such as getting exposure to different domains early in a career.

There is no clear rule for determining if a candidate is a potential job hopper. Every job transition is worth discussion, but here you are looking for problems related to excessive rate of movement. In the software industry, long-term employment is the exception rather than the rule. I don't have

A case of premeditated job hopping

We were reviewing a candidate who had somewhat fewer years of experience than we wanted, but several mitigating circumstances led us to believe he was a low experience/high potential candidate worth interviewing.

One thing that was troubling was the high rate of job hopping. We could see that the candidate was attending school for much of the period of hopping, but not all of it. On the call, we asked about it.

Indeed, there was some moving around to compensate the educational demands. When asked about the hopping after graduation, the candidate gave us a surprising explanation. He told us the changes were intentional. He was exposing himself to a range of domains and environments so that he could decide where he wanted to establish a long-term career. He had settled on e-commerce.

This is a good example of two justifiable reasons for job hopping. It also shows why it is important to think beyond rules, weigh all the information together, and respect your hunches when you think someone is a winner even though they don't fit the mold.

actual data, but there is an interesting topic at *Joel on Software Discussion Group* in which developers shared how long they stayed at jobs.[7] Both the numbers and the discussion are interesting. Without doing the math, it seems about two to three years are the average according to this unscientific survey.

I focus on the duration of several of the most recent jobs, looking for the trend. I am not very concerned about job hopping early in the career if it has not happened lately.

For full-time work, durations of less than two years should be scrutinized. The shorter the duration, the more weight it should be given. One short stint by itself doesn't deserve more than an inquiry on a phone screen, but a series of short stints may be a problem pattern.

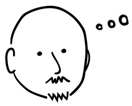
Be sure to document your job hopping concerns, so that if the candidate is invited to a phone interview, people will know to focus on the suspect jobs. There may be good reasons, but get the facts. If there aren't good reasons,

[7] "What's your average length of time at any one place?" [Spo08]

this is a serious problem.

The guidelines for contracting jobs are different than for full-time jobs. Typically, there will be some indication if the jobs are contract jobs. Contract jobs are often in three-month increments—three, six, nine, or twelve months—so this may give you a hint if nothing is called out.

It is important to understand the candidate doesn't have much influence over the duration of contract jobs. Therefore, job hopping may not be easy to detect in contractors. For contractor jobs, I feel very little can be accomplished or learned on a three-month project, so I put little value in those types of projects. Because so much of that time must be spent getting up to speed, there is very little time to grow in skill. Furthermore, consider that shorter contracts don't allow the contractor to experience the results of their technical decisions; they don't get as much feedback for learning. I don't consider these three-month stints job hopping per se, but I don't value them as much for learning. For example, four three-month jobs are not equal to one twelve-month job. Put greater value on longer projects and focus on those accomplishments.



Longer engagements give more time for learning and growth. In short engagements, the employee often doesn't have to suffer the results of his decisions.

Six-month contracts are just long enough to get the benefit of growth. Nine-month, twelve-month, or longer-term contracts are good opportunities for growth within a contract employment situation. Remember that contracts can be terminated at anytime, so the longer the duration, the more likely the employer was pleased.

I get concerned if a contract is less than three months. I may note if a contract seems to end on a strange duration boundary; it is possible the contract was terminated abnormally. A series of short stints is always a concern, but it may not be job hopping. The contractor may have simply worked for a company that only signs short contracts.

When looking at a contractor résumé, it takes some patience to tease out continuity. For example, there may be a series of projects listed that appear independent, but on careful examination you see they are different phases of a larger project within the same team. Give credit for the aggregate work on the larger project, because the contractor is working in the same environment and opportunities for growth span the entire duration.

Be careful when considering a candidate with a recent history of contract work. It may be the candidate is tired of contracting and wants to settle down. Getting contractors "on the rebound" is usually good, because you get the benefit of a broad range of experience plus you hire a person who's lived the contractor life and chosen to leave it for stability.

On the other hand, contractors will sometimes use full-time employment to tide them over between contract stints. In tough economic times, for example, companies cut back on contract work, so contractors seek safe-havens until the economic situation improves. Sometimes you can see evidence of this behavior on the résumé where a series of contract jobs are interrupted by a fairly short, full-time job followed by more contract work.

The rule[8] with contractor résumés is that if you decide to do a phone screen, make sure to explore the reasons why the candidate is considering a change. Keep in mind the economic and market forces. Listen for cues that the candidate might be under some financial stress and might be seeking a period of stability. Make it clear you are looking for long-term employees and ask directly whether that is what the candidate is seeking.

Résumé 3 is an excerpt containing just the job history headings. One thing that bothers me about this résumé is the absence of location information. If the candidate is moving through jobs fairly quickly and is jumping locations, that increases my concern. In this example, we have job durations of: 2.5 years, 1.5 years, 1 year 2 months, 2 years, 8 months, 9 months, 1 year, 8 months, 1 year, and 10 months. With the exception of one job, these are full-time positions. Not only does the candidate fail to stay in a job, the candidate also switches business areas.

I would consider this candidate a solid job hopper that is unlikely to stick around. The résumé would have to be very strong for me to consider a phone interview, and if so, the top issue to resolve would be whether the candidate was planning to settle down.

Stagnation

A healthy career is portrayed in a résumé by a progression of responsibility, skill mastery, accomplishment, and job titles or positions. Evaluate the scope of work described over time and the role the candidate played in that work.

[8] I do mean "rule."

EXPERIENCE

Senior Software Developer, Radio Company Inc.	02/2008 – 12/2008
Technical Team Leader & Project Manager, CRM, Inc.	12/2006 – 01/2008
Senior Software Developer, Some Health Group	01/2006 – 09/2006
Software Consultant, Company 7	11/2005 – 11/2006
Senior Software Developer, Some College	03/2005 – 12/2005
Project Manager, Company 5 Inc.	05/2004 – 01/2005
Technical Team Leader, Company 4 Inc.	12/2001 – 12/2003
Senior Software Engineer, Company 3 Inc.	08/2000 – 10/2001
Software Engineer, Company 2 Inc.	01/1999 – 07/2000
Software Engineer, Company 1 Inc.	05/1996 – 12/1998

Resume 3 · Excerpt containing just job history headings.

If the candidate seems to be working on the same thing over and over again, then stagnation is a concern.

Try to get a sense for the magnitude of responsibility and accomplishment. Be cognizant of the technology mix the candidate uses over time. Is it always the same or does it change over time? Change over time indicates flexibility and growth.

Are the technologies outdated? Why is the candidate still dabbling in COBOL? Often valid forces exist that prevent the candidate from growing. Of course it raises the question why someone would remain employed in such a situation; the candidate may have poor judgment.

Be suspicious of candidates who have had the same job title or position for too long. It is reasonable to expect a title change every two to three years during the first eight to ten years of a career. As always, there are mitigating circumstances. For example, some companies (usually startups) feel a hierarchy of development titles is counter-productive. If the candidate reaches a phone interview, ask about these issues.

Sometimes stagnation can be difficult to spot. Consider a résumé where the candidate has a few jobs with the title of "Developer," then a fairly short stint as "Team Lead" or "Architect," then back to another job titled "Developer." You should wonder if the candidate was qualified for the job. You might also wonder whether they had problems being a leader. On the other hand, there are reasonable explanations for these "up and back down" situations. If you see it on a résumé, be concerned and note in the tracking system that it should be explored.

The following, rather long, job history is interesting for several reasons:

Experience

Software Engineer, Company 6, UT **10/07 – 2/08**
Responsibilities:

- Involved in design and development of **Servlets and JSPs using Struts framework**.
- Involved in the Design and development of Domain Object Model and KTMI Mappings.
- Using **Spring** framework handles all injections and transactions in the application.
- Developed **DAO** object to perform generic database operations using hibernate mappings in application container.
- **Hibernate** Tools were used as persistence Layer - using the database and configuration data to provide persistence services (and persistent objects) to the application.
- Used **Junit and Jmock** for testing the application components.
- Design and development of integration tests to check the functionality of various Components of ECIS application.
- Involved in the **Web Services** development using **Xfire** and using **Soap UI** for testing the **WSDL** Requests.
- Utilized **Log4J** for debugging and logging and used Maven to build the deployment **JAR and WAR** files.
- Created **HTML, CSS, and DHTML** pages for front-end.
- Using **IDE Eclipse 3.2** to develop the Application.
- Used **Struts & JavaScript** for server/client-side validation.
- Used **Clear Case** for the Version/Source control management.

Sr. J2EE Developer, Company 5, MA **4/07 – 11/07**
Responsibilities:

- Involved in design and development of **Servlets and JSPs using Struts framework**.
- Configured the **WebLogic 9x** and deployment descriptor.
- Used **Spring** framework to handle all requests to the application.
- Used **JDBC, Data Sources and Connection Pooling** in Application server **to interact with the NetEzza**.
- Developed **Generic DAO** object to perform generic database operations using data source **JNDI** name defined in application container.

- Implemented J2EE Design Patterns such as Session Facade to reduce the Network Traffic, Service Locator, and Transfer Object for Lookups, Value List Handler for search and DAO.
- **Hibernate** Tools were used as persistence Layer - using the database and configuration data to provide persistence services (and persistent objects) to the application.
- Used **Jasper Reports** for report generation.
- Utilized **Log4J** for debugging and logging and used **Ant** to build the deployment **JAR and WAR** files.
- Created database including tables, views, functions, constraints etc.
- Created **HTML, CSS, and DHTML** pages for front-end.
- Used **Struts & JavaScript** for client-side validation
- Used Eclipse as a IDE, Apache as a web server and Weblogic 9x as an application server, and NetEzza as the database to develop and deploy the **RCOM** application.
- Used Clear Case for the Version/Source control management and Clear Quest for creating Deployment as well as testing tickets.
- Used ANT and UNIX Shell programming to write build scripts as well as deployment scripts.
- Conducted extensive **functional and unit testing**.
- Provided production support by interacting with the end-users and fixing bugs.

Sr. J2EE Developer, Company 5, MA **11/05 – 3/07**
Responsibilities:

- Involved in design and development of **Servlets and JSPs using Struts framework**.
- Configured the **WebLogic 8x** and deployment descriptor.
- Developed **Generic DAO** object to perform generic database operations using data source JNDI name defined in application container.
- **Hibernate** Tools were used as persistence Layer - using the database and configuration data to provide persistence services (and persistent objects) to the application.
- Used **Jasper Reports** for report generation.
- Utilized **Log4J** for debugging and logging and used Ant to build the deployment **JAR and WAR** files.
- Created database including **tables, views, functions, constraints** etc.
- Created **HTML, CSS, and DHTML** pages for front-end.
- Using **IDE Eclipse 3.1** to develop the Application.

- Used **Struts & JavaScript** for client-side validation
- Used ANT and UNIX Shell programming to write build scripts as well as deployment scripts.
- Conducted extensive **functional and unit testing**.
- Provided production support by interacting with the end-users and fixing bugs.

Sr. J2EE Developer, Company 4, AZ **6/04 – 10/05**
Responsibilities:

- Involved in design and development of **Servlets and JSPs using Struts framework**.
- Configured the **WebSphere Application Server 4x** and deployment descriptor.
- Used **JDBC, Data Sources and Connection Pooling** in Application server **to interact with the DB2** .
- Developed **Generic DAO** object to perform generic database operations using data source JNDI name defined in application container.
- Implemented J2EE Design Patterns such as Session Facade to reduce the Network Traffic, Service Locator, and Transfer Object for Lookups, Value List Handler for search and DAO.
- **Hibernate** Tools were used as persistence Layer - using the database and configuration data to provide persistence services (and persistent objects) to the application.
- Test Cases and Execution, Preparing Test Data and workflows for Business Requirements.
- Developed WebServices to publish and developed web Services Client to access published Webservice.
- Created database including **tables, views, functions, constraints** etc.
- Created **HTML, CSS, and DHTML** pages for front-end.
- Used **IDE WASD** to develop the Application.
- Used **Struts & JavaScript** for client-side validation
- Used Eclipse as an IDE, Apache as a web server and WebSphere 5.x as an application server, and DB2 as the database to develop and deploy the **e1 Management System** application.
- Used ANT and UNIX Shell programming to write build scripts as well as deployment scripts.
- Conducted extensive **functional and unit testing**.

J2EE Developer
Company 3, Africa **6/03 – 5/04**
Responsibilities:

- Designed and implemented Struts (MVC Paradigm) components such as Action Mapping, Action class, Dispatch Action class, Action Form, DynaAction Form, Validation Framework, Struts Tiles and Struts Tag Libraries.
- Extensive use of HTML/Servlets/JSP/XML for the presentation layer along with JavaScript for the client side validations.
- Participated in all the phases of the development namely Web Component Creation, Enterprise Bean Creation, Application Assembly, Application Deployment, and maintenance.
- Developed Web-tier authentication consisting of HTTP basic authentication, form-based authentication, and HTTP mutual authentication.
- Used JDBC to connect to the Database, and JNDI to lookup data and EJBs.
- Extensively implemented Stateless beans to capture user conversation and to achieve scalability, memory management and transactions, and Entity beans to bring the relational data to the object world.
- Implemented J2EE Design Patterns such as Session Facade to reduce the Network Traffic, Service Locator, and Transfer Object for Lookups, Value List Handler for search and DAO.
- Used Eclipse as a IDE, Apache as a web server and **JBoss 3.2.5** as an application server, and Postgres Sql as the database to develop and deploy the **WebPay** application.

J2EE Developer, Company 2, MD **4/02 – 5/03**
Responsibilities:

- Involved in designing the views using Dream viewer Editor.
- Designed and implemented Views using JSP.
- Extensive use of HTML/Servlets/JSP/XML for the presentation layer along with JavaScript for the client side validations.
- Used EJB Session Beans for implementing the Business Logic for the Arida.
- Participated in all the phases of the development namely Web Component Creation, Enterprise Bean Creation, Application Assembly, Application Deployment, and maintenance.
- Developed Web-tier authentication consisting of HTTP basic authentication, form-based authentication, and HTTP mutual authentication.
- Extensively implemented Stateless and State full Session beans to capture user conversation and to achieve scalability, memory management and transactions, and Entity beans to bring the relational data to the object world.
- Used Eclipse as a IDE, Apache as a web server and WebLogic 7.x as an application server, and Oracle8i as the database to develop and deploy the Arida application.

- Used **Visual Source Safe** for the Version/Source control management and Clear Quest for creating Deployment as well as testing tickets.

J2EE Developer, Company 1, NY **5/01 – 3/02**
Responsibilities:

- Involved in designing the views using Dream viewer Editor.
- Designed and implemented Views using JSP.
- Extensive use of HTML/Servlets/JSP/XML for the presentation layer along with JavaScript for the client side validations.
- Participated in all the phases of the development namely Web Component Creation, Java Bean Creation, Application Assembly, Application Deployment, and maintenance.
- Used JDBC to connect to the Database, and JNDI to lookup data sources and EJBs.
- Used Eclipse as a IDE, Apache Tomcat as a web server and Oracle8i as the database to develop and deploy the LSRC application.
- Test Cases and Execution, Preparing Test Data and workflows for business requirements.
- Used Visual Source Safe for the Version/Source control management.

The first point to note is that the candidate seems to be doing the same things over and over. In fact, many of the bullets are cut and pasted from one job to the next. The second point is that most of the tasks are mundane and many are trivial.

The third point to consider is that after seven years, the candidate is still doing mundane tasks. There doesn't seem to be any progression at all. Finally, the candidate has chosen to list items in the responsibility section that are not responsibilities, such as which editor was used, or which source-control tool was used. This shows a serious lack of judgment. The entire résumé was about nine pages and it should have been about three or four.

Leadership aversion

Some developers are averse to leadership of any kind. Leadership aversion is often associated with other problems, so regardless of how you feel about hiring developers with leadership potential, you should care. Leadership-averse people tend not to possess good people skills, communication skills, or judgment. They often just want to be left to solve programming problems. Most organizations cannot support these employees long-term. You cannot hide people like this in a collaborative organization. Furthermore they stress your leadership supply chain.

On one hand, a candidate can have a complete career in technology without ever becoming a manager. On the other, it is unlikely a talented technologist can have much of a career without becoming a technology leader of some sort. If your organization has a strong technology track for employees, you can tolerate people who do not have a desire to lead others. That said, senior technologists still have leadership responsibilities in communicating and promoting technologies, ideas, methods, architectures, and the like. A senior technologist without leadership skills is a drag on an organization.

If, after about seven years of experience, a candidate has never led at least a small project, that should be reason for concern. Likewise, a candidate who held a leadership position in the past, but not recently, may be leadership averse.[9] Review the section on leadership. If the résumé doesn't have any indications of leadership, that could be a problem.

Be concerned if the candidate appears to have dabbled in leadership and then settled into a career without leadership. Another possible indicator is repeated "lone-wolf" projects. Isolated developers fail to learn leadership skills like developers in collaborative environments do.

Apparent leadership aversion in the résumé may not be a problem. In interviews, explore the candidate's career plans in this regard to gain understanding. Match that with the requirements of the job position and the current and future organizational needs. Explore situations where the candidate has moved into leadership and then backed out.

Education issues

Be very concerned by any candidate that has no college degree of any kind. In the absence of a strong job history, this alone may be sufficient to reject a candidate. Often candidates that have no training in computer science or programming will have large holes in their understanding. They may lack foundational information that is so helpful in solving difficult problems.

Some candidates state possession of degrees but do not state the institution or the type of degree. As with any standard résumé information, absence may indicate a problem. This typically happens either when mentioning the institution would not reflect well on the candidate, or when the candidate is

[9] Although be mindful of the "two steps forward one step back" discussion earlier in the chapter on page 70.

not a citizen and assumes you would not recognize the institution or that you would not care. It usually boils down to poor judgment in leaving it off or an attempt to suppress unflattering information.

Less concerning, but sometimes a problem is when the candidate attended the listed institutions. It is helpful to know how time spent learning relates to time spent pursuing the career. Attendance dates are helpful, but are often left off. I tend not to be too concerned unless there are other issues that seem to be related to information hiding.

Degrees in information technology,[10] or associate degrees, degrees from community colleges, technical colleges, *etc.*, should be accompanied by a strong job history. For lower-level positions, this may not be a serious concern, assuming the candidate possesses offsetting positives.

Surprisingly enough, great care should be taken with candidates who possess a PhD—even in computer science (CS)—in the absence of a significant work history post-PhD. Earning a PhD in any discipline is a major accomplishment, yet I have observed several issues that seem to arise.

People with PhDs have made a huge investment in a narrow area and typically want to leverage that investment. Given the effort required to get a PhD, an interviewer should always ask what the motivation was. The major reasons are:

- To teach: Often people decide halfway through they don't really want to teach, but they show the initiative to finish the degree anyway.
- To become an entrepreneur: Maybe it didn't work out. Maybe it was a bad idea. Follow up and question judgment.
- To get a higher salary: Not a good investment. Question judgment.
- To stay in school: Perpetual students are not always well suited to the professional world.

Pay special attention to behavioral traits of candidates with PhDs. Sometimes an unearned arrogance or superiority can accompany the degree. There may be an attitude of entitlement without sufficient experience to back it up. Be on the look out for these behaviors. Try not to confuse arrogance with confidence; earning a PhD can certainly build confidence.

[10] As opposed to a computer science degree.

All this said, earning a PhD is a significant accomplishment. With a few successful years of employment post-PhD, chances are the potential concerns listed here are not an issue. Additionally, if the candidate's thesis happens to be related to the job requirements, there may be a good match.

I have found that candidates with a PhD from a non-CS background and strong work history are often a great catch, especially if the degree is complimentary to CS. These people are experts in *some* field, but not in your field, and they know it. Their desirability is supported by the idea that mastery is transferable. Ensure the evidence is present that the candidate has begun to master areas of interest to you.

Red, yellow, or green flag?

Sometimes it helps to tally the problems you detect. By now it must be clear that few things in résumé reviews are black and white. We deal with partial information and facts that are incredibly diverse. Some problems are very clear and serious; I call these *red flags*. One or two red flags are serious enough to cause you to reject the candidate.

A *yellow flag* is a problem or possible problem; take the time to investigate if the candidate gets to a phone interview. Several yellow flags, maybe with a red flag, could be enough to reject the candidate.

Problems can sometimes turn around and become *green flags* if the "problem" was misinterpreted and on further investigation you learn something positive.

This is just one way to tally what you see. Keep track and review what you find, because often problems (or positives) reinforce one another and form a pattern that tells a story about the candidate. As you gain experience, you may pick up how certain problems tend to associate with certain behaviors or character flaws. You may feel like you are becoming an amateur psychologist after a while! When you are unsure, give people the benefit of the doubt. Try to track the candidate through the process to see if your diagnosis is correct.

Buzzword bingo

A common résumé style includes a skills section where the candidate lists technology experience. As mentioned before, this section often has two purposes: 1) to provide human résumé readers and filterers with a list of technologies the candidate deems relevant, and 2) to provide automated résumé filters the keywords they are searching for.

How candidates address the skills section varies. Some people show restraint and only list things they have relevant experience with. At the other extreme are people who list all the technologies used in their general vicinity for all time! Most résumés fall somewhere in the middle.

Scan the buzzword list for technologies of interest, but always seek evidence of mastery in the employment history. If there is an apparent mismatch, be concerned. Since people tend to pad this section, if it doesn't overlap well with your needs it is very unlikely the candidate will be a fit.

On some résumés, the list of buzzwords gets so large as to raise suspicion. Use your intuition here as there's no good way to know how much is too much. Too much is an indication of a résumé that may be less than accurate. In the presence of other concerns, this can be a big problem.

My experience is that strong developers can maintain mastery in only a few technologies and useful competence in less than ten. I doubt I have ever seen a buzzword section that listed only ten technologies!

I tend to put little weight behind the skills section unless it is at either extreme. Even then, by itself it is not very significant. When considering other problems, be concerned with a disingenuous and excessive use of buzzwords. There might be a general trend of truth-stretching or outright falsification.

If a phone interview is scheduled, take extra care to explore just how much experience the candidate has in your areas of interest. Be very wary if you discover a "truth stretcher." This behavior leads to trust issues that damage team dynamics.

A buzzword section is normal. Buzzwords accompanied by lack of deep knowledge anywhere are a big problem. A few examples of truth stretching in a résumé are very serious.

Unbelievable accomplishments

Another form of potential truth stretching is unbelievable accomplishments. Unlike buzzword bingo, the listing of unbelievable accomplishments is both easier to spot and more serious. Presumably the candidate put significant thought into the way the accomplishments were written, so if they are unbelievable—there is a more serious problem.

Sometimes the accomplishments seem hard to believe, but you can't decide with much certainty. This is where you turn to other parts of the résumé—like the buzzword usage—to see if there is a pattern of reinforcement. If there doesn't seem to be, then note your concerns but don't weigh them too heavily; leave it for the phone interview to resolve.

Remember that the candidate is trying to put the very best light on his or her accomplishments. There is a temptation to embellish. Since honesty is a key value, truth stretching should always be a concern. Do not be surprised to see this or even outright lying. Use good judgment when evaluating whether the candidate is merely putting a little lipstick on the pig or whether he or she is telling you the pig is really a unicorn.

Use common sense in evaluating the candidate's claims. I once saw a résumé in which a candidate claimed to play a significant role in ten major projects over an eight-month period! Do the math on the dates to see if the story hangs together.

Another common example of unbelievable accomplishments is candidates who claim ownership of tasks they did not play a primary role on. This can be exposed on phone interviews, but can usually only be detected if there is some sort of tip off. The candidate may claim ownership of too many parts of a project, or parts that are too large. The candidate may obfuscate the fact that she did not work alone, or only played a supporting role.

Check whether the candidate's accomplishments seem plausible for his years of experience. After participating in the hiring process for a while, you begin to get a feel for what someone with two, five, or ten years of experience is capable of. Take note if something seems out of whack and explore it on the phone interview.

Usually a truth-stretcher is consistent, so you should see signs throughout the job history and elsewhere in the résumé. Honesty is a key behavioral value, so if you are confident you have someone who is not truthful, weigh this problem heavily.

Sloppy résumé

Everyone knows the game: the résumé is the candidate's primary selling tool and usually sets the first impression with the potential employer. So why would an intelligent, thoughtful person present a sloppy résumé? The answer is: they wouldn't.

> **Stereotypes lead to bad hiring**
>
> You might be inclined to cut a developer some slack on the quality of a résumé. All the outdated stereotypes apply here: "Developers are anti-social," "Developers are poor communicators," "Developers don't need to communicate as long as they write good code." Buying into these stereotypes leads to bad hiring. Communication, both oral and written, ought to be considered core competencies for all software developers.

There are many resources on the Internet and at universities for writing excellent résumés. Placement agencies are will help if they are representing a candidate. Spelling and grammar checkers are ubiquitous. Even if the résumé language is a second language, there are plentiful options. There is no excuse for a sloppy résumé. Likewise, never assume there's an untold story or the candidate has somehow left out important information.

Why might a résumé possibly be sloppy or incomplete?

- Carelessness
- Inexperience (*E.g.*, the candidate didn't research how to write a good résumé.)
- Ignorance (*E.g.*, the candidate didn't do the research.)
- Lack of judgment

You don't want someone who exhibits any of these traits working for you. The résumé is the most important tool a candidate has. If the candidate can't do a quality job on the résumé, chances are the candidate won't do quality work either.

Be fair about what you deem a sloppy résumé. Keep personal preferences on style and formatting out of your thinking. Focus on bad grammar, misspellings, vagueness, and unintelligibility.

I am usually quite concerned if a candidate misspells or incorrectly capitalizes technology names. One isolated incident may not be a concern but repeated technology misspellings draw into question just how much experience a candidate has with the technology. One who uses a technology regularly and actually studies it should encounter the name often and should be able to spell it.

Take sloppiness seriously. People who are careless communicators, or use poor judgment, are not good hires.

Employment gaps and overlaps

Always scan the job history dates looking for gaps and overlaps. Gaps and overlaps may not be an issue, but they may indicate something serious. By themselves, date irregularities may not reveal their nature, but you can often find clues elsewhere in the résumé that tell the story.

If you find a job history gap, look for clues on what the candidate was doing during that time. Look at the education history to see if the candidate was attending school. Even without education dates, if the candidate has a Master's degree and the gap is about two years, this might reduce concerns.

Sometimes people have gaps in their résumés because they were working in a different type of job. I have seen résumés where the candidate tried real estate or Forex trading for a while. Some candidates have become entrepreneurs in unrelated businesses for a time. It is common for female candidates to have gaps where they may have had a baby. You have to run the different scenarios through your head to see what might be going on.

Some women (and increasingly men) take a few months maternity or paternity leave, and this is not an issue at all. Some may take extended leave or even drop out of the ranks of the employed for several years. Any gap of that length should be a concern, the larger and more recent the gap, the bigger the concern. Caretaking of children or the sick or elderly, is an extremely important personal responsibility. Nevertheless, from the potential employer's perspective, it is like the candidate took a job in a different field for a while.

The problem with taking a few years off is that the candidate may not be technically current. Look at what the candidate did before taking leave to project how he or she might get back up to speed when returning to the working ranks. I have encountered candidates that have worked part-time or on pet projects while on leave just to stay sharp and current. This shows good judgment and initiative. When evaluating a gap of a few years, consider how important it is for a person to be current in the technologies you use. Things may not have changed much in two years.

Occasionally gaps are due to unemployment, the larger and more recent the gap, the bigger the concern. Be aware that the more years of experience a candidate has, the longer it usually takes to find a job. So don't automatically count unemployment against someone.

Some explanations for unemployment gaps are surprising. One person I interviewed had financial fortune and didn't need to work. Another had to take care of a sick parent. Another stayed home with the newborn while Mom worked. In many cases, the candidate wasn't idle during these times. One was able to provide a list of technologies he had explored and example software he had created during his unemployment.

I think it shows lack of judgment for a candidate to fail to include in a résumé an explanation for a recent and significant gap. A one-line description will rarely hurt your chances and avoid leaving the employer guessing.

Overlaps are often more serious than gaps. Why is the person working two jobs at the same time? If it is financial need, it brings into question all sorts of troubling possibilities. Did the candidate get into financial distress through poor judgment? Is the candidate unable to survive on a typical software professional's wage? Is the candidate not earning (and maybe not worth) a typical software professional's wage? Is there going to be a focus problem? Is the candidate a security risk?[11] Hopefully your company does background checks where serious financial issues show up.

Most overlap situations involve a private venture like a consulting company or a part-time role in a very small company. These may be innocent, but consider the potential for distraction from your company's needs. Also consider what might be going on in the candidate's life where two jobs are called for. There may be serious behavioral or life choice issues going on. Maybe home life is unbearable. By simply being aware of the possibilities, you will increase your odds of detecting other clues if you decide to continue the hiring process with the candidate.

Ultimately, a second job is unhealthy for the candidate and a distraction for you. Determine how you feel about employees working two jobs. Check with legal counsel because there may be a conflict of interest. Often a candidate who has a small side job or company will be willing to let it go for a

---

[11] Companies where developers are trusted with access to financial systems often consider a candidate with severe financial problems a security risk because he may attempt to "solve" his problem by taking advantage of that access.

fair offer.

I have seen some pretty confusing job histories with respect to dates. If you think you see a problem, take extra care to review the history. For example, sometimes a candidate will provide a sort of hierarchy where the top level is employment for an outsourcing company with an umbrella duration, and the next level are all the projects worked on and their dates. You may see something similar when a candidate has worked for a single company on multiple projects or with multiple job titles. At first glance, this looks like overlap but it isn't. Take the time to work it all out; you might be surprised by what you discover.

If the résumé does not include dates in the job history I immediately wonder what the candidate is trying to hide. The typical job history format with dates is so ubiquitous that either the candidate is ignorant of the basics of writing a résumé, or must be hiding something. Either case is a problem to weigh heavily.

The bottom line is that if you decide to talk with a candidate who has gaps or overlaps, be sure to ask about all of them. Be sensitive to honesty here because obfuscation is something you should already be suspicious about.

Associated costs

Some candidates come with strings attached. Relocation and visa sponsorship are the most common. If you are considering candidates that require additional costs to hire, you should factor in these costs when evaluating the candidate. As with salary, you should decide before interviewing how to handle situations where additional costs are involved.

Out-of-state candidates are more expensive to interview. The phone interview costs no more, but flying them out for an on-site can be very expensive. If the candidate is marginal, it is usually not worth proceeding.

Depending on your company policy, relocation can be quite expensive. Minimal relocation is typically around five thousand dollars; higher-end packages for developers could cost tens of thousands of dollars. Often the candidate is required to stay with the company for a certain period of time or repay part or all their relocation costs on early termination.

If your company sponsors visa and Green Card applicants then consider those costs. Ask your HR organization what is offered and what the typical costs are. Always keep associated costs in mind when viewing candidate resumes. It may not be an overriding factor, but it may tip the scales.

Information hiding

I've already discussed missing information. Sometimes there is a general theme of obfuscation throughout the résumé. Keep track of missing information as you review the résumé and see if you spot a trend.

Things that might be missing but should not be left out include:

- Education: No secondary education listed, failure to specify the degree studied or earned, no institution named, or no dates are all concerns.

- Technology details: Lack of clarity about what technologies were used on each job is a serious problem. Look for a summary of technologies that seems targeted, rather than a laundry list, to mitigate this concern (See "Buzzword bingo" on page 87).

- Employer details: Missing dates, titles, names, and locations are all concerns. Some candidates feel they should not provide the names of their employers; they usually will include a moniker like "E-commerce Company." There is almost never a good reason to do this, and it at least shows poor judgment.[12]

Keep an eye open for anything that you feel ought to be present but isn't. If you see a trend of missing information, it is a problem. At best the candidate shows poor judgment, at worst obfuscation is at play. In either case, this is probably someone you don't want to hire.

3.4 Résumé review techniques

In this section I discuss some résumé review techniques that will get you started. If you follow these techniques and the other recommendations in this book, you will learn to hire well. But this is just a starting point. As you grow in the art of hiring, you will develop your own techniques, style, and intuitions. You will go beyond what is written here and you will be able to coach others.

[12] Some security-related jobs might require the candidate not disclose their employer. The candidate usually will mention a security clearance if this is the case.

Approaching the résumé

Up to this point, I have discussed the details of résumé reviews. This section gives you some high-level techniques and structure for approaching a résumé and applying your detailed review knowledge.

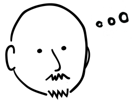
Some experts argue whether to read résumés front-to-back or back-to-front. I feel these arguments are meaningless because I use both for different things. In total, I use three résumé-reading techniques: scans, front-to-back, and back-to-front. Referring back to the perspectives, certain reading techniques are better suited to one perspective than another. I will give my preferences and reasoning and leave it to you to decide what works for you.

Scans

I like to scan the résumé a few times before diving into detail. I look for several things and I find it is easier if I look for one thing per scan. Here is what I often use scanning for:

- Are all the expected parts of the résumé present?
- Where does the candidate live?
- How long has the candidate worked in the field?
- Are there any gaps or overlaps present in the job history?
- What do the job titles tell me?
- Does the skills section have useful content, or is it just buzzwords to be ignored?

Scanning is a good way to answer the "meta-questions." I do this first but sometimes I come back to scanning if I think I may have missed something or if the details encourage me to revisit some meta-question. Think of scanning as skipping over the surface of the résumé, looking mostly at the headings, and occasionally drilling into detail when it serves the purpose of your scan. You can scan in either direction.



Scanning is good for understanding the "big picture" or for filtering through a large number of diverse résumés.

If you are filtering résumés, a scan can sometimes quickly reveal whether the candidate should be immediately removed from the résumé pool. For example, if there is a skills section, and it does not list your required technologies, that candidate might be rejected. I will usually affirm this by next scanning the job history for signs of the technologies before rejecting outright. I do this because people tend to be a little sloppy about the skills section and may have failed to keep it consistent with the job history.

Scanning is a great way to get an overall picture of the structure of the résumé, which in turn may lead you down other paths in your review. It is important to view the candidate at multiple levels of detail. The big picture is just as important as the specifics, and scanning is a good practice to help you step back and view the candidate from a career perspective.

Front-to-back

Our natural tendency is to read the details from front to back because that is the way the résumé is presented. The argument goes that this is the way the candidate intends you to read the résumé, so this is the best way to approach it. This is true, but two things are nearly always also true about résumés: many of the sections can be read for detail independent of each other, and job history is universally presented in chronological order.

This is not to say the sections are independent of each other because together they tell the candidate's whole story, but scanning can be used to see the big view. I tend to scan forward for everything but the job details. So I might do a front-to-back read but skip the job history.

Back-to-front

I prefer to read the job history from back-to-front. I find it counterproductive to read the job history front-to-back because it forces me to think backwards in time. By reading the job history in chronological order, it is easier for me to understand the candidate's "story." I can see how skills, responsibilities, and leadership have evolved. The career trajectory is easier to quantify.

I prefer scanning back-to-front as well when I am scanning for chronologically related meta-items such as job titles. This helps me get a sense of the candidate's career trajectory purely from a title over time perspective.

Often using a combination of front-to-back and back-to-front will illuminate different things. Get used to doing both and explore which work best to help you see different things.

Tabulating pros & cons

Establish some way to tabulate the pros and cons you observe in the résumé. You want some objective way to measure what you see and to identify factors that tend to amplify one another. When you have tabulated your observations, it is easier to make a fair decision about the candidate.

Using a checklist

Create a checklist, or use the one I provide in Appendix C, to tabulate what you find. This is a Shu level approach, but it is a good way to get familiar with what you should begin looking for. Use the checklist to transfer your comments to your tracking system.

With checklists it may be difficult to capture your intuitions. Likewise, the combinatorial or offsetting effects aren't captured on a checklist. I suggest after completing the checklist, you review it looking for meta-level patterns. Augment the list with notes and also add any hunches or impressions you feel about the candidate.

Tabulating on the résumé

You might find it easier and more flexible to take notes on a printed résumé as you review. You can annotate specific things with a plus or minus sign and maybe even a multiplier if you want to get fancy. Write your observations next to what you see.

When finished, log your findings into your tracking system and enter your observations and recommendations. As always, enter complete information about your observations into the tracking system.

Tabulating in the tracking system

As you get better, you may find it easiest to simply transcribe your observations directly into the tracking system. This approach has the benefit of avoiding repeating yourself, but it takes a lot more skill and experience to do it well.

Be aware that the repeat effort involved in the other techniques will often help you clarify your observations. Get some experience under your belt before trying this shortcut, because you are likely to miss or misrepresent information if you start off doing this.

I like to open the résumé on the computer and go back and forth between it and the tracking system, adding comments as I encounter things I observe. It's like transcribing my thoughts. As I make passes on the résumé, I add information to the tracking system.

When I am finished reviewing the résumé, I go back to the tracking system and review my stream-of-consciousness notes. Typically they need some clean up, so I will edit them. Next, I perform the meta-level analysis I suggested above. I look for items that compound or offset each other. I look for general patterns and make notes of these at the end. If I have any gut hunches about the candidate, I enter them. Finally, I make my recommendation concerning a phone screen.

3.5 What to enter in the tracking system

The first thing is to figure out is how to present your detailed observations. You may want to write in prose. This is useful for telling a story with your observations, for making a point, or drawing meta-level conclusions from the details.

It might be helpful in making your decision to organize your observations in a pros/cons or +/- format. You can keep each item short and simple. When you write them out this way and see all the good and bad juxtaposed, it will help you make a definitive recommendation.

Blending prose and pro/con organization works well too. Be terse in the pros and cons, but use complete sentences to explain yourself and to draw meta-level conclusions. Be concise and accurate so that your comments are not misconstrued.

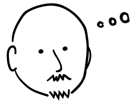
Whenever expressing feelings or hunches, use language that makes it clear these are not facts but feelings. Likewise, employ words that capture your level of confidence in what you think you are seeing. If something appears to be true but you don't have facts to back it up, use language that makes clear your uncertainty. Never write a hunch or feeling without some supporting evidence.

Include suggestions for the interviewers in the next phase of the hiring process. Examples include concerns or hunches about technical depth, job hopping, employment gaps, or personality questions. Even if your recommendation is negative, you might be in the minority, so include suggestions regardless of your recommendation.

The last and most important thing you must do is to make a recommendation on whether to invite the candidate for a phone screen. I recommend you put a one-word recommendation at the top of your comments, either a yes or a no. (Or pass/fail, accept/reject; Chose some common terms indicating whether or not to invite the candidate.) Putting your recommendation at the top makes the process coordinator's job easier when tallying the review results.

Weighing the pros and cons

Whatever method you use to tabulate what you see, keep a few things in mind. First, pros (or cons) are not equal. In other words, it would be extremely naive to choose based on the number of pro or cons: "There are six pros and three cons so I recommend a phone screen."



Weigh each pro or con appropriately according to relevance. Reread your evaluation to see if it supports your decision.

Every pro or con has its own relative importance. A good skill match is more important than a good education; excessive buzzwords are not as significant as job hopping. Address this mentally as you review your observations, or assign some sort of weight to each observation. I personally think the mathematical approach leads to rigidity and a tendency to stifle intuition, but it might be a good place to start.

Second, keep in mind that certain observations can compound or offset each other. For example, work gaps and information hiding might indicate a significant untold story. At a basic level you have the individual observations; at another level, you have the compounding or canceling aspects.

Third, if you foster any hunches or intuitions about the candidate, weigh them appropriately. If you don't have much evidence, it is worth mentioning but may not influence your recommendation much. The more evidence you have to support your hunches, the more weight you should give them.

The art and practice

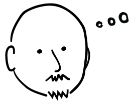
Rules and checklists will not take you very far, because good résumé reviewing—and hiring in general—is an art and a practice. Getting better takes conscious effort, self-awareness, and intentional learning with feedback and mentoring.

Self-awareness means you are conscious of how external forces influence your ability to perform quality hiring. You recognize and accept when the opinions of others are influencing your choices. You monitor your moods and how they affect your ability to observe and be even-handed. The fact is that external forces affect how well we perform in the hiring process.

Seek out feedback loops to reinforce and validate success and to recognize and correct failures. Use feedback loops in the process to hone your powers of observation and intuition. When you have the opportunity to be in multiple phases of the review process, check your early hypotheses to see if they are true. Develop what works and question what doesn't.

In each type of review, you form a hypothesis about a candidate on incomplete and fuzzy information. Whenever you get the chance, test your hypotheses. If you can't interact with the candidate, read the comments from phone and on-site interviews. Talk with other reviewers. Re-review the résumé to see if what came out in later phases of the interview process were tipped off in earlier phases.

As you improve, you will begin to see details and relationships you didn't notice before. If you always hone your skills and check your hypotheses, you'll begin to trust your intuitions and high-level correlations. You will make mistakes but as you reflect on things, you will also learn how far you can take your intuitions.



Be willing to change your hypothesis about a candidate if the data demands it.

When data fails to support your hypotheses, you must be willing to abandon them for new ones that better fit the data. You will realize that "being right" isn't what it's all about; hiring great people is the goal. You won't judge yourself by how perfect or prescient you are, but how efficiently and accurately you are able to identify the right people to hire.

3.6 Transcending the trivial

Checklists and simple techniques are great for the beginner. As you master them, you will find they are incomplete. You will begin seeing patterns that transcend the trivialness of the checklist. You are beginning to *see*.

You will start to notice how two or three pros add up to something very compelling; or how two or three red flags combine to cause you to strongly reject a candidate. Little things will become more meaningful against the backdrop of more obvious things.

You may find you have to explain yourself to the Shu-level folks in your process, and they may not understand. Consider this a learning opportunity for you and your Shu-level colleagues. It is a great sanity check for you. Be wary of the oracle complex, where you begin to think you are able to see things that aren't there. Self-deception is a real danger. Rely on your validation loop and listen to others when they question your *sight*.

There is a fine line between accurately inducing likely outcomes from details in the résumé, and spewing forth hyperbole. Hone your intuition and tap into its power, but be conscious of the fact that intuition is fallible.[13]

3.7 Conclusion

To the untrained, the résumé is nothing but some facts and a lot of filler. But to the trained eye, a résumé is full of useful information. Learning to see the information is just part of the challenge. The rest is learning how to evaluate and weigh that information.

The information in a résumé is fuzzy and incomplete. The reviewer must piece together the puzzle to form a consistent picture of the candidate. As the picture takes shape, the reviewer forms opinions. As new information is added, the reviewer may need to adjust the picture to make it consistent. Seeing the plain facts is easy; piecing together the fuzzy tidbits is an art.

Learn by first using a checklist. Tabulate pros and cons and weigh them appropriately. With practice, you will transcend the need for checklists and will start painting your pictures as you go. Good résumé reviewing saves time and money. As you begin to see beyond the obvious, what was once a tedious task, can become a fun and challenging game.

[13] Gladwell, *Blink: The Power of Thinking Without Thinking* [Gla09]

Chapter 4

Phone Interviews

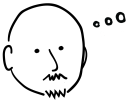
The purpose of a phone interview is to determine whether the candidate should be invited to an on-site interview. Because on-site interviews are costly, phone interviews are used to filter out candidates that don't meet the requirements of the open positions.

The fidelity of the information in a phone interview is much higher than a résumé review, but that fidelity is still much less than that of an on-site interview. Just as with résumé reviews, a well-executed phone interview will reveal a surprising amount of information. On the other hand, a poorly run phone interview reveals little about the candidate and also runs the risk of leaving a bad impression.

Given how important phone interviews are, I find it surprising how badly companies perform them. I have been the candidate on many phone interviews, and I cannot recall one worth mentioning for its quality. This goes to show that phone interviews—and hiring in general—are hard to do well.

4.1 Principles

The phone interview is usually the first interaction between your company and the candidate, so it is important that both make good impressions. You and your colleagues are entirely responsible for the effectiveness of the phone interview. It is *not* the candidate who runs the interview or is responsible for providing the information you desire. Your interview team must conduct the interview and ask the questions that will give you the information you require to make a decision.



Diligently follow the phone interview principles as a basis for great phone interviews.

As with the overall hiring process, defining organizational principles will aid in maintaining quality and consistency across phone interviews. Phone interview principles add to and augment the general hiring principles discussed earlier. Your organization should develop its own set of phone interview principles, but here is a set of effective and tested principles you can use as a starting point:

- Be gracious: The candidate is probably much more uncomfortable than you, so be considerate and accommodating.
- Be professional: You are representing your employer and you have no idea what impact—beyond this candidate—your behavior might have.
- Be informative: At the appropriate times, provide the candidate with quality information.
- Be patient: Answer the candidate's questions honestly and accurately; clarify if needed. Take responsibility to work through any communication difficulties.
- Be prepared: Know where you want to go with the interview and try not to waste anyone's time.
- Be prompt: Start and end the interview on time.
- Be succinct: Avoid talking too much; let the candidate tell the stories. Avoid asking complicated questions.
- Be flexible: Every interview is different and most interviews require adjustments from plan. Be willing to adapt and go where you need to.
- Be assertive: If a candidate gives a vague or incomplete answer, follow up with further questioning to ensure you understand the candidate's ability.
- Be judicious: Make hard decisions when necessary. Gracefully wrap up interviews of unqualified candidates while preserving their sense of

dignity. Reschedule bad interviews. Re-establish focus if the interview drifts.

The interviewers are working from a position of strength so they should be in control and take responsibility for the quality of the phone interview.

4.2 Preparation

The phone interviewers should be prepared as individuals and as a group. The interviewers should be identified early enough to have time to prepare for the interview by reviewing the résumé, comments, and other candidate collateral such as cover letters or questionnaires.

Although you may only be able to have one interviewer, it is better to have at least two. I find three is the best number because it allows tie breaking and gives one more perspective. More than three active participants get unwieldy unless the extras are observers only.

Invite only those who are qualified to interview for the position. Having the wrong people in the interview does more harm than good. Try to invite those who are hiring and have interest in the candidate. That said, hiring is an organizational effort and interviewers must be aware of needs beyond their team and interview accordingly.

Schedule a private, quiet location where the interview can be conducted without interruptions. Interrupted interviews send several messages—all of them bad. Candidates will detect the chaos and get a bad impression of your company.

Schedule no less than one hour for all phone interviews, unless it is a follow-up from a previous interview. An effective interview can sometimes be performed in forty-five minutes, but in my experience, this usually is not enough time. It is easy to expedite an interview when a candidate is clearly not going to work out, but it is rude to expect a candidate to attend longer than planned, and you may run into room scheduling problems if you extend an interview.

All participants must understand and fulfill their roles in the phone interview process. One interviewer will lead the interview, while the rest observe and support. Everyone should be familiar with the typical interview agenda, and each member should have a personal plan for what they hope to get from the interview.

Pre-meeting

It is a good idea to have a brief, pre-interview meeting. Gather five to ten minutes early, preferably in the interview room, to get everyone on the same page. The less experienced the interviewers are, the more important this meeting is. If everyone is experienced and comfortable with one another, this pre-meeting might not be necessary.

Review the job opening, assign the interview roles, and review the agenda. Establish whether any interviewers have specific needs related to the candidate or any specific areas they are interested in exploring. Review the candidate résumé and the comments in the tracking system to be aware of focus areas identified in the résumé reviews.

The interview leader should let the rest of the interviewers know his or her preference for transferring control. Agree on signals to be used to get the leader's attention for a question. This will help make a smooth interview for the candidate.

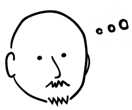
4.3 Phone interview roles

The primary purpose of phone interview roles is to establish guidelines that promote an orderly phone interview. There are three key roles: leader, helper interviewers, and observers.

Interview leader

The interview leader, or just leader, is responsible for most of the interaction with the candidate. From the candidate's perspective, it is easier to understand what is going on if fewer interviewers speak. The leader coordinates the other interviewers' speaking opportunities to minimize confusion.

The leader has mastery of the phone interview principles, is a good communicator, and is aware of the specific needs of the other interviewers.



To minimize confusion, the interview leader does most of the talking and coordinates questioning among the other interviewers.

As coordinator, the leader provides opportunities for the other interviewers to ask questions. He or she achieves this by picking up cues from the others and occasionally asking if anyone has any questions. A good time to check for questions is at transition points in the interview: transitioning from one job description to another, one technical area to another, *etc*.

The leader is responsible for keeping the interview on track. This requires managing both the candidate and the other interviewers. Be sensitive to the perceptions of the candidate. It is better to hurt a colleague's feelings and explain later than to give a bad impression to a candidate.

Use the other participants and their specific knowledge; if a database expert is in the room and you need a few good database questions, defer to the expert. Judiciously spread the responsibility while avoiding too much jumping from person to person. The candidate can become confused by a discussion that bounces from interviewer to interviewer. Make transitions explicit and clear; use introductions or leading phrases that make clear the discussion is being handed over temporarily to another interviewer.

Helper interviewers

Helper interviewers, or just "helpers," should make every effort to work with the leader to ensure a quality interview. The leader has a difficult job, so try to be a positive support while at the same time achieving your specific goals. To that end, make clear to the leader any special needs or instructions in the pre-interview meeting.

Keep in mind that the leader, being focused on conducting the interview, often misses nuances. The helpers are listening more than thinking about what to say. Ideally, the helpers could exert all of their efforts on hearing and understanding the interview, while the leader conducts the interview.

There will be times when a helper wants to ask a question or inject a comment. Unobtrusively get the leader's attention through eye contact, raising a hand, or even by passing a note. Sometimes the best thing is to pass your question to the leader to ask. This allows the flow to be maintained.

Be judicious about interrupting the leader. Try to do so only if necessary and in such a way as to preserve the quality of the interview. Balance this with your need to achieve your goals.

Observers

If your hiring staff is growing, you will need a safe way to train new participants in the art of phone interviews. An effective way to do this is to have new people play the observer role.

The goal of the observer is to watch and learn. The observer should not ask questions, interrupt the other interviewers, or make comments during the interview. The observer should take notes and save questions or comments until after the interview, during the interview evaluation.

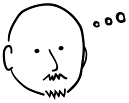
It might be helpful to let observers provide candidate evaluation, but I recommend allowing this only after all the participants have spoken. New participants tend to introduce some noise into the conversation, and it is best if the experienced interviewers are not distracted.

Others who might fill the observer role include HR employees and recruiters. You might wish to have HR representation on a phone screen, although probably not on a regular basis, for a variety of reasons. HR may have particular concerns about the candidate, or HR may simply be curious about how you conduct interviews.

If you have a close relationship with your recruiters, you might consider allowing them to observe some phone interviews. It will help them understand your processes and enable them to better tailor their filtering to your needs. A word of warning: a recruiter's interests are ambiguous. Although your satisfaction is one of their goals, ultimately they don't get paid unless you hire someone. A recruiter may, for example, be undisciplined in filtering résumés, hoping a shotgun approach will lead to more hired candidates. If you do not trust a recruiter, you are better off keeping a boundary in place..

4.4 Agenda

Having a standard phone interview agenda for your organization serves at least two purposes. First, it gives your hiring staff a baseline framework for achieving a quality interview. Second, it provides consistency across phone interviews. If there is no consistency in your hiring processes, it will be very difficult to make consistent hiring choices. Quality and consistency increase the chances of leaving candidates with a good impression of your company as well.



Use a standard phone interview agenda so that all interviewers are on the same page.

I recommend your organization develop its own agenda, but here is a phone interview agenda that has worked well in several organizations:

1. Preparation: Hold the pre-interview meeting described previously.

2. Calling: Call the candidate at the specified time. (one minute)

3. Introductions: The leader introduces him- or herself by name and title as a representative of your company. The other participants are introduced by name and title. This can be done by the leader or by each participant in turn. (two minutes)

4. Setting expectations: The leader asks the candidate if they have an hour to participate in a phone interview (Hopefully they say yes!) and checks that the candidate is interested in the same position the interview team is calling about (sometimes there are mix ups). The interview leader then explains the rest of the agenda without mentioning times. (two minutes)

5. Exploration: The leader begins the formal interview, allowing interviewers to participate as described previously and answering the candidate's questions, if any. (thirty to thirty-five minutes)

6. Company overview: Someone on the interview team briefly describes the company and the hiring organization. Discuss the structure, culture, team sizes, and methodologies. (three to five minutes)

7. Position overview: Briefly describe the open positions in the context of the organization; it is preferable to have the interviewers representing each position perform this. Describe the expectations, interactions, tools, and skills related to the position. (three to five minutes)

8. Q&A: Give the candidate an opportunity to ask more questions. Hopefully the candidate is curious about processes, tools, team structure, culture, *etc.* (five to ten minutes)

9. Closing: When there are no more questions, or you are out of time, the interview leader closes the interview by briefly explaining the next steps, a response timeframe, and thanking the candidate for taking the time. (two minutes)

10. Candidate evaluation: Discuss what you liked and disliked about the candidate. Follow your organization's protocol for determining whether to invite the candidate for an on-site interview.

11. Interview evaluation: Discuss what went well with the interview, what didn't go well, and what can be done to improve.

The leader is responsible for following the agenda and coordinating the other interviewers. The interview should take about an hour, not including preparation and evaluation.

Be sure to start the call on time. This is your first opportunity to demonstrate your company's competence and respectfulness. If ever you should start late, be sure to apologize to the candidate and be prepared to accommodate any fallout such as having to reschedule the interview when a full hour is available.

Introductions are another opportunity to show the quality of your company. They also give you the chance to measure the attentiveness and respect the candidate gives you. Does the candidate ask for clarification on your names? Are you later addressed by your names? It is a lot to expect for a candidate to get everyone's names, but if they do, that reflects well on them.

Setting expectations is about being sure there are no surprises. If you and the candidate know what is coming, you can manage your communication. If the candidate knows there will be time to hear about your company and ask questions, they will be less likely to interrupt your flow with questions that would be better asked later. After setting expectations, it is a good idea to ask if the candidate has any questions before starting.

The exploration portion of the interview is dominated by your information gathering. You control the discussion by modulating the open and closed questioning, probing, and transitioning from point to point. This portion of the phone interview is the most complex and therefore will dominate the rest of this chapter.

The company overview provides a context within which to discuss the job position. This is the time to discuss the business model, customers, organization, team structure, culture, values, methodologies, and the like. This is a great opportunity to sell the great things about your company and why you like to work there. It would be wise to develop a script that everyone uses, to promote consistency.

Next, provide an overview of the position or positions available to this candidate. Describe the team context, the skill and behavior expectations, and the nature of the work.

The leader must manage the call to ensure there is sufficient time for question and answer. If you do not leave enough time, or need to lengthen the interview to accommodate this, you will leave a bad impression with the candidate. On the other hand, if all the timing works out smoothly, it again demonstrates your competence and respect for individuals.

Complete the interview by explaining what will happen next. Be sure to set a timeframe within which the candidate should expect to hear back regarding an on-site interview. Always contact the candidate as soon as possible, whether or not they will be invited for an on-site. It is extremely disrespectful not to notify a candidate in a timely fashion, or worse, not let them know they won't be invited for an on-site interview.

If the interview time expires before you are finished, you have a few options. First, be sure the room you are using will be available. If you do, you could ask the candidate if he or she can stay on the call. This is risky because it may put off the candidate, so try to measure the candidate's ability and willingness to accommodate before asking. If the candidate has more questions, this is often the best way to go.

A second option is to schedule a follow-up call. This has the advantage of being planned. The third option is to move to the next phase and take the risk of an incomplete phone interview. This may not be bad as long as the candidate's questions were answered and the risk of bringing the candidate on site is low. The risk can be low, for example, if you are very confident in the candidate's ability or the candidate is local.

Finally, perform the evaluations. Evaluate the candidate and make a decision about an on-site interview. End with an evaluation of the interview itself. Hiring is a continuous improvement process, so never let pass an opportunity to reflect on performance.

4.5 Have a plan

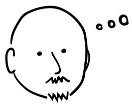
There are at least two levels of planning in preparation for a phone interview: the plan for the team and the plans of each individual interviewer. If you have an established agenda, there may not be much for the team to plan. In the pre-interview meeting discuss any significant points drawn from the résumé review comments that you need to address for the particular candidate and hiring situation. That should only take a few minutes.

Each interviewer should also have a personal plan. The leader should plan how to structure the interview and ensure all identified concerns and questions are addressed. The other helper interviewers' plans will depend on their needs. If an interviewer is hiring, they should have a clear understanding of what they are looking for and how to get that information.

Interviewers who are not hiring should be familiar with the positions and the hiring teams' needs. All interviewers should have questions in mind that will serve their goals.

Planning is valuable, but sticking to the plan is not so important because most interviews produce unanticipated surprises. For example, you would normally give senior candidates open questions, but if the person is overly talkative, you might use narrow or closed questions to manage the tempo of the interview.

Sometimes the goal of an interview may change dramatically. For example, you may become confident you really want to hire a candidate. Then the plan changes to sell the company and job to them. Other times, you may realize the candidate is better suited for a different opening than they are interviewing for and want to shift the trajectory of the interview to focus on that position (and inform the candidate).



Going into a phone interview cold rarely ends well. Study the résumé and plan what you want to explore.

Know what you want to explore beyond the basics described above. Know what questions you need answers to. Know the requirements of the open positions and how you will determine if the candidate is a good fit. Have a questioning plan on how to narrow in on specific areas in the résumé that seem promising or concerning.

4.6 Troubles with telephony

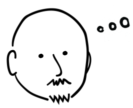
The obvious problem with telephone interviews is the lack of proximity. You don't have the advantage of observing body language. The body positions that give away so much are totally concealed. Furthermore, the subtle yet more revealing movements of the face are also out of reach.

It is debated just how much body language contributes to communication and understanding, and it certainly depends on context. Yet researchers agree that often 50 percent or more of understanding can be communicated through body language.[1] Losing half the communication bandwidth through the medium of telephony is sobering when faced with the important decision of whether to hire someone.

Unfortunately, it gets worse. Telephone communication is highly bandwidth restricted. The information-rich human voice is modulated and compressed to fit into a comparatively narrow bandwidth to conserve network resources. Telephony performance is only about 20 percent that of the human ear.[2]

Other variables can make the situation even worse. The candidate may be on a cell phone where partial or complete drop out can occur, and where you are subject to the variety and vagaries of manufacturer-specific voice encoders and decoders. Furthermore, cell phones have even less bandwidth than landline phones.

Finally, you are not likely to be privy to the situation the candidate is in at the moment. He or she may be at home with one eye on a rampaging two-year old or distracted by a barking dog in the background. The candidate may be outside with background noise and wind distorting the communication. The candidate might be in a conference room, distracted by the fear of being discovered by a co-worker or manager. The candidate might be instance messaging an expert or Googling. You just never know.



Take into account the limitations of telephony and you will get better information.

It is vitally important that you keep in mind just how much communication bandwidth is lost in a phone interview. A word of caution: In the face

[1] See www.wikipedia.org/wiki/Body_language for starters.

[2] Rodman, "Effect of Bandwidth on Speech Intelligibility, The." [Rod05]

of this, it is easy to throw up your arms and assume you can achieve nothing better than low-fidelity interviews. This is not the right response. Instead consider what you must do to maximize the fidelity of the phone interview given the constraints of the communication medium. For example, ask better questions, and pursue understandable answers. I will cover these topics in the next section.

4.7 Exploration

From your perspective, the exploration part of the phone interview is the most important part. It consumes about half the agenda and is where you get an understanding of the candidate through questioning.

There are an infinite number of ways you can go about this part of the phone interview. I am going to present what I have found to work for me, but you should experiment with other approaches. My approach relies on a few basic principles:

1. Build a relationship of trust.
2. Focus questions more on what the candidate has done, not what you want.
3. Always be mindful of the questions that need answering, but don't necessarily ask them directly.
4. Try first to get answers through opportunistic and detailed exploration of key parts of the candidate's job history.
5. Ask specific technical and behavioral questions after exhausting the job history for answers.

Most phone interview candidates are quite nervous and possibly afraid, and rightfully so because this is their first personal contact with your company and they want to make a great first impression. They realize their performance will determine the next step and they understand there are many risks in a phone interview and factors they cannot control. They know they are at your mercy and they are hoping you aren't a jerk. It's nerve wracking!

If you manage the introductions and the expectation setting well, then you have already begun to build trust and put the candidate's fears to rest.

How you explore will cement the trust or blow it out of the water. I try to do several things that I believe help build trust.

The first thing is that I try to keep my tone and demeanor friendly and collegial. I avoid sounding superior, demeaning, or accusing. Regardless of what I am thinking, I maintain a respectful dialog, and this is made easy as long as I focus on information gathering rather than judgment. I do have a running hypothesis about how well the candidate fits the job that I develop during the interview, but I try not to allow it to affect my respect for the candidate as a person.[3]

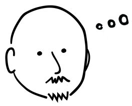
Second, I don't dive into details right away. I let the candidate warm up. I let them answer a few easy questions and get comfortable with the interview and with me. I try to build rapport. This is not easy if the other interviewers are interacting at this stage, so I prefer to have the interview leader do all the talking early and most of the talking as the exploration progresses.

Third, I like to start by traversing the job history in chronological order. Not only does it help the dialog progress in a natural time sequence, but it also makes room for the previous point of not diving in too fast. Nearly always, the older positions are of less interest to me so I can ask the light questions with low risk of missing important information. I may start with questions about education before entering into the job history, which usually also allows for easy questioning with senior candidates. For more junior candidates, the questions surrounding education might be more important and so I must find other ways to ask a few light questions.

Finally, I try to transition gradually from easy questions to deeper questions. Any "abrupt moves" are likely to "spook" the candidate, so I try to keep things smooth and natural. If I am about to make an abrupt transition, I like to warn the candidate so there are no surprises. Surprises lead to confusion and renewed discomfort, and might unwind the trust you've built.

One of the most common and serious mistakes inexperienced phone interviewers make is to focus their questioning on what they want to learn rather than what the candidate has done. By focusing on the candidate's accomplishments, you can get answers to most of your questions while learning much more.

[3] Although the hypothesis I am forming may impact how I am conducting the interview. For example, if I realize early the candidate isn't going to work out, I will respectfully shorten the interview. Conversely, if the candidate is clearly desireable, I may emphasize selling the position more (without compromising the quality of the interview).



Exploration should first be driven by the relevant parts of the candidate's job history. Concentrate on accomplishments; this reveals both knowledge and capability.

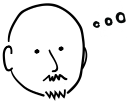
So what's wrong with asking questions focusing on what you want to know? If you do it really well and at the right time, there's nothing wrong with it. But I have found that, at least to begin with, I want to learn as much as I can about what the candidate has done. First of all, the candidate will be much more comfortable talking about his or her reality than yours. That comfort translates to more accurate and truthful information as well as increased trust.

Second, in talking about the candidate you give the candidate an opportunity to speak from authority and knowledge. By doing so, you can get a sense of what the candidate sounds like when she is confident and comfortable. This is used later when you start asking more difficult questions. You can compare the timbre and tempo of the candidate and see if she is still comfortable. This tips you off to just how confident the candidate is with the answer she is giving you.

Third, you will learn much more about the candidate's capability and promise by focusing on what she has accomplished. For example, you learn whether the candidate has developed expertise in anything. You learn whether the candidate has dealt with particularly difficult problems. You learn whether the candidate has handled leadership. These are all vitally important to you even if they don't directly answer your specific questions. If you short-circuit this by asking your direct questions first, you will miss a lot of good information.

Although you are not asking your questions directly, at least not initially, you still must be mindful of what you need to learn about the candidate. For example, if you need strong database programming skills, your questioning can drill down into database-related responsibilities the candidate lists on the résumé. Tease out what they did and whether it stands up to your needs.

After exhausting the job history you should feel free to ask more direct questions regarding your specific needs. You probably have a good idea of most of the candidate's skills but there are certainly going to be some gaps to explore. The next section covers questioning strategies that you can use in the interview, including direct questioning.



After exhausting the job history, you should ask more direct questions regarding the specific things you haven't learned but need to know.

Questioning a candidate is an art, so there is no recipe I can give to enable you to succeed. Instead, I will elaborate on a variety of topics related to questioning candidates that you could use to develop your questioning skills. Much of this information applies to on-site questioning as well.

Strive to be understood

Since telephony is such a problematic mode of communication, it is critical that you work hard to be understood. The candidate is nervous, but often the interviewers are nervous too. When you are nervous it is common to talk too quickly, or use incomplete or compound sentences. Slow down, think before you speak, and repeat the question if necessary.

Although it is difficult on a phone, try to pick up cues on the level of the candidate's understanding. If your question is followed by a long pause, it might mean the candidate doesn't understand the question. On the other hand, he may be thinking about the response. After a few questions, you should know whether the candidate spends time considering the question or whether he starts answering quickly. As you familiarize yourself with the candidate, try to pick up these communication patterns as an aid to your questioning. When a candidate does pause, avoid the temptation to jump in with clarification. Be patient and let the candidate respond.

Consider asking the candidate if he understood question. If not, repeating the question is only helpful if the misunderstanding was due to telephone problems. Instead of repeating the question, rephrase it. Repeated misunderstandings might be a sign of a communication barrier for the candidate. Eliminate yourself as the cause before considering whether the candidate has a communication problem.

Open versus closed questions

An *open* question gives the candidate freedom to answer in an expansive fashion. A *closed* question can be answered with a single word or short phrase. An open question requires the candidate to think and reflect, and possibly express an opinion or feelings. Open questions pass control of the conversation to the candidate whereas closed questions do not.

For example, open questions would be, "Tell me the difference between Java and C++" or "How does an Object-Relational Mapping (ORM) tool help or hinder your program's interaction with a database?" The candidate could speak to these questions for several minutes and has control of how much or little to say.

A closed question would be, "Have you ever used the Java programming language?" Or "Did you complete your degree at that university?" Closed questions are useful for getting factual information that is easily expressed.

Sometimes a candidate may push a closed question open and begin elaborating. It may be that the candidate feels there is something important you should know, so be patient. If you expected a terse answer you might become annoyed. If, however, the candidate repeatedly opens up closed questions, you might be in a power struggle. Be conscious of the dynamics here, because a candidate who can't accept the role constraints of an interview is not going to be able to do so in a work environment either. Some candidates seem to answer all questions in an open fashion, and this may be an indication that the candidate is a poor communicator who is oblivious to the conventions of open and closed questions.

You may see just the opposite: you ask open questions and get very limited responses. This can be frustrating, because you are not able to get a deep sense of the candidate. Sometimes it feels like avoidance, but more often it is a communication problem. In either case, the talker or the silent type, consider what it would be like working with the candidate. Is your team going to be effective, or is the candidate going to drive you nuts? Is the behavior a sign of communication immaturity or something more serious?

By carefully choosing open or closed questions, you can manage the tempo and breadth of the conversation. Be conscious of the kind of answer you are asking for; if you are careless, you might be surprised by an answer, which could influence the quality of the interview you conduct. If you merely require an acknowledgement, use a closed question. If you want to get a deep understanding of the candidate with respect to a technology, behavior, philosophy, or skill, *etc.*, ask open questions. You can open or narrow the questions as needed; it's a range, not one or the other.

Question according to the expected level of the candidate. Senior candidates ought to be given more open questions and be allowed to express their ideas and opinions. Mid- and junior-level candidates will need more guidance. Open questions are fine, but start with narrow or closed questions, and open up as the candidate responds positively. This will also tell you a lot about the candidate's confidence and communication abilities. You might not get interesting answers to philosophical or wide open questions as you might get from a senior candidate, so don't spend too much time there if it is not productive.

Insist on answers

Another common mistake phone interviewers make is to settle for incomplete, unclear, or even evasive answers. There are many reasons interviewers do this. They might feel it is rude to repeat the question. They may be preoccupied with the interview and not realize the candidate's answer was insufficient. Whatever the reason, a key goal of the interview is to get your questions answered.

Although it may be rude to repeat questions in polite conversation, the rules are different for interviews. As an interviewer, insist on getting answers to reasonable questions, but do so in a respectful fashion.

There are several ways to get a question answered. One way is to repeat the question. This approach may be a bit clumsy. For example, the résumé leads you to believe a candidate was responsible for the design of a messaging component:

> **Interviewer**: *What forces influenced your choice of Java Message Service instead of, say, RMI?*
>
> **Candidate**: *JMS is a standard for asynchronous message passing among different Java Virtual Machines.*

This isn't exactly what the interviewer was looking for. He could repeat the question, which might cause the candidate to think, "Hmm, he just repeated the question. Apparently he didn't like my answer. I wonder what he's looking for?" Or the candidate might just get annoyed and repeat the answer. Alternatively the interviewer could acknowledge the answer and restate the question.

> **Interviewer**: *I am familiar with JMS and how it works. What I would like to know is why you chose JMS over the many other possible choices for your message passing technology?*

By doing this, you've respectfully acknowledged the validity of the candidate's answer while making it clearer what you are looking for.

At this point, you would expect the candidate to understand the question and give a reasonable answer.

> **Candidate**: *We needed the ability to reliably send a message to multiple consumers who might come and go independent of the sender. We considered RMI, but it was clear we would have to write a lot of code to get RMI to do what JMS was already designed to do.*

This would be a nice answer and you could move on to another question. But if the candidate said something like:

> **Candidate**: *We used JMS on several other projects and were familiar with it.*

You might be tempted to accept that as a reasonable answer. It does answer the question as stated but it doesn't give any insight into the candidate's ability to make design decisions, which was the real intent of the question. This is an easy trap to fall prey to. But you could follow up with:

> **Interviewer**: *Ok, so your team is comfortable with JMS. Please explain why it is or is not an appropriate design choice for this particular problem, regardless of the fact you chose it for different reasons.*

Now you've given the candidate the opportunity to answer your real question and additionally to possibly reflect on the choice in hindsight. Hopefully a useful answer is in the offing. If the hindsight answer follows, and the response is that the technology wasn't a good fit because of certain reasons, then it might be that the candidate didn't make the choice.

It would be wise to ask what role the candidate played in the design decision. This is nearly always a good question. If the candidate wasn't involved, it might be that he was too junior. That's fine. How good was the hindsight answer? If the candidate was involved and felt he made a poor decision, ask how the candidate would approach the problem differently today.

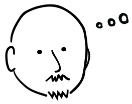
The goal is to learn important things about the candidate so you must be persistent. Often these re-ask strategies reveal an unfortunate truth:

Candidate: *Well, I really wasn't involved in that decision. It was made by the architect.*

This can be really useful. If the résumé said or implied that the candidate was involved ("Selected the message passing solution") then there is falsehood or truth stretching in the résumé. Your strategy should change to question every significant assertion. There is nothing wrong with respectfully asking the candidate:

Interviewer: *Why did you write in your résumé that you selected the technology?*

Yes, it is uncomfortable and feels confrontational, but you need to know. If this candidate is not truthful, you do not want to make an offer. Hopefully the candidate has a good explanation for the résumé gaffe. A pattern of truth stretching like this is serious enough to not hire the candidate.



Follow a line of questioning long enough to get a complete and useful picture; one that contributes to decision making. A common mistake is to fail to ask the last one or two questions that complete the picture.

The bottom line is to doggedly pursue answers. Do not tolerate obfuscation, ambiguity, lack of detail, or responses that don't answer the question. Follow up until you either get an answer or until you are convinced the candidate will not answer. If necessary, explore the avoidance issue.

Avoid compound questions

Another common problem is that a phone interviewer will ask compound questions.

Interviewer: *Tell me about that project and what technologies you used and what was your role in selecting those technologies and whether the project was successful?*

Candidate: *Could you repeat the question?*

Prejudice and ambiguity

We once interviewed a candidate who had impressed us with an application he wrote just for our interview that interacted with our company's Web site. He also came with an excellent reference. We were predisposed to believe this candidate was very intelligent before we even spoke with him.

After speaking with the candidate, we realized that most of the answers were ambiguous. Someone asked the insightful question: "If we didn't already believe this candidate was very intelligent, what would those ambiguous answers have implied?"

Replaying the discussions we realized that we had assumed what the vague answers meant because, after all, this person was smart. As we replayed the answers with healthy skepticism, we realized that they were unsatisfactory. Furthermore, it was clear that the candidate rarely communicated in detail. All the answers were vague or ambiguous.

Be careful not to let external factors influence your pursuit of complete answers.

Sometimes there's a tendency to try to pack all the questions you have into one big question. Who knows why; fear of forgetting? In the example at the bottom of the previous page, there are four open questions, each of which could have a long and important answer.

It is unfair to confuse and frustrate the candidate by asking compound questions. If you have a line of questions you wish to ask, write them down so that you do not forget them, and then be patient and ask one question at a time. You will get better information, run a smoother interview, and have a happier candidate.

Candidates often lose track of the questions within the compound question and fail to answer all of them. I have even heard interviewers say, "She didn't fully answer my question!" What is happening is that the interviewer has a *line of questioning* in mind but has compounded it into one question rather than asking them one at a time. Not only is a compound question difficult for the above reasons, but it also locks in the line of questioning to one specific path. This is unfortunate because, if asked one at a time, the questions might lead the discussion elsewhere.

Because the candidate often has trouble with a compound question, answers are often fragmented, disjoint, or simply incoherent. Maybe in our example, the candidate misunderstands the last open question in the compound question and explains how the selected technologies were a success rather than whether the project was a success. The interviewer gives the candidate a demerit for not understanding the question when the real problem was a bad question.

Avoid problem-solving questions

Yet another common mistake is to ask problem-solving questions on a phone interview. This is almost always a bad idea:

- Explaining a problem-solving question accurately over the phone is difficult.
- Understanding a problem-solving question over the phone is difficult.
- Whatever the answer, it will not tell you much.

Think about how problem-solving questions are typically communicated. They are written down on a piece of paper, a white board, via electronic document, *etc.* You cannot use these media effectively over the phone.

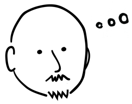
An example problem-solving question is: *using the C++ programming language, write me a method that computes the result of taking one number to the power of a second number.* You would ask the person to write down a solution and then read it back to you. This would be unwieldy and would waste a significant amount of time when you could be asking other questions.

Verbally describing a problem of any complexity over the phone becomes a syntactical and semantic challenge that is not worth attacking. You have to have the right kind of problem, one that lends itself to being verbally described (of which there seem to be very few). Then everything depends on your ability to precisely and accurately express the problem.

Unfortunately, no matter how well suited the problem is, and how well you express it, you have very limited ability to determine whether the question was understood. In a face-to-face, problem-solving scenario, candidates not only have the written problem to point at and ask questions about, they also have all the in-person communication forms such as body language and facial expressions, discussed previously, that are not available when using the phone.

Then the candidate is faced with clearly communicating the solution back to the interviewer. The accuracy of that communication is critical. If the communication hasn't been going well, or if the candidate is even a slightly unclear communicator, there will be confusion. There may also be frustration that could lead to prejudiciary decisions by the interviewers. The "problem" is no longer solving a power function, but instead, one of communicating the solution over the phone. This communication modality is rarely, if ever, encountered in the work environment, and so is entirely artificial.

Finally, whatever answer the candidate gives you has very little value in the context of an interview. If the candidate knows the answer, all that tells you is that she knows how to solve one problem. Because the problem-solving question was asked over the phone, it probably is not a very interesting problem anyway. If the candidate fails to answer correctly, what do you know? The odds are quite high that the failure was due to miscommunication and not the candidate's inability to answer the question. This is misleading and benefits no one.



Problem-solving questions rarely work in phone interviews. Use the limited time you have to ask questions better suited for the limitations of a phone call.

Problem-solving questions are an excellent way to explore knowledge, creativity, and analytical skills. To successfully communicate problems really requires in-person interaction. In a face-to-face scenario, you also enjoy a wealth of secondary information: performance under pressure, thought processes, presentation skills, and collaboration skills, for example.

You could devise ways to communicate problem-solving questions to phone interview candidates. For example, you could send an email to the candidate just before the interview. I feel this sort of approach is highly error-prone. It is likely to create an uncomfortable environment for the candidate, and you still fail to receive a high-bandwidth communication.

Deal-breaker questions

You may sometimes ask questions whose answer could break the deal. For example, if the candidate cannot relocate and you do not allow telecommuting for the position, then there is no point in continuing with the interview.

Deal-breaker questions ought to be asked at the beginning of the exploration phase to avoid wasting people's time. The degree to which you must focus on deal-breaking questions depends upon how the candidate arrived at this phase of the interview. If a recruiter with whom you have a productive relationship referred the candidate, the recruiter should have asked your deal-breaker questions for you. Don't waste time in the phone interview asking them again. If candidates pass through your HR organization, HR should be asking the deal-breaking questions on your behalf.

Not having the requisite skills and experience can be a deal-breaker. If you are hiring a senior .Net developer, of course, the candidate must be accomplished with .Net. That usually can be gleaned from the résumé, though, and need not be asked directly. Rather, in the exploration part of the phone interview, you are more interested in determining whether the candidate is strong enough in .Net for your purposes.

Typically deal-breaker questions are closed questions. "Are you able to work in the office forty hours a week?" "Are you able to relocate to our area?" Or, "Are you able to legally work in the U.S.?" A "No" to any one of these sorts of questions breaks the deal, and you should end the interview immediately and explain why.

Knowledge-testing questions

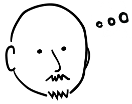
Although problem-solving questions are not very useful on phone interviews, knowledge-testing questions are. The challenge with knowledge-testing questions is to ask questions that are appropriate for the candidate and for your open positions.

For example, asking a candidate how the Java programming language implements inheritance might not be appropriate for a candidate with eight years Java programming experience and a Bachelor of Science degree in Computer Science. On the other hand, this could be an appropriate question to ask a candidate with three years of Java programming experience and limited formal training in computer science.

If you ask questions that are too easy for a senior candidate, you run the risk of insulting the candidate or conveying a lack of knowledge on your part. A strong candidate is going to wonder what kind of operation she is dealing with. Asking questions that are too difficult for a less senior candidate will create an uncomfortable atmosphere. The candidate might become defensive or fearful. In either case, you won't get good results.

Show your excellence by asking the right questions. Just as the wrong questions put doubt into the candidate's mind, the right questions will elevate the perception of your organization. Insightful and penetrating questions are refreshing to the candidate and will instill an interest in your job position.

To ask good knowledge questions, the phone interview team must be familiar with the job for which the candidate is applying. You must know the skills required and the expected level of mastery. For example, a job may require expert level Java programming skills and a functional, if not expert, ability to programmatically interact with a database using SQL and an ORM tool, say Hibernate. The databases with which your developers interact are very large and performance is critical. Just this brief description establishes clear expectations from which you can ask many knowledge questions.



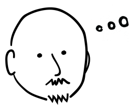
Ask knowledge-testing questions at the appropriate level of expertise and that the candidate ought to be able to answer off the top of her head.

For this example, Java related questions would focus on more expert aspects. It would probably not be appropriate to ask the inheritance question or what interfaces are; both basic Java concepts. It would be appropriate to ask about generics or the `java.util.concurrent` package for concurrency because these are advanced concepts. Ask questions about specific aspects of the language that are important to you. If you use Java EE, Swing, or RMI, you have many lines of questioning that would allow the candidate to demonstrate knowledge. Since performance is important to the job, you might ask the candidate about specific performance enhancements a Java programmer could use, or performance problems she has solved in the past.

Canned questions?

It is important for an organization to have a consistent set of topic areas to explore on phone interviews. I hesitate to call these "canned questions." I prefer a "bellwether line of questioning" because "canned question" comes with some baggage.

A canned question is something you read and get a response. This approach is less than ideal because interviewers abdicate the responsibility for deep discovery. Furthermore, a canned question might imply a right answer, which may also be problematic. Using questions that have one right answer—closed questions—are of limited use when evaluating a candidate. What can you say about a candidate that gets the right answer? You can say they know the answer to that question, but you cannot say whether you should hire them on the basis of that answer. What can you say if they get the wrong answer? You can say they failed to come up with the right answer in this situation but you cannot say they wouldn't come up with the right answer in a real-world situation, and you cannot decide based on their answer whether to hire them.



Use a line of questioning, a dialog, that delves deeply into an important subject matter. The path taken will depend on the candidate's answers.

Following a line of questioning, on the other hand, is goal directed and interactive. It encourages the interviewer to explore with the candidate and encourages open questions. Open questions allow the candidate to elaborate on what they know, demonstrate their communication skills, and give insight into how they think. The downside is that this approach requires practice and skill. Here is an example line of questioning for the hypothetical Java Developer position.

Goal: Explore Java knowledge through the candidate's understanding of Java 1.5 features. The line of questioning goes something like this (although it varies depending on the answers):

> **Interviewer**: *What Java 1.5 features have you used?* (A serious programmer should be familiar with the versions of key feature introductions. If they are not, it puts into doubt their mastery of the language.)
>
> **Candidate**: *I have used generics, the advanced for loop, and annotations.*

Interviewer: *What about the* `java.util.concurrent` *package?* (This is one people forget about, but is very rich for questioning.)

Candidate: *Oh yes, I didn't realize that was a 1.5 feature. I have used it.*

Interviewer: *Tell me what problems you have solved using* `java.util.concurrrent`*?* (Next discuss the concurrency features and how the candidate used them in their design. This is a good way to determine good versus great.)

Interviewer: *Tell me how you used generics.*

Candidate: *I used them with the new collections such as* `List` *and* `Map`.

Interviewer: *Have you ever written a generic class?*

Candidate: *No I haven't, but I think I could.*

Or:

Candidate: *Yes I did. I wrote a...* (Then we discuss that.)

Interviewer: *Are there any problems or deficiencies you are aware of with the Java implementation of generics?* (Here we are looking for some understanding of the concept of erasure. This is an advanced topic that many developers are not aware of, but the strong ones often are.)

I follow a similar approach with other Java features as needed until I have a sense of the candidate's competence. I don't follow the above verbatim, or always in that order, and I may branch off at any time based on an answer. I am always looking for fruitful areas of discovery.

The goal is to see where the candidate falls on a scale rather than whether the candidate can answer all the questions correctly. I get an overall sense of their knowledge in key areas. Moreover, because it is a dialog, I usually learn about communication ability, learning style, design choices, *etc.*

My recommendation is to develop lines of questioning for key skill areas for which you are hiring. Consider having one group of questions that are always asked and another that is discretionary. Document the lines of questioning in a story fashion similar to what I showed earlier and be sure the hiring team understands and can execute the lines of questioning skillfully. Review these lines of questioning with the interview team periodically, reflecting on their effectiveness in practice.

Be careful not to ask problem-solving questions or questions that will not translate well over the phone. For example, do not ask questions of programming language syntax, because they are almost impossible to verify over the phone.

Regarding databases for our example job, you should ask specific questions about the size and performance constraints of databases the candidate has worked with. Explore how the candidate interacted with the database management system (DBMS). Often programmers work directly with the database whereas in many companies, database administrators (DBAs) do most of the database work. Ask about any database performance challenges the candidate faced and how he or she addressed them. Find out what tools they used to analyze the problem and what those tools led them to do.

Ask about the ORM experience. You might ask specific technology questions relating to Hibernate, such as whether the candidate specified the directives using XML or Java annotations. Explore competence by trying to determine whether the candidate misused or misunderstood the technology. ORMs are great for this because if the technology is misunderstood, performance problems are often encountered. Ask the candidate if they experienced performance problems with Hibernate and if so, what did they do to resolve them. A candidate who misuses one significant technology is likely to make the same mistake with other technologies.

You can learn a lot about the candidate's fluency with a technology by simply asking what they think of it. Ask them to list their favorite and least favorite aspects or features. Ask them what they would add or take away from the technology. Their answers often reveal their mastery and depth of understanding. Candidates that merely learn to wield a technology are much less valuable than those who are reflective on the technology itself; this is a sign of craftsmanship.

Certain design questions are quite good for knowledge testing as long as they aren't problem-solving questions. For example, you could ask a candidate to contrast the use of inheritance versus composition. Follow up with a question about when to use one or the other. You could ask a similar question regarding a specific design pattern such as Facade, Decorator, Service Locator, *etc.* Try to tie it back to patterns your development staff often uses.

One danger with regard to knowledge-testing questions is what I call the *Google effect*. I have been on several interviews where it was clear the candidate was using a search engine to find an answer to a question. Some might think this shows initiative and cleverness. I do not agree. The question I asked is one I expect the candidate to know. There is a class of questions that are not important to have in one's head; that's what reference documents are for. I avoid asking those sorts of questions. I ask questions I expect the candidate to be familiar with because it is fundamental to their craft.

You can detect the Google effect in several ways. The first is the sound of typing. Surprisingly, this isn't always detectable. Many phones have a rapid decay of sensitivity with respect to distance and direction and so the typing can be inaudible. Another give away is the tempo of the candidate's answer. Often there will be an initial pause that could be confused with pondering. But then it is followed by an answer whose tempo slows down and speeds up. Imagine yourself explaining to someone, in your own words, what you are reading at that moment. Your attention is divided and so your speech will slow—possibly stop—and then flow again as you switch your attention from reading to speaking. Finally, the answers will sometimes give away the Google effect. Most people cannot rattle off a textbook answer. If the candidate does that, they may be reciting to you. Often their verbal tempo and intonation will help tip this off as well. Consider how you sound when you are describing something you are composing in your head compared to something you are reading.

Of great importance concerning knowledge-testing questions is this: be sure you can distinguish a good answer from a bad answer. There is no point asking questions for which you—or others attending the phone interview—cannot distinguish good answers from bad.

Be careful how you respond to the candidate's answers. If the candidate answers a question incorrectly or is unable to answer a question, try to be encouraging. Incorrect answers don't necessarily need a response. If the candidate doesn't know the answer and says so, you may feel compelled to provide the answer. If done properly and at the right time, this can actually create a positive atmosphere. Other times, it can be condescending or even humiliating. Sometimes it is best to just say, "That's alright, let's move on."

Question your motives when considering a response. Are you thinking of showing your own intelligence? If you think so, don't be ashamed about it; just don't say anything. If you think you can deliver a response with a tone and spirit that will encourage the candidate, consider giving an answer. Some candidates will even ask you what the right answer is. In that case, certainly answer in a collegial manner that educates.

Sometimes giving an answer—properly delivered—can break the ice and increase the level of trustful communication between you and the candidate. The candidate was vulnerable and failed, but you handled the situation with respect and even helped the candidate by providing a little education. If you are skilled at this, you can turn what could have been a fearful moment into a learning opportunity. Candidates, for the information they learned on the interview, have actually thanked me. You can learn a lot about the character and behaviors of a candidate in these situations.

Using the résumé

I like to use the résumé to drive the exploration of the candidate's accomplishments and abilities. The idea is to use the résumé—predominantly the job history—as a framework for discussion. I have found the technique extremely fruitful for discovering competence while also building a great relationship with the candidate. This is not an easy questioning style, so I will spend several pages discussing it.

This is an opportunistic approach in which you get answers to key questions indirectly by focusing on the candidate's accomplishments. In all likelihood, once the job history is exhausted, you'll have outstanding questions that must be addressed through direct questioning.

In this approach to questioning, allow the candidate to elaborate on the details of past efforts and tell their story. Ask leading questions to delve into the details that are likely to reveal important information in relation to the open positions. In guiding the candidate through their story, seek out experience that is applicable to the needs of the open positions. Explore any areas likely to reveal important technical and behavioral information. Having a plan is helpful here.

Use the work history as a guide. A proven approach is to go back in the history to some point and work forward, covering the most recent position last. The justification for this approach is the same as for résumé reviews described on page 95. When picking the starting point, keep time frames in mind. Go back only as far as makes sense. A candidate may have twenty years of experience, but it may only make sense to discuss the last eight or ten years. Use the requirements of the job descriptions and technology lifecycles to determine how far back to go.

Working chronologically through the descriptions, spending more time on the jobs that seem to directly apply to the positions, the jobs that have particularly interesting work (possibly not related to the open positions), the jobs that are most recent, and the jobs that have significant duration. For example, some candidates have been consultants and may have worked on several three-month projects where they list all sorts of pertinent technologies. Since little can be accomplished on a three-month project, don't spend much time discussing it.

You could ask an infinite number of things about each interesting job. As you gain experience, you will be able to identify fruitful areas of the job history and formulate probing questions that get you meaningful answers. As usual, if you are at the Shu level, you can benefit from a checklist as you develop your own abilities. Here is a checklist of things to consider for each relevant job on the résumé, in no particular order:

- Ask why the candidate took the job. This sometimes reveals the candidate's priorities in a way they wouldn't normally express.

- Ask questions sufficient to understand the company, products, and projects, to the extent it provides context for further questioning.

- Explore red flags from the résumé review; other red flags may come up in the phone interview.

- For each job, understand the candidate's title and role.

- For important topics, find out specifically what the candidate did. Résumés often list things the team did, and the candidate may have been peripherally involved. Nail down exactly what the candidate did on each important piece of work.

- Did the candidate demonstrate leadership? Leadership can be formal (team lead, manager, *etc.*) or informal (mentoring, teaching, evaluating, *etc.*). Drill down into specific leadership roles as they apply.

- Determine skill levels for important technologies by asking what the candidate did with the technology in relevant roles.

- Seek opportunities to explore the candidate's problem-solving abilities. Dive deep on promising résumé items or follow up on comments made by the candidate. Don't confuse this with problem-solving questions.

- Explore the size and make-up of teams that were involved in interesting work. This is especially important when leadership roles are involved.
- When it seems relevant, explore the development methodologies used. Try to get a sense of how comfortable and competent the candidate would be using your methodologies.
- Find areas in the résumé you know well and explore those for level of mastery, even if they may not be directly pertinent.[4]
- Try to determine the work environment the candidate prefers and how well it matches yours.
- Don't shy away from areas in the résumé in which you don't have expertise. Ask questions with confidence; the candidate will never know, and you will learn a lot. You are not obliged to reveal your weaknesses!
- Utilize the specific skills of other interviewers when asking technical questions.
- Ask why the candidate left each job and took the next. This can be very important in evaluating the likelihood of long-term employment.
- Look for any signs of behavioral strengths or weaknesses at all times. Evaluate both how the candidate interacts during the interview, and the behavioral information shared in his stories about how he acted in the past. Get personal: would you want to work with this candidate?
- Trust your intuition. If something doesn't seem right, explore it until you are satisfied.
- For each position, see if you can get some sense of how the candidate worked with others. Was he on a team? Was the team collaborative? What was the relationship among customer/product owner, developers, and testers?

---

[4]By hearing the candidate respond in areas you know, you may be able to gauge mastery in areas you don't know. This is done by noting the tone, tempo, confidence, *etc.*, in the area you know well, and then by observing how the candidate responds in areas you don't know well. Mastery in any area is desirable because the ability to master something is reproducible.

- Consider asking the candidate if there is anything else you should know about the job under discussion. Often the candidate will bring things up that aren't on the résumé but are important. I like to do this on jobs that seem less relevant to me, which I intend to move through quickly or even pass over. I might say, "I don't see anything I want to discuss with you about your Software Engineer job at Company 3. Is there anything you feel I should know about it before I move on?"

- Ask why the candidate left, or is interested in leaving their current job. This is very important when the candidate is a suspected job hopper, but is often revealing in general.

Again, the purpose of this checklist is to get you started. Develop your own organizational and individual approach as you transcend from Shu level.

Gauging Growth

There are several lines of questioning you can follow to gauge how the candidate grows professionally. During the resume review you attempted to get a handle on this, so on the phone screen, verify what you observed and go deeper. Keep in mind the ways your environment encourages growth and seek candidates that can fit into it, if not thrive.

It is reasonable to ask the straightforward question: "How do you feel you learn best? Is it through training, reading, being mentored, doing, or something else?" Expect quality candidates to know what their learning styles are and be concerned if they don't.

Follow this up with questions about their preferred learning styles. Ask for an example where the candidate was mentored into a significant new skill. Ask about the most impactful training the candidate attended. Ask what the candidate's favorite technical books are. Ask what technical book she is currently reading.

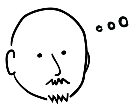
The software profession is ever-changing. If a candidate is not actively promoting self-growth, then I question their commitment. The best software professionals realize that personal development is an on-going effort; a part of the job.

Behavioral questioning

Behavioral traits of a candidate are nearly as important as their technical abilities. Now that you are speaking with the candidate, you can begin to analyze their behavioral traits.

I am not a big fan of hypothetical questioning to determine behaviors: "If you were pair programming with someone who wouldn't let you take a turn at the keyboard, how would you handle it?" That said, they clearly can produce interesting discussion and should be considered if some really important behavioral trait must be tested. For example, if your team always pair programs, you might take an approach such as the above question.

You can make hypothetical questions more concrete with the following line of questions: "How much pair-programming have you done?" "Have you experienced any conflicts or other problems with a pair partner?" "Can you explain those conflicts and how you handled them?" The answers are real, not hypothetical (to the extent the candidate is truthful).



Prefer concrete lines of questioning from the candidate's experience over hypothetical questions. Experience is what the candidate did do. Hypotheticals are what the candidate might do; they may also be what the candidate thinks you want to hear.

I dislike oblique behavioral questions, because they are too hard to interpret. "If you could be any animal, what would you be and why?" The answer could be quite interesting, but I doubt my ability to interpret it in a meaningful way. Questions like this might leave a bad impression with the candidate. There may be oblique questions that are meaningful in certain situations, but my advice is to be conscious of the limited value of information oblique questions might reveal and the potential problems they pose.

My preference is to continually monitor behavioral indicators in the phone interview as I question the candidate. Every observable thing the candidate says or does during the interview paints the behavioral picture. If the picture is incomplete or troubling, I will resort to behavioral questioning.

Keeping Shu, Ha, Ri in mind, here is a Shu-level list of things you might look out for in your questioning during the phone interview. You could not hope to cover all these items in a one-hour phone interview, so focus on the important ones and pick up as much as you can throughout the interview rather than using direct questioning.

- Strong oral communication is essential. Co-workers can adapt to strong accents, but the candidate must be understandable. Oral communication should be clear, unambiguous, and concise, especially when discussing technical concepts. Confirm that the candidate listens and is able to understand your questions.

- Seek signs of adaptability by the candidate's ability and willingness to adjust to different situations. Explore how the candidate handled changes in roles, knowledge domains, methodologies, or other significant transitions.

- Question for the ability to work with others in an environment like yours. The candidate should be able discuss examples of collaboration with other developers, testers, and non-technical personnel or customers.

- Determine if the candidate is liable to take responsibility for his or her actions. If you stumble upon a failure or ill-advised choice, see how the candidate responds regarding accountability.

- It is often easy to determine if a candidate is detail oriented based on how thorough their answers are. You would like to see the candidate discuss at an appropriate level of detail. Questioning on the candidate's view of testing and quality in general, can be revealing. Conversely, be concerned if the candidate veers onto unimportant minutia.

- Look for signs that the candidate is able to make important decisions in a timely fashion that usually result in successful outcomes. This often is related to leadership roles, so pursue them together. If the candidate seems wishy-washy or overly submissive, lack of decisiveness may be a problem.

- Try to gauge whether the candidate is energetic and powerful in action or effect. The energetic candidate is easy to spot and is also usually passionate in discussions of topics she is interested in.

- Look for signs that the candidate can work independently, without excessive oversight or direction. Often special assignments or pet projects will reveal this trait. On the other hand, be wary of the "lone wolf" who doesn't work well with others, or the person who wanders off into the weeds rather than staying on task.

- Seek out examples of initiative where the candidate followed through energetically with a plan or task. Specifically look for instances where the candidate went beyond expectations for the benefit of his employer. Be cautious of extracurricular activities or endeavors that were not supported by the employer; the candidate may have shown initiative in pursuing distractions![5]

- Establish whether the candidate seems honest. Explore any suspicious résumé items and interview answers; always test for accuracy and consistency. Catching a person in a lie or deception is very serious. Be mindful that not catching a person does not mean they are honest.

- Follow lines of questioning that will reveal whether the candidate has the ability to innovate. Talk about the hard problems solved or challenges that required out-of-the-box thinking. If the candidate is published, or has patents or advanced degrees, explore these for signs of innovation. Conversely, be wary of a candidate that "reinvents the wheel." Candidates that are able to leverage existing work should be highly valued.

- Try to discover how influential the candidate is. Look for the ability to marshal resources toward a common cause or the ability to make good things happen. Sometimes volunteerism goes hand-in-hand with influence.

- Does the candidate seem to expect favorable outcomes? This is revealed in a "can do" attitude that is based on rationale and information, rather than fantasy and guesses. Be careful, because overly optimistic people often must pay for their optimism by working long hours to "make it so."

- Ensure that senior-level candidates have been successful mentors. This is an area where direct questioning is sometimes best: "Please give me

[5] Conversely, it is common for people who are bored to pursue extracurricular activities. They are seeking fulfillment that their job does not give them. This is often a good sign, because you are dealing with a person who values stimulating challenges.

a few examples where you have successfully mentored others." "How did you know the mentoring was successful?"

- Based on the candidate's oral communication, determine if she is tactful and diplomatic. On the other hand, avoid the candidate who takes the extreme route to the point where nothing impactful is ever said. This may be a sign of conflict aversion, which is counterproductive in an environment that values innovation and open scrutiny of ideas.

- Does the candidate express a healthy level of pride and self-confidence? You will observe this in the way the candidate talks about significant accomplishments. Be wary of excessive pride in minor accomplishments or expressions of confidence that are not backed up by achievement.

This list was derived from some of the behavioral traits listed in Appendix A. I suggested in Chapter 1 on page 31 that you create a list of behaviors your organization values. It is straightforward to augment this behavioral checklist with your organization's valued behaviors. Many of the above points can be determined without direct behavior questioning, but some do require you to ask.

Irrelevant questions

Your phone interview time is precious, so avoid asking irrelevant questions. I have read that "Where do you want to be in five years?" is considered irrelevant because it does not directly relate to the qualifications required for the job.[6] I think that advice is misleading.

If your goal is to fill a position as quickly as possible—say your company is a start up and if the project doesn't succeed everything else is irrelevant—then this strategy makes sense. On the other hand, if your organization's hiring strategy includes hiring for the future with the intent of retaining employees as long as possible, then the question is *not* irrelevant. It may not be the best question, but it can certainly provide useful information.

Possibly a better question for the company hiring for future leadership, which is something I believe every sustainable company should do, might be, "Have you considered a leadership role in the future?" Depending on the

[6] Rothman, *Hiring the Best Knowledge Workers, Techies & Nerds* [Rot04]

answer, you should then ask the candidate about his or her career plans for the next five years.

If you are hiring senior developers, they will often have a decision to make in the near future if they stay with your company: Do I go into management or do I stay technical? Many senior developers have no management aspirations, and you need to understand how your organization can accommodate such individuals. Do you have a technical track?

Some companies have Staff Engineer or Architect positions that go as high up the job scale as Vice President on the management track. If your candidate tells you he wishes to stay technical, you should consider whether he is likely to succeed on the technical track your company offers.

So what is an irrelevant question? Any question that does not either help answer questions you need answered or illuminate the character and behaviors of the candidate. You need the following answers:

- How qualified is the candidate for the job?
- How well does the candidate conform to our organization's behavioral standards, values, and principles?
- How well will the candidate fit in on the team in question?
- What potential does the candidate have for long-term success at our company?

These questions will determine whether your interview question is irrelevant or not.

If you ask a leading question for which the answer you want to hear is explicitly or implicitly provided, then you are asking an irrelevant question. Avoid tipping the candidate to what you think is a good answer.

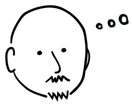
Restating the obvious is irrelevant too. The candidate used COBOL for eighteen years and continues to do so. "You really like COBOL, don't you?" is a waste of time. You might ask, "Tell me why you haven't moved to more modern languages rather than continue to program in COBOL?"

Irrelevant questions are the ones that don't help you decide whether to hire a candidate. They waste precious time and run the risk of offending the candidate. Do you want the candidate thinking, "Why the hell did they ask me my favorite ice cream flavor? Who are these yo-yos?"

Questions you can't ask (in the U.S.)

I am not a lawyer and will not attempt to provide a complete list of questions you cannot ask during the interview process. That said, I think it is important to cover some of the basics. Your company should have guidelines based primarily on your country's laws, but possibly additionally on company policy. Check with your HR and legal departments to be sure you understand the limits imposed by law and your employer.

The bulk of the questions you cannot ask are for the purpose of avoiding discriminatory hiring. You cannot ask a person's age, race, religion, sex, disabilities, sexual orientation, country of origin, or marital and family status. You cannot ask if a candidate is a U.S. citizen. You can, however ask if the candidate is authorized to work in the U.S. There are many questions that are not illegal but are best avoided, because they skirt too closely to illegal questions. Often the answers are irrelevant anyway, and if they are relevant, there are preferred ways to gain the answers without offending.



Be sure you understand what questions you cannot ask. There is always a legal way to ask the questions you need answers to.

You should avoid asking a candidate if she is fluent in English. You can, however, ask what languages the candidate can read, write, and speak fluently. Of course you will have sufficient evidence regarding English mastery by the end of the phone interview.

Maybe a little less obvious, avoid asking a person's height, weight, or other physical attributes. You should not ask about any personal data the hiring process does not require, and the bulk of that need not be asked at this stage of the hiring process. Certainly, you need to know what city the candidate lives in.

Marital or family status and arrangements are out of bounds. You cannot ask who the wage earner of the family is or what the childcare arrangements are. You can ask questions to determine whether the candidate can fulfill the work hours and times of the job though.

The list goes on. Again, check with your HR and legal departments to be sure you know what to stay away from. In most cases, there are ways to ask questions that will give you the answers you need without compromising the legal status of your company. I suggest you review any such tactics with your HR and legal teams to be sure.

It is legal for you to know the answers to many of the questions you cannot ask. In the interview process, answers to many of these questions will be obvious through observation. The candidate will volunteer other answers. For example, it is almost always the case that a candidate who goes through the full hiring process will eventually discuss family. You just cannot ask about it.

Be careful here. A male candidate mentions he has a wife and two children. You are free to say, “I have two kids as well. My daughter is six and my son is two.” You are not free to ask, “How old are your kids?” Or, “Does your wife stay home with the kids?” As personal dialog evolves with a candidate, keep in mind the legal boundaries. It is common to develop a bond with a candidate as you would in any new relationship, but you must protect the interests of your company during the hiring process.

4.8 Learning to trust your instincts

Using your instincts is even more important during phone interviews than during résumé review. Fortunately you have much more information to go on, even with the problems of telephone conversation. As before, hone your instincts by using feedback loops. Test your senses by observing the candidate during the on-site interview or once he is hired.

Negative behaviors like arrogance are easier to pick up on a phone interview than from a résumé. Be conscious of the possibility that behaviors such as impatience, frustration, sarcasm, or even anger might be motivated by the quality of the phone interview. That said, one would think a candidate desiring a job would be patient and compliant.

Take note of your hunches and intuitions during the interview for later reference during the evaluation. Refrain from communicating your hunches to your fellow interviewers during the interview unless you wish to pursue a line of questioning that might confirm your suspicions. The concern is that you might prejudice the rest of the interview. In no case should you make your concerns known to the candidate.

4.9 Evaluating the candidate

The result of the phone interview is a determination of whether to invite the candidate for an on-site interview. The simplest way to make the determination is to have a vote among the interviewers right after the call. Unlike the résumé review, indecision is an option.

If the phone interview was done well, then in most cases, all the interviewers will agree one way or the other. Occasionally, there is a mix of opinions, and a frank discussion about what they liked and didn't like can usually create consensus. It is important to define a protocol for deciding whether to invite the candidate on site.

If consensus is not possible, there are several options you might include in your protocol. You could allow majority rule: if there were three interviewers, then two votes would be enough to decide. You might allow the interview leader to make a final decision. You might have a protocol rule that a tie errs on the side of an on-site interview if the candidate is local, or against an on-site otherwise. You might bring in an arbitrator, say a senior manager, to help the team decide. You might agree to schedule a follow-up call. Or the interviewers might enter their recommendations into the tracking system, and a coordinator or hiring manager decides.

A follow-up call often arises when the interview team realizes they did not get answers to critical questions. If those questions were not asked, or if the phone interview was not executed well in some other way, then a follow-up call is a reasonable option. On the other hand, if the questions were asked and the candidate was evasive, then he may not deserve another chance. Schedule a follow-up call when the team failed to get necessary information to make a decision. The culture should always support such a decision rather than punish for failing to decide after the first interview.

In case there are differing opinions, it can be helpful to use a risk management perspective to reach consensus:

- What is the risk of inviting the candidate for an on-site interview? Several people might waste time and not get work done.

- What other costs are involved? If the team is on the fence, testing the candidate in an on-site interview may not be wise.

- Is the candidate local? If so, the cost of an on-site is much lower, and it is safe to shorten the interview if the candidate doesn't work out.

- What is the risk of losing a mid- or junior-level person? Not much, so don't worry about making the perfect decision.
- What is the risk of not hiring a senior level candidate you know you want? High. They are hard to find.
- How tight is the market for the positions of interest?
- How urgent is the need?
- What is the risk when you hire a candidate you are not sure about? High. If you make a bad choice, that employee will be a drag to the organization, consume valuable resources, may cause others to leave, and will be expensive to replace even if you can get rid of them.

If the evaluation process requires deliberation, you have several things to consider from a personal perspective:

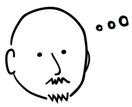
- Be sure of your opinion: you want them, you don't want them, or you are on the fence. Don't irrationally give the benefit of the doubt to the candidate.
- Be accountable for your own opinions; don't let yourself be swayed by others without good reason.
- Don't be too stubborn.
- Don't assume other interviewers saw something you didn't. If you don't know one way or the other, say so.
- Don't underestimate behavioral issues. Ultimately, if the person is easy to work with and reasonably competent, there's a good chance they will succeed.

The evaluation process for determining whether to invite the candidate usually should not take more than five minutes. If it seems to drag on, then the leader must step in and take action to resolve the indecision.

4.10 Evaluating the phone interview

Every step in the interview process is a learning opportunity, and the best time to discuss your phone interview is right after a decision is made. Evaluation is always a good idea, but it is especially important if less experienced interviewers are present. Since you may have to leave the conference room once the interview is over, you may take this discussion elsewhere.

Here having a protocol is again a good idea. Try something simple: briefly discuss what worked well, what didn't work, and what you might want to try next time.[7] In a high-trust environment, consider just talking about the interview. If observers are present, take the time to review what happened and highlight key concepts.



Spend a few minutes after the phone interview to reflect on what went well and what needs improvement. This feedback loop provides individual and organizational improvement.

However you proceed with the evaluation, show respect to each other. Avoid arguments and allow differences of opinion. It is better to agree to disagree than to damage work relationships. If the point of contention is important, you can later refer to your organizational guidelines or bring in someone to resolve the dispute.

4.11 Finishing up

All participants should enter their complete comments in the tracking system along with their decision. Do this quickly so that the process coordinator can contact the candidate either to set up an on-site, inform the candidate that there will not be an on-site, or set up a follow-up call.

If the candidate is being rejected, the quality organization will still promptly follow up. If you are interested in the candidate, it is certainly in your best interest to respond as quickly as possible. A quick response will leave a great impression with the candidate in any case.

[7] These three questions are standard fare in reflection meetings, which are common to agile development. An interview evaluation is essentially a reflection exercise.

4.12 Troubleshooting

By their nature, phone interviews will have problems. There are so many variables and the separation between interviewers and candidate gives rise to all sorts of challenges. Here are some of the more common ones. In all cases, be sure to comment in your tracking system about the situation.

Getting a late start

If you are late in starting an interview, you've already made your first bad impression on the candidate. What you do to recover is vital. The first thing to do is get in contact with the candidate as soon as possible. You might be tempted to just give up—say if you are really late—but that is the worst thing you can do. If you cannot reach the candidate, leave a message and, if a recruiter is involved, contact the recruiter and try to reschedule.

Whether you leave a message or reach the candidate, the next thing you must do is apologize and take full responsibility for your error. Then, to show your respect, give the candidate the option to reschedule or continue with the call. When you do so, also ask if they can do a full, one-hour interview. Do not shorten the interview unless you absolutely have to. For example, you may know that you cannot keep the conference room beyond your initial hour. In that case you may encourage a reschedule. If you are only five or ten minutes late, you might continue and let the candidate know that you may need to follow up with a second call if you do not finish.

If you continue with the interview, run it as you would any other interview. Do not rush. If you must give up some time, eliminate or shorten the least important parts of the interview. If you have to shorten the time for questions, offer a follow-up call, possibly with a senior manager or executive to demonstrate contrition.

Bad connection

A bad phone connection will ruin an interview, so do not try to force it. If either party is having trouble with the connection, the first corrective action should be to inform the candidate you will call them back. Often a new connection will improve the situation.

If recalling does not work, ask the candidate if she is on a cell phone. If so, ask if she can use a landline. If not, try a recall one more time. If that doesn't help, you should reschedule the call for a time the candidate can use a landline.

Throughout this process, be polite and do not show frustration. You can be sure the candidate is feeling at least as bad as you. If you reschedule, as usual, be prompt and try to get the situation worked out as quickly as possible. If a recruiter represents the candidate, inform the recruiter, as they will grease the skids; it is in their best interest to get another call set up as soon as possible. The recruiter can also be much more forward in insisting the candidate get a better connection.

Nobody home

From time to time you will call a candidate and get no answer. In my experience, there is usually a good reason, so I give the candidate the benefit of the doubt until I learn otherwise. I will dial three times before giving up. Then I will try to figure out why the candidate did not answer.

If a recruiter represents the candidate, contact them as soon as possible and let them investigate and possibly reschedule. Otherwise, have the process coordinator attempt to contact the candidate and find a reason as soon as possible. Reasons I have encountered include:

1. Someone along the way wrote down the wrong phone number.
2. Either party had a schedule mix up.
3. The candidate's cell phone died or ran out of charge.
4. A meeting preempted the call.
5. The candidate was stuck in traffic and didn't get home in time for the call.
6. The candidate ignored the call (for a variety of reasons).
7. The candidate accepted another offer.

Some of these reasons are benign and others are showstoppers. The last two options are no-go situations. In the other cases that are the candidate's fault or lack of judgment, I would reschedule with prejudice: another screw up and it's over.

I cannot recall ever getting a busy signal, but I would handle it the same as getting no answer.

Obviously not

Often you realize during the interview that the candidate will not work out. Find a way to get consensus such as pressing the mute button and asking the team whether they agree.

> **The mute button is your friend**
>
> It seems inexperienced phone interviewers resist or are afraid to use the mute button for fear of screwing up, or that the candidate might find out, but you can only learn to use the mute button with practice.
>
> Use the mute button when nonverbal communication won't do. For example, when you need to communicate a complex comment to the interview leader, suggest a question that would be difficult to write down, or when you need to reach consensus among the team on an issue.
>
> To avoid confusing the candidate, use the mute button when he is responding to a question—preferably an open-ended question that requires a time-consuming answer. Be sure you are prepared to speak concisely to your fellow interviewers because you may not have much time.
>
> You may want to use a protocol regarding the mute button. For example, only allow the interview leader to mute the phone. You request a mute with a simple hand signal of pushing down a button with your hand near the phone. This works well in practice, but you may want a more open approach. If anyone can put the phone on mute, they should use the hand signal and make eye contact with the interview leader to get approval before muting.
>
> If the candidate should stop talking and you must to respond before you have finished saying what you needed to say, give precedence to the candidate. Unmute the phone in a deliberate way so everyone knows they are about to be heard, and then let the interview leader continue. With good open-ended questioning, another opportunity will arise shortly.
>
> If the mute button is used more than two times in an interview, then it should be a point of discussion during interview evaluation. I have found that the two most common reasons to use muting more than this are because there is a difficult issue regarding the candidate, or the interview leader is struggling. In the later case, the rest of the team must be as supportive as possible without disrupting the interview.

If consensus is that the candidate does not have a chance of being hired, the goal of the interview team should be to shorten the interview as respectfully as possible. The ideal result is a shortened interview and a candidate that has gained a respect for your company and desires to work there.

As mentioned before, work within the standard agenda but shorten segments where you can. Using the agenda presented earlier, the earliest you are able to shorten the interview is in the exploration phase. Here are some suggestions for shortening each step of the interview.

The exploration phase is the largest and ripest for shortening. It is also within this phase that you will realize the candidate will not work. In rare situations—usually severe communication problems—you know the candidate will not be hired at the beginning of exploration. In that case, pick one of the most recent jobs to begin discussing. If the latest job is a few years or more in length, start there and say something like, "We are most interested in your recent work. . . "

More likely, you will be in the midst of the exploration phase when you realize the candidate will not work out. In this case, your best tools are compression and skipping. Compress your questioning, remembering that you are trying to make an honest show of the questioning. Seek a balance between shortening the interview and leaving a good impression.

Another good compression technique is to be judicious in the use of open and closed questions. Try to narrow the questions to reduce the length of the responses. The savvy candidate might pick up on this so avoid a closed-only approach. Again, seek the appropriate balance.

Skip jobs whenever possible. Good choices are jobs of less than six months in length or jobs that have little relevance to your position. It is a good idea to state a reason for skipping a job so that the candidate is not surprised or put on the defensive. The point of carrying on at all is to leave a positive impression of your company. Often candidates will suggest your company to friends or will re-apply when they have more experience. I have been happily surprised many times by the outcomes of positive interviews where we did not make an offer.

Together, the company and position overviews take about five to ten minutes and you have total control over them, so this is a good opportunity to cut some time. I have eliminated one or the other but not both. If your company is well known, skip the company overview and talk briefly about the position. If you chose to talk about the company, try to do so in sound bites without too much elaboration. With a little creativity, you should be able to shorten this phase to three to five minutes without being too obvious.

The Q & A phase is the candidate's time, so be careful about shortening here. Let the candidate ask all the questions she wishes. The candidate controls the questioning but you control the answers. Do your best to be concise without being terse. Avoid elaborate answers; give the necessary facts and no more. Be compliant, patient, and respectful as always. This phase is difficult to shorten so the emphasis should be on leaving a good impression. Do what you can to shorten your answers without being rude.

The closing does not provide any opportunity for shortening. Follow through with the usual closing script for all candidates. This is your final chance to leave the candidate with a positive impression of the interview. Don't give any false hopes but do show the same respect you would the most desired candidate.

In rare cases, when you know very early that the candidate has no hope of an offer, it may be best to simply terminate the interview without pretentions. For example, if you are looking for a senior Java developer and the candidate has one year of experience with Java, then it is best for everyone to end the interview. Explain to the candidate that the requirements of the position are not met and to avoid wasting each other's time, state that you wish to terminate the interview. Thank the candidate and end the call.

Some candidates will ask what it is they should work on to qualify for a job. This is a great opportunity to end the interview on a positive note and to leave a positive impression. Answer the question as if you expect this candidate to return in a few years qualified for the job.

Obviously so

With strong candidates, it becomes clear at some point that you want to invite them to an on-site interview. This will invariably change the tone of the interview into a sales job; it's human nature. I think it is fine to allow the tone to show your interest in the candidate, but it can go too far.

Foremost, be sure to do a complete phone interview; don't let the candidate slip through without scrutiny. You have a responsibility to the company and to the people performing the on-site interview who will be reading your comments. Notes like, "This candidate is really smart and we need to make an offer" are not helpful.

Avoid going "gaga" over the candidate. On the one hand, leaving the candidate with the impression your company is very interested is a good thing. On the other, if you go too far, you run the risk of coming across as incompetent. "Gee," the candidate might think, "if they are so impressed with me, just how competent can they be?" You don't want the candidate thinking, "These people sound desperate to me. I think I am going to pass." Be professional; being a groupie won't impress.

If you are really excited about a candidate, channel that energy into more collegial and in-depth questioning. Sell the strong points of your company and the position. I find these candidate interviews often get more personal; the interviewers might express the things they like and dislike most about working at your company. This intimacy is usually endearing to the candidate. Always be honest and never exaggerate, even when you deeply desire to hire a candidate.

The Google effect

Yes, candidates do search the Internet during phone interviews. There are a couple of ways to look at this. If you are asking a question that you feel the candidate ought to know, then Googling hurts the candidate's chances for an on-site interview.

On the other hand, if you suspect the candidate is searching for answers on the Internet, you ought to first reflect on the question you asked. Should the candidate know the answer or is it something that you would typically need to look up? Sometimes our questions just aren't very good. For example, it is probably unfair to ask an obscure programming language syntax question. In a case like that, is it wrong for the candidate to pull up the language specification? Wouldn't you hope that as an employee he would do that? Avoid this problem by not asking stupid questions!

It is usually easy to detect whether a candidate is performing Internet searches. Usually, you will detect searching by the way the candidate is speaking, as mentioned before.

I have never been in an interview where the interviewers expressed their suspicion to the candidate, and I doubt there would be any benefit in doing so. Let the candidate continue and check with the team for agreement. You may be misinterpreting the situation.

In my experience, most candidates who rely on searches to answer questions quickly demonstrate their lack of qualification anyway. Searching isn't a showstopper, but searching coupled with an inability to answer other questions usually is enough to decide to shorten the interview.

An encounter with a Googler ought to cause you to consider your questioning. If you repeatedly ask searchable questions, are you really asking

A candidate comes clean

During one phone interview, the candidate told us he was looking something up on the Internet and referred to the document he was reading. The question was a good one about language features that some candidates do not know. It is reasonable to expect a decent programmer would review the concept before using it, and that was what this candidate was doing before answering the question.

After briefly reviewing the page, the candidate proceeded to expand on what he read and cited a few examples where he had used the feature. He explained that it had been a while since he had used the feature and that he needed to jog his memory. Once he did, it was clear he had actually used and understood the feature.

useful questions at all? This is a great topic for the interview evaluation.

Candidate talking too much

Sometimes you just can't get a word in edgewise with a candidate. Nervousness is a common cause of excessive talking, so look for ways to make the candidate more comfortable. Much of what I have been describing will help here: setting expectations, following an agenda, minimizing confusion and surprises, drilling down gradually with your questioning, and being respectful, all aid in easing the stress involved in an interview. If stressed or talkative candidates are common, reflect on how you are running your phone interviews to see if you are promoting stress.

Sometimes excessive talking is an unconscious defense mechanism against a lack of self-confidence. This can be a serious liability for a candidate so look for other signs of low self-confidence and low self-esteem.

Excessive talking can also be a strategy for controlling the interview. This is pretty serious, because it either means the candidate is controlling the interview to ensure you don't realize their ignorance, or they simply have a controlling personality. In either case, be concerned and keep your radar tuned for other danger signs.

In any case, do not let the candidate control the phone interview. The interview leader must control the interview to ensure the agenda is followed and necessary questions are answered. Your attempts to regain control of the interview will often shed light on the candidate. The smart candidate will get the hint and relinquish control. The unconfident candidate may let go or may be too fearful to let go—both outcomes are informative. The control freak will fight and loose the job.

Attempt to gain control through escalated levels of assertiveness. The first level is to try to insert yourself into the discussion when the candidate pauses between ideas or sentences. This doesn't always work. I would swear that some candidates could breathe through their ears because they seemed to never stop long enough to inhale.

The next level of assertiveness is to politely interrupt the candidate. Start with an "Excuse me." If that doesn't work, try, "Excuse me Mr. Jones." Something about the formality of the last name is disarming. Because of the nature of telephony, you may be required to trample over the candidate mid-sentence with a long and steady plea: "Excuse me Mr. Jones. Excuse me. Excuse me, please."

Most candidates will get the hint after a few interruptions. The nervous and the intimidated will tend to back down. The clueless and the controlling may continue. At some point you should interrupt the candidate and explain the situation. Tell the candidate you are unable to ask the questions you need to ask because he or she is not allowing you to run the interview. Ask the candidate to provide succinct answers and to allow you more opportunities to speak. If this doesn't work, I would suggest terminating the interview immediately. Simply state that you are looking for candidates who are able to communicate effectively and that this interview has revealed an unacceptable communication issue. Thank the candidate and say goodbye as respectfully as possible. Follow up with the sponsoring recruiter if there is one.

One approach some recommend is to manage the excessive talker by asking narrower questions. This could work, but requires a high degree of skill to ensure that the interview team is getting useful answers. Narrowing the questioning may merely result in less information. I suggest trying the other techniques before resorting to changing your questions.

Interviewer talking too much

If the interviewer is doing all the talking, then the interview team is not getting information about the candidate. Some interviewers are nervous or lack confidence—just like some candidates—and talking is their way to cope. Some interviewers use the opportunity to demonstrate how intelligent they are, a symptom of low self-esteem or a controlling nature.

It is up to the other interviewers to decide what action to take. Is immediate action required, or can it wait until after the interview? Imagine being an interviewer on a call with a candidate you are interested in for your team. The interview leader is blathering about the right solution to a problem the candidate faced and you are not learning anything about the candidate. This calls for immediate action. The best thing to do is mute the phone and politely tell the interview leader that he or she is doing too much talking and to please refocus on what the candidate has to say.

On the other hand, if the candidate is not so strong, or you are getting sufficient information, let the issue rest until the interview evaluation. In any case, insist on an evaluation and be sure to bring up the issue. Do so with respect, focusing on skill improvement. Be sure to explain why you feel the problem exists and say what it is you want the outcome to be.

If the person in question continues with the behavior in later interviews, it may be best to discuss the issue with a manager responsible for the hiring process. Incompetent interviewers cause mass destruction for the company in reputation and ability to hire well. It cannot be tolerated.

Too many distractions

It is nearly impossible to perform a quality phone interview in the face of ongoing distractions. Do everything you can to minimize distractions on your end. Ensure a quiet place to conduct the interview and keep interruptions to a minimum.

If people regularly stick their heads into conference rooms to see if they are free, maybe your company can benefit from better scheduling tools and communication about conference room protocols. For example, if the door is closed, assume the room is in use.

Do not tolerate interviewers going in and out of the interview to handle issues or take calls. Make it clear that a phone interview takes precedence over nearly everything except the Web site crashing or the building burning down. Discuss these sorts of distractions in the interview evaluation so that participants are aware of the disruption they cause. It also gives the person a chance to explain their action as it may have been justified.

If an interviewer does have to leave, they should do so as unobtrusively as possible and they should try to let everyone know why. If it is truly an emergency, then it is acceptable to interrupt the interview and explain to the candidate that you must leave to handle the situation. Apologize and leave quietly. If the interview leader must leave—and it had better be for a good reason—then there must be an explicit hand off of responsibility that is transparent to the candidate.

It is critical your organization build a culture of accountability around the interview process. This will go far in discouraging distractions or other negative behaviors.

If the distraction is on the candidate's end, the best thing to do is to discuss it. If there is background noise, ask the candidate to find a quieter place. Call back if you have to but be sure to get a number from the candidate before hanging up. Chances are fair that the candidate will have trouble reaching you, especially if you are in a conference room.

I remember one call where the candidate had to borrow a cell phone because his had run out of charge. We had to call back because the connection was bad, but we forgot to get the new number and didn't realize our mistake until after we hung up the phone!

I have been on calls where there are screaming kids or barking dogs in the background. In one case, the candidate let the dog into the yard and we proceeded without further interruption. In the other case, the candidate explained that he was home with his infant child and his wife was out. We rescheduled the call to provide the best environment for the candidate.

Candidate is not answering the questions

It is very common that a candidate will not entirely answer questions. There are many possible reasons for this. Fear and stress can cause all sorts of strange responses. The candidate may not understand the question. Nervous candidates often lose track of the question in the midst of a long answer to an open-ended question. The candidate may not know the answer and is not willing to admit it or thinks he can talk a good enough answer. The question may be poorly phrased, confusing, or excessively complicated.

The interview leader can follow an escalation of assertiveness to get complete answers:

1. Repeat the question, possibly mentioning that you felt it was unanswered.

2. Rephrase the question.

3. Ask the candidate if she understood the question and rephrase if necessary.

4. Ask the candidate if she does not know the answer. It is important that a candidate (and future employee) be able to admit lack of knowledge.

5. Be blunt: "You seem to be avoiding answering the question. Why is that?"

Going to extremes to get an answer only makes sense for important questions. Be conscious of the risk of simply moving on to the next question. If the candidate demonstrates a pattern of avoidance, this is very serious. Consider agreeing to shorten the interview.

4.13 Conclusion

It seems to me that phone interview skills are the hardest of all the hiring skills to master. There is nothing better than a combination of study, observing skilled interviewers, learning by doing, and reflection on team performance. Begin the learning process as an observer. Work your way up to interviewer. Then, when you are ready and have the support of the team, try your hand at interview leader.

Always seek feedback from your peers and participate in hiring discussions. Ask recruiters to solicit feedback from their candidates on the quality of the phone interviews. Consider asking those you hire what their phone interview experience was like. They may give you some penetrating insights to how you conduct your phone interviews.

Chapter 5

On-Site Interviews

I chose to devote more pages to résumé reviews and phone interviews because I believe this is where companies struggle the most. That said, the on-site interview is critically important because it is the first time you and the candidate meet face-to-face.

A good *phone* interview results in two important decisions: the interviewers decide whether the candidate is likely to be a good fit, and the interviewee decides whether to pursue your company. The *on-site* interview is intended to go deeper in an attempt to validate and reinforce these two decisions to help you decide whether to make an offer. More specifically, you are trying to determine whether the candidate:

- has the desired technical skills and aptitude you demand,
- has the general behaviors and values you desire,
- will be comfortable in your work environment and culture,
- and has a personality that will fit on the hiring team(s).

Remember the dating analogy. You've caught each other's eye (the résumé); you've gone out together (the phone interview); and now you are getting serious—talking politics and religion, meeting the family (the on-site interview)—you are trying to decide whether to make a serious commitment (an offer).

The quality of phone interviews can be measured by the percentage of offers made after on-site interviews. If offers are made to 50% or more of on-site candidates, the phone interview process is working. If that percentage gets too low, the phone interview process is not screening out undesired candidates well enough. On the other hand, an offer rate over 75 or 80 percent might indicate that good candidates are not getting through the phone interview process. Your organization may prefer more thorough phone interviews resulting in higher offer rates to keep costs down, or may prefer a lower offer rate to let more candidates through. Getting the staffing team to adjust their expectations requires careful messaging and monitoring of the process.

The quality of the hiring process can be measured by the long-term satisfaction with the candidates who are hired. Expect satisfaction rates of 95 percent or better for a good process. Also consider the degree to which employees are satisfied with the company and how well it meets their expectations. This is a sort of "truth in advertising" check.

5.1 Principles

The principles for on-site interviews are the same as for phone screens, with a few additions:

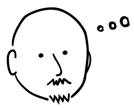
- Minimize interruptions—Focus all your attention on the candidate. Work can wait.
- Reserve interview rooms—Be sure the rooms are prepared and available when the candidate arrives.
- Be hospitable—Attend to the comfort of the candidate. Provide, water, coffee, rest room breaks, and a safe place to leave luggage or a jacket.
- Empathize—Be mindful of the candidate's state of mind. If you detect excessive stress, for example, is there anything you can do to mitigate that stress?
- Communicate well, physically—Shake hands, keep eye contact, and be mindful your body language.
- Communicate well, verbally—Use proper grammar, speak at an appropriate speed, communicate clearly and concisely, and avoid filler words.

- Represent your company well—You are an emissary.
- Be honest—It's worth repeating. Be honest.

Treat the candidate at least as well as you treat your employees. If you are faking it, the candidate will pick up on it. If you treat the candidate poorly, she will probably lose interest in your company. Whether you make an offer or not, every candidate should want to work for your company.

5.2 Logistics

Once a decision has been made to invite a candidate, follow through quickly by scheduling the on-site interview. Time is often of the essence. The candidate usually has other interviews scheduled and may even have offers in hand. By moving through the process quickly, you keep the candidate engaged. If the candidate has other offers, he will usually delay making a decision to capture one more offer. An additional benefit of promptness is that the résumé and phone interview is still fresh in the minds of the interviewers. This leads to a better on-site interview.



Schedule the on-site interview as soon as possible after the phone interview.

The process coordinator contacts the candidate or candidate's representative to make arrangements for the on-site interview immediately after the phone interview. This may require arranging flights, hotels, and transportation for out-of-town candidates. Make the process as easy as possible for the candidate. A well-executed interview leaves a strong positive impression, and that impression begins and ends with the results of good planning. You don't want the first or last impression of your on-site interview to be botched travel arrangements.

Selecting the day of the interview involves balancing the candidate's schedule along with the schedules of the interview participants. Therefore, the coordinator needs to know who is hiring, who should meet with the particular candidate, and what scheduling constraints the interviewers have. This is a complex enough problem that mistakes are inevitable. The interview team needs to be flexible and take necessary measures to ensure a smooth interview for the candidate.

Once the candidate has committed to a day, the process coordinator sets up the appropriate meeting invitations, and the invitees respond quickly so that the coordinator can make any necessary adjustments.

Occasionally someone other than the process coordinator might schedule the interview. If there is a sense of urgency and the administrator is unavailable, then someone else should act. If the candidate is highly desired and there is a perception that developer contact would carry a positive influence, let a senior developer initiate contact. Or if the candidate is a personal friend of an employee, it might be helpful for the employee to initiate the process. The idea here is to move quickly and give a personal touch to the hiring process.

5.3 Interview schedule

The on-site interview should last about four to five hours and include a lunch. The actual time will depend on the number of interviewers, the candidate's availability, whether the candidate is local or flown in, and the seniority of the candidate. Spend more time on senior candidates and those that would have to be relocated.

I like the interview schedule to begin with a one-hour technical interview given by two very senior developers. Then there will be some number of team lead interviews; these seem to work best when there are two hiring interviewers together. Take the candidate to lunch with two or three other interviewers, possibly managers. The on-site schedule that the coordinator sets up should look something like this:

1. Technical interview—One hour with two interviewers, drawn from your strongest developers.

2. Lead interviews—A few forty-five-minute interviews with two team leads in each interview.

3. Lunch—Two or three people, often including a director, taking the candidate out to lunch.

4. Lead interviews—One or more, depending on how long your interview process is.

> **Extended interviews**
>
> When hiring for senior management or executive positions, it is common to hold multi-day interviews. I worked for a small company that did the same for its senior developers. We were a remote group consisting of fourteen employees working on complex, bleeding-edge software. It was critical we had the right chemistry as well as very strong technical ability.
>
> We held a highly technical, eight-hour interview that included lunch. After the candidate freshened up, a few of us took the candidate to dinner. The following day, we invited the candidate in a bit later—after we'd had time to debrief everyone—to finish up the interview. The candidate was on the way home after lunch.
>
> This may seem like overkill, but for our situation it was worth the effort. We were a small team; we did not hire often; and when we did, it was critical we had the right person.

5. Close—A technical executive or a director does the closing with the candidate.

6. Post-meeting—Interviewers decide whether to tender an offer.

This schedule works well but consider different arrangements. The entire interview should be technical, but the technical interview is more focused. Do it early while the candidate is fresh. It also tends to be stressful, so get it out of the way.

Start in the morning, but not too early: 9:30 or 10:00 is fine. It may be more practical to take the candidate for dinner rather than lunch if travel constraints cause the interview to start later than this. Flexibility is the key, and the goal is to execute a quality interview.

5.4 Safety in numbers

Some experts say it is too intimidating to have more than one interviewer in a face-to-face situation. My experience is that if it is more than two, it can become overwhelming, but it is in your interest to have two.

As mentioned before, the person asking the question often misses important information. The second person in the room can listen and observe while the first is talking. She can detect voice modalities that might indicate stress or uncertainty. She can observe body language that might be saying, "I feel threatened" or "I am confused."

The second interviewer can detect when the first interviewer is being unclear and can follow up to add clarity. The second person can nudge the interview in a better direction in response to feedback she sees but the first interviewer does not.

When there are two interviewers, they can switch roles: the listener asks the next few questions, while the other listens. This avoids those awkward moments when the single interviewer runs out of questions to ask and has to buy time to think of something to say. Switching works especially well as two interviewers grow accustomed to each other. They begin to anticipate each other and compliment the lines of questioning.

Having a second interviewer adds validity to the evaluation of the candidate. A lone interviewer, beyond missing information, may be a weak interviewer and thus distort the interview process. It is difficult to manage overall quality if there are no checks and balances during interviews.

With two interviewers, mentoring can occur. The pairs can reflect on the quality of the interview, thus giving a strong feedback loop for improvement.

I would avoid having more than two interviewers as that does tend to get overwhelming for the candidate. If the candidate gets rattled, he will not do his best and you might miss the opportunity to hire a great employee.

Most intimidating situations are that way because the interviewers make them so. Pair interviewers should take pains to put the candidate at ease. Crack a joke. Tease each other. Do something to break the ice. Monitor the candidate and adjust accordingly. Watch for speech patterns: awkward and long pauses on simple questions, repetition, stumbling over simple phrases, unnatural vocal variation, or talking too fast. Look for physical signs: fidgeting hands, rapid movement, dry mouth (how fast did they drink the water you gave them?), sweaty palms or other perspiration.

Stress is natural in an interview but it shouldn't persist at a high level. Monitor the candidate and help him to relax. Remember that the work-a-day routine is nothing like an interview. If you can help the candidate relax, you will get a much clearer picture of him.

If you cannot get a candidate to settle down during the interview day, that is a red flag issue. Is this stressed-out person going to be able to function on

Dysfunctional introverts

Joke: How can you tell if a developer is an extrovert? He looks at *your* shoes when he's talking.

As with most stereotypes, there is some truth to the notion that many software developers are introverted or otherwise lacking in social skills. Some developers are so introverted that they become a liability in a collaborative environment.

A one-hour session with two interviewers is likely to be more than the dysfunctional introvert can handle. That's fine because he is not the candidate you want to hire. Most socially adept people will handle a two-on-one interview just fine.

When hiring, you don't have to settle for dysfunctional introverts, or even the excessively shy. In fact, these behavioral traits tend to prevent an individual from becoming a great developer. There are plenty of developers with solid social and communication skills. Hire them!

a daily basis? The uncontrollable stress is there for a reason. There is some fear the candidate cannot manage and he probably will not be able to manage it on the job either. Although the daily work routine is not as stressful as an interview, it does have stressful moments when people must perform.

5.5 Planning

Although the process is essentially the same for every candidate, each one is different, and therefore the on-site requires some planning. If you are prepared, you can handle the differences the candidate brings.

It is a good idea to hold a brief planning meeting that includes the participants in the on-site interview. It should not take more than fifteen minutes. In some situations, it is reasonable to avoid the meeting by agreeing to follow "what we did yesterday," or by sending an email. If the interview team is "in a groove" and/or everyone is experienced and understands the process, then the pre-meeting can be skipped. Sometimes, informal coordination by a point person is sufficient. Things to consider in a pre-meeting are:

- Review the positions available to this candidate.
- Review the job-level expectations.
- Review the schedule and make any last minute adjustments.
- Define how to communicate among interviewers as the candidate transitions through the interview process.
- Identify areas from the phone interview or résumé that need special attention.
- Coordinate technical questions, *etc.*, to avoid repetition.
- Identify any special handling for the particular candidate (*e.g.*, we know we want the candidate, focus on selling the company).

Clearly, many of these points can be covered by having an understood process and through email communication. The last three bullets, though, can benefit from planning before the interview. I suggest starting with a meeting and then abandoning it if the staff finds it unnecessary. Do, however, define a process and use electronic communication to express expectations and schedule.

5.6 Starting off

First impressions are critical, so it is important your team is ready to greet the candidate and execute a flawless interview. The coordinator ensures someone is there to meet the candidate when she arrives. If you have a receptionist, then that person should be expecting the candidate, and have the name and number of the person who will meet the candidate.

Whoever is first to interview ought to meet the candidate and take her to the interview location. Actively preparing the candidate for the interview, negotiating your company's security, and leading her to the interview room is an opportunity to build trust and comfort. On the other hand, if a receptionist dumps the candidate into the interview, not only is the opportunity lost, but also it may put the candidate on the defensive. She might feel as though she is being thrown to the lions.

Start the interview off with a good first impression. It will set the tone for the rest of the day.

The interviewer who picks up the candidate should briefly explain the schedule for the day. It is often advisable not to mention how many interviews or whether there will be a lunch. This leaves the interview team with some options if the candidate does poorly. For example, after pleasantries, the interviewer may explain the day as follows: "First you will participate in a technical interview lasting an hour. Then you will visit with some of our team leads. At the end of the process you will meet with a manager who will close the interview."

On the other hand, you may decide that lunch will occur regardless of how the candidate performs. Whatever course of action taken, you want the candidate leaving the interview with a great impression of your company and a desire to work there. That can justify a lunch but maybe only with two employees instead of three.

5.7 Technical interview

The primary purpose of the technical interview is to probe the depths of the candidate's technical abilities to see if they are sufficient for the job opening. It turns out there are several secondary purposes, as we shall see.

Although the entire interview may be technical, it is good to set aside one part of the interview that is even more technically focused. The technical interviewers must be very strong technically and behaviorally, and also be skilled interviewers.

I like to do the technical interview first for a few reasons. If the candidate clearly does not match your needs, you have the opportunity to gracefully shorten the interview schedule. In addition, because the technical interview is the most stressful part of the process, it is considerate to get it out of the way quickly. Third, the candidate tends to be fresher in the morning and is likely to represent herself best. Finally, since the technical interview is difficult and stressful for the candidate, warning signs often emerge; these can be identified for other interviewers to explore.

A centerpiece of the technical interview should be some sort of test to measure the candidate's technical ability. This could take the form of a coding sample, a design problem, a series of probing questions, or a written test. If time permits, some combination of these is best. Whatever challenge you choose, it should be relevant, measurable, and resistant to gaming.

The challenge is relevant if the candidate's performance is an indication of how well he will perform in the job he is applying for. The challenge is measurable if you can compare candidates with some level of confidence. The test need not be quantitative but there should be enough information to confidently rank the candidate's ability. A challenge is resistant to gaming if it cannot be prepared for, nor can the candidate detect external clues that would allow him to solve the challenge.

The technical interview can affect the agenda for the rest of the day. Here are some things the technical interviewers can pass on to the rest of the interview team:

- Areas not covered that should be explored by others.
- Behaviors to watch for.
- Advice on affinity to a particular team.
- Adjustments to the interview schedule. For example, the candidate might be suited to a team whose lead isn't participating. A swap could be made with another or the lead could be included for lunch.
- Whether to sell the position over asking questions, if it is obvious you want to hire.

The result of each interview is a recommendation to hire or not. You may recommend shortening the interview process if the candidate has no hope of being hired. Update the tracking system with your decision, along with comments, observations, and suggestions for the other reviews.

The duration of the technical interview should be an hour unless there is a strong reason to make it a little longer. Two hours would be a strain on the candidate and probably would have diminishing returns.

Let's look at some of the ways to challenge a candidate in a technical interview.

5.8 Coding problem

My preferred technical challenge is the coding sample. The candidate is presented a problem to solve in code on the white board or on a computer. Although solving the problem correctly is important, there is much more to be gleaned from the exercise:

- Whether the candidate can use the language of interest.
- How the candidate solves problems—the candidate should be encouraged to talk through the solution.
- How the candidate handles stress.
- How the candidate responds to suggestions that may require abandonment of her ideas.
- Communication and presentation skills.
- How the candidate collaborates to solve the problem when presented with suggestions.

The code sample should be sized to take about fifteen to twenty minutes for a good candidate to solve. It is important to allow the candidate to reach a conclusion even if it takes more time and more assistance. Eventual success sets a positive tone for the rest of the interview. When things get rough for the candidate, it is important to protect his self-esteem.

One surprise with the code sample challenge is just how hard it is to select a good problem. Beyond the criteria mentioned above, a code sample should be:

- Sized to be completed in a short amount of time.
- Sized to fit easily on a white board.
- Challenging at the desired skill level.
- Related to the job.
- Obscure enough that the candidate isn't likely to have solved it recently.

- Solvable without obscure knowledge, *e.g.*, not a trick question.

These criteria are quite hard to meet, and you will find it a serious challenge to come up with additional quality problems. When you think you have a solid programming problem, be sure to test it by having some of your people solve it to see if it will work.

Relying on just one problem is a dicey proposition. Word gets around on the question you ask. There are Web sites that candidates post interview questions to, and sometimes include solutions.[1]

There is an art to conducting a successful code sample with a candidate. The outcomes are nearly as varied as the number of candidates. Here are a few things to help conduct a successful coding challenge:

- Clearly describe the problem.
- Provide sufficient code, examples, test cases, *etc.*, so that the candidate can understand the problem.
- Encourage the candidate to think out loud and ask questions.
- Do what you can to make the candidate comfortable.
- Give suggestions judiciously; provide assistance without solving the problem for the candidate.

The last point deserves elaboration. If your coding problem is a good one, some candidates will struggle. You might be tempted to rescue the candidate, but be patient. You will learn much more if you allow the candidate to struggle a bit first. Give the candidate a chance to self-correct.

The time to assist is when it appears the candidate is stuck and unlikely to recover without a lot more time expended. Give advice that is sufficient to get him unstuck but doesn't give away the solution. The tendency is to be too generous; strive to let him work out the solution with the minimal amount of help. Increase the help as necessary if the candidate continues to struggle.

After conducting about a dozen coding challenges, you begin to get a sense of when and how to assist. You will be familiar with the typical sticking points of your coding problem and the hints that will be most useful to you and the candidate.

[1] For example, see `www.glassdoor.com`.

You will occasionally have candidates that will require a lot of help to have any hope of solving your code sample. You will learn how to appropriately ramp up the assistance while not doing emotional harm to the candidate. Be patient, be kind, and try to give opportunities for small successes regardless of how much you must feed the candidate.

Using a computer

If you prefer that the candidate use a computer rather than a whiteboard, then you must consider some issues. The first is the problem of the editor. Rich interactive development environments (IDE) are powerful tools that make good developers much more productive. Unfortunately, weak developers often use the IDE as a crutch so as to avoid mastering the programming language. This can hide the candidate's true knowledge.

Consider having the candidate use a text editor rather than an IDE. Require the candidate to compile the program without automated tools. Give the candidate the opportunity to choose the editor of her choice. This brings up the second consideration. Sometimes the candidate is not versed in the simple tools you have available. You have to determine whether this is a disqualifying event. Is the candidate required to use those tools? If so, they do not meet the job requirements.

You will want to observe the candidate as she uses the computer, for all the same reasons you observe using a whiteboard. Try to make the endeavor as interactive as possible. This brings up another consideration. Many people become quite nervous when being watched at a computer.

Try to make the task as collaborative as possible. Rather than standing over the candidate, pull up a chair. If you use pair programming, set yourself up as a pair and role-play a junior developer. Have the candidate teach you as she solves the problem.

If a computer approach fails for a candidate, you always have the option of using the whiteboard. Try not to make the situation demeaning; keep the candidate's self-esteem intact and don't get frustrated.

As you conduct these technical interviews, you will develop criteria for evaluating the candidate against your challenge. If the candidate really struggled with the problem, there would probably need to be exceptional circumstances to recommend the candidate for hire.

5.9 Design problem

If you are hiring senior developers, you probably expect them to be competent in software design. Guidelines similar to the coding problem apply here. The problem should be of the right size and complexity, and should be relevant to the job.

Selecting a design problem is much easier than selecting a coding problem. One approach is to use a problem you have been working on lately that you are intimately familiar with. Sometimes that problem is especially tricky. If so, let the candidate know it is a hard problem and that you don't expect a perfect answer; you are more interested in the thought processes.

Another approach is to pose a design problem drawn from your product line. You could pose a code design, database schema, or an architectural problem. By drawing from your company's product line, you ensure relevance, but be sensitive to the tractability of the problem.

When describing a design problem, indicate how much detail you want, and how you want the candidate to express the solution. For example, you may only want to hear a verbal description of how he would approach the problem, or you may desire a UML[2] class or sequence diagram.

Define the scope of the problem. Is the problem to be solved in the context of a class, a library, an application, or a system? I've noticed that candidates who do poorly on design problems usually struggle with understanding more than the actual problem.

If a candidate doesn't do well on a design problem, it isn't always as revealing of ability as a coding problem. Design problems often are domain-specific and that could be the reason the candidate struggled. If you choose your design problems to be small and diverse, you can ask a few of them to get a better sense of the candidate.

5.10 Written test

Written tests have many problems. I include it because many companies use them, and it might make sense for your company. One challenge is to keep the test up to date and relevant to the positions for which you are hiring. Tests tend to get ignored and become out of date. If you implement written tests, review them regularly.

---

[2]Unified Modeling Language.

Another problem with written tests is that they don't provide much information. All they tell you is whether the candidate knew the answers off the top of her head. You do not get to interact with the candidate or observe her in a real-life situation. Work is nothing like taking a test.

Yet another problem with written tests is that, depending on when they are taken, you do not know whether the candidate actually took the test. If you ask the candidate to fill out the test online or in the privacy of their home, it is impossible to know who took the test or what resources were used.

In the context of an on-site interview, I've seen companies use the test *as the technical interview*. Every time I've encountered this, it has been accompanied by other patterns. The interviewers were not on time, they were not prepared, they were not very technical, and there were only a few of them. In other words, the test was being used to replace human interviewers because the company failed to commit the necessary resources to the hiring process. Remember the agile principle: people over processes.

So when would a written test be helpful? If the job requires specific knowledge, and that knowledge lends itself to a written test, then it might be justifiable. On the other hand, you could always just ask the questions and that would be better. Furthermore, if you can capture the essential parts of the job in a test, why are you wasting money trying to hire a senior developer?

I've seen written tests take many forms. I don't like any of them. I would avoid them unless you can come up with a very good reason to use them.indexon-site interviews!tests, during interviews|)

5.11 Technical questioning

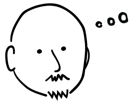
Any technical interview would be incomplete if it didn't include some plain old technical questioning. Therefore, it is important to leave some time for questioning that fills in the gaps not covered by other technical challenges.

Many of the guidelines for good questioning presented in the chapter on phone interviews apply here. As interviewer, you probe an area by asking a series of questions that become increasingly more detailed. The critical differences are:

1. You have the all the modalities of communication available.
2. You can clearly explain more complicated problems.
3. You can use props such as paper and pencil, white board, or computer.

4. You don't need to ease into the questioning at this point.

Let the job description and your environment's technical constraints drive the questions. As with phone interview questioning, be prepared with lines of questioning that make sense for the job.



Be sure to follow up problem-solving with technical lines of questioning to fill in any gaps in your understanding of the candidate's knowledge.

As with phone interviews, it is useful to focus on what the candidate has done. The differences are a richer communication environment and a sharper focus on what is relevant to your job openings. Find work on the résumé that is relevant and dive deep. Ask what the challenges were and why. Ask about design choices. Ask about technology choices and the trade-offs considered. Ask the candidate to use a white board to illustrate responses. These lines of inquiry will reveal the depth to which the candidate was actually involved in the work listed on the résumé. Look for convictions, understanding of issues, recognition of external forces, *etc.* In addition to questioning the candidate on what she has done, be sure to cover areas you care about regardless of what appears on the candidate's résumé.

I like to ask questions in technical interviews that allow the candidate to express opinions. Sadly, many developers just develop and don't really think much about the tools they use, what they do, or how they do it. These people tend not to be inquisitive or creative. I prefer to hire people who not only think about the problem at hand, but also think about, and form opinions on, various aspects of their profession.

I also like to ask the candidate to contrast or compare technologies. This not only reveals their ability to think above the bits and bytes, but it also shows their depth of mastery in the technologies being discussed. This book is about hiring the best software professionals, but if you find yourself needing to hire less experienced employees, this type of questioning might prove less fruitful. It takes most people a few years to form an understanding of their profession in the abstract.

Having a set of target areas for questioning is great, but be flexible. One of the things that makes technical interviews challenging and fun is that you really don't know which way it will go. Get the important questions answered, but also go with the flow.

5.12 Team interviews

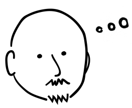
Back to the dating metaphor, team interviews are about matchmaking. It's like speed dating: the interviewers are evaluating how well the candidate would fit on their teams, while the candidate is hopefully doing the same. Therefore, much of the evaluation is focused on behavior and work style.

There are usually two or three team lead interview sessions of forty-five minutes with two team leads in each session. In a crunch, these interviews can be shortened to thirty minutes but that is tight; going a full hour is fine.

If you have a wealth of open positions, it is easy to schedule leads with a vested interest in the candidate. More likely, not all the interviewing leads are actually hiring. They are interviewing on behalf of their peers to provide additional perspective on the candidate. They should role-play as if they were hiring, and act as proxies for the teams that are hiring.

All phases of the on-site interview should be monitoring behavioral characteristics of the candidate. But in this part of the interview, the interviewers should be more specific. The interviewer focuses on the behavioral values of the organization to see if there is a good fit. If there are characteristics of the team's work or environment that demand certain behaviors, that should be a focus as well.

Although chemistry is important, so is the technical match. Team interviewers should explore whatever technical areas they feel are necessary to ensure the candidate will work out. Guidance should be taken from the results of the technical interview to avoid duplicate questions and to probe unexplored areas. The interviewer should focus on technical questions or problems that are specific to their needs.



Team interviews are about match-making: technical ability, behaviors, and personality. There is a two-way sharing of information.

Team interviewers who are hiring take full responsibility for staffing their teams and therefore should be somewhat self-absorbed with their team's needs. At the same time, interviewers collaborate with each other to create a positive atmosphere, to cover as much ground as possible, to explain the nature of their teams, and to answer the candidate's questions. As always, you want the candidate leaving with a desire to work at your company. As an organization, you want to hire the best people, regardless of what team they end up on.

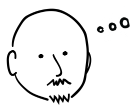
5.13 Lunch

The goal of the lunch portion of an on-site interview, beyond attending to the candidate's physical needs, is to observe the candidate in a more relaxed setting. The focus is on behavioral aspects and the hope is that in a relaxed setting, the candidate will let down his guard and let his personality show through. The interviewers try to remove pressure and build trust. Small talk about hobbies, recreational interests, or the work environment are helpful icebreakers.

If the candidate is out of state, talk about living in your locale. What is the housing market like? Where are the best neighborhoods? How are the schools? Consider judiciously sharing some mild "dirty laundry" as a way to open things up. Hopefully, you don't have *loads* of dirty laundry!

Expect discussions about family, but try to avoid initiating these discussions. You don't want to appear to evaluate the candidate on issues not related to the job. On the other hand, don't be evasive. If the candidate talks about family, don't hesitate. Share within reason; give and take is a great way to build some trust.

Lunch is an opportunity for everyone to be themselves. I like to have a mid- or senior-level manager attend the lunch. She can answer a variety of questions that the other interviewers cannot. If you have many open positions for this candidate, lunch gives you an opportunity to expose the candidate to additional hiring leads.



Lunch is an opportunity to encourage the candidate to be herself. She will usually relax and let her real personality show.

Send at least two people to lunch with the candidate; three seems to be the ideal number. It is easier to foster an informal atmosphere with three, less so with two. Four or more can be overwhelming.

Lunch is an area where you can save time and money if you know the

> **A quiet lunch**
>
> It is important to send the right people to the lunch interview. I was once scheduled along with two extremely shy team leads. I am a fairly outgoing person and I was able to engage the candidate, but there were times when an uncomfortable silence fell over us. I'd wait for one of the other two interviewers to say something and, when it became clear they weren't going to talk, I would try to start a new conversation. Needless to say, I did not enjoy my meal. The interview went well; I believe we hired the candidate. It is common to have to help the candidate open up and talk, but I was in the position of also trying to get the other interviewers to say something!

candidate is not going to get an offer. If the candidate is local and you haven't told him about lunch, you might be able to move to the closing of the interview if it is around 11:30 or 12:00.

If the candidate is out-of-state, or if it would be awkward to skip lunch, consider sending only one or two people to lunch. Be sure the lunch interviewers know the score. If leaving a good impression weren't important, you'd just send the candidate home, but you don't because a good impression can yield benefits later.

5.14 Behavioral interviewing

Interviewers should be evaluating the candidate's behavioral and personality traits throughout the interview process. Doing this well requires practice and, for some, getting out of their comfort zone.

Behavioral interviewing is much easier if the organization is clear about its behavioral values. There are two parts to this. The first is what the organization says its values are, and the other is what it actually practices. As mentioned before, hire to the stated values, as they are usually loftier than the practiced values. Hire to become the organization you desire to be.

Table 5.1 lists behaviors from Appendix A along with ideas about how you might gauge the candidate for each.

Table 5.1: Behaviors and how to gauge candidates

Behavior	Questioning
Adaptable	Explore a time the candidate had to work outside his comfort zone. Ask about a time he had to learn a new language, skill, or methodology.
Accountable	Ask about a time she took the lead when no one else would. Discuss examples of leadership in a group.
Analytical	Observe how the candidate approaches solving technical questions. Ask her how she analyzed and solved a difficult problem in her past.
Collaborative	Explore the dynamics of teams the candidate has been on in the past. Discuss examples of problems that were solved (or examples of failure) due to collaboration. Pose hypothetical questions or role-plays that expose the candidate's collaborative abilities.
Detail-oriented	Observe whether the candidate highlights subtle, but important distinctions, or glosses over them. Pay attention to the questions asked by the candidate.
Decisive	Explore areas of job history where the candidate likely had to make decisions. Observe whether the candidate "owns" his responses during the interview.
Delegates	Ask about a time that the candidate was overwhelmed by workload and how he (should have) addressed it. Motivate a similar situation with a hypothetical question.
Energetic	Easily observed in questioning. Does the candidate get excited about accomplishments, challenges, or learning opportunities? Is the candidate enthused about her profession?
Entrepreneurial	Ask about a time when a risk was taken. Explore whether the candidate understood the cost/benefit or value/risk proposition. What steps were taken to improve chances for success or reduce the cost of failure? Ask about a failure and how catastrophic loss was averted. What lessons were learned that she later applied elsewhere?

Independent	Is the candidate assertive about his needs during the interview? Ask for examples where the candidate worked alone to solve a difficult problem. In what ways did he have to step out of his comfort zone?
Initiative taker	Does the candidate *appropriately* take control of the interview to ask her questions? Does she offer more detail than requested in a way that enhances the interview? Can she describe a time when she took a positive action even though it wasn't her job?
Honest	Look for examples of dishonesty or truth stretching. The honest candidate will be impeccable. When questioning about accomplishments on the résumé, does he tell a different story, describe a smaller role, or disavow it?
Innovative	Ask the candidate for examples where he thinks he was innovative and evaluate his criteria and accomplishment. Are there multiple, interesting solutions in the candidate's past? Innovative people repeat innovation.
Leader/Influencer	Referring to the résumé, ask about leadership situations (see Chapter 3 on pages 70 and 83). Ask the candidate for examples where she was able to change how people thought about something.
Listener	During technical problem solving, how well does the candidate understand the questions? During all interviewing, are the candidate's answers to questions on point, or do they miss the mark? Does the candidate allow you to finish, or does he interrupt?
Optimistic	Does the candidate have a "can do" attitude? Does she seem to relish challenges to learn new things? Ask her where she sees her future going. Does her answer seem optimistic? Optimism is not the same as blind faith.
Mentor	Ask for examples when the candidate was responsible for training another. If he was in a leadership position, ask what he did to foster personal growth in the people who reported to him. How does he respond to simple questions? Is he patient? Have the candidate teach you something during the interview.

Motivator	Ask for an example of the candidate motivating others, especially one where there was a motivation problem. Ask where the candidate derives her motivation. What does she do to motivate herself when she doesn't want to do something?
Consensus builder	Ask the candidate to describe how decisions were reached on his teams. Explore more deeply his role in the process and how consensus was, or should have been, built.
Tactful	How does the candidate respond when asked why she left a job? How does she describe former co-workers or managers? Is she diplomatic or vindictive?
Planner	Ask about how his projects were planned and how he was involved. Does the candidate have questions prepared? Has he thought through the issues of changing jobs? Ask the candidate where he wants his career to go next.
Relational	In the work stories the candidate tells, did she seem to effectively build and rely upon key relationships? In her interactions through the interview, does she seem to be actively building relationships?
Team player	Ask the candidate for examples where he was a team player. Ask him what being a team player means to him. Set up a role-play that requires teamwork to solve and observe how the candidate responds.
Pride	Does the candidate show healthy pride in his significant accomplishments without being boastful? Is the pride justified or inflated? Ask the candidate what accomplishment, skill, or personal trait she is most proud of and why.
Quality	In coding tests, does the candidate seem to care about the quality of her solution? Ask about instances when she felt she had to make a tradeoff between quality and, say, time. Explore how she felt about the tradeoff and how the outcome could have been improved.

Like technical questioning, I prefer to elicit behavioral understanding more subtly than through point-blank questioning. If you go through the list shown in Table 5.1 and ask "behavioral questions," eventually you will tip off the candidate and, if she is clever, she will tell you what you want to hear.

Instead, try to weave behavioral questions into your dialog in such a way that it seems natural and does not feel like behavioral questioning. And always, be mindful in observing behaviors during the interview. After the interview, track and discuss any quirks or concerns that you observed. It may have been an anomaly, or it may be a pattern of behavior that other interviewers experienced.

Sometimes a candidate will have left a good impression on all the interviewers but one. Listen to and reflect upon that one person who identified a behavioral problem. She may have stumbled upon a hard-to-spot problem, so pay attention. It could be the interviewer had the opportunity to observe the candidate in a stressful situation, for example.

It is a good idea to communicate behavioral concerns to the rest of the hiring team during the interview. You can do this verbally, by email, or via the tracking system you use. Giving others a heads up may allow them to pick up on a critical issue that would otherwise be missed.

5.15 The seven deadly sins of interviews

Marc Cenedella, founder and CEO of TheLadders.com, Inc., wrote an advice column for job candidates called, *Seven Deadly Sins of Interviewing—seven traits that can kill your chances with a recruiter or hiring manager*.[3] I liked this model so much that I asked Marc if I could use it to describe the interviewer's perspective.

Pride: An excessive love of self

When interviewing seasoned and successful candidates, you should expect them to display confidence in their abilities. What you don't want to see is cockiness or arrogance. These traits make an otherwise stellar employee very difficult to work with. Given the premium for collaboration, a prideful employee can be destructive to a team.

To detect excess pride, observe how the candidate treats people in service roles: the person at the front desk, the administrative assistant, or the host and waiter at lunch. Contrast how the candidate interacts with the lead developers

[3] TheLadders, "Seven Deadly Sins of Interviewing" [Cen]

against interaction with senior management or architects. Pay attention to how the candidate feels about the job opening; does the candidate express that it is below his or her stature? Does the candidate take cell phone calls or respond to text messages at inappropriate times during the interview?

Always be careful here because the candidate may be over-qualified for the position. It may be that the person is expressing confidence. It may be that you feel threatened by confident people. As with all potential behavioral issues, sample the body of interviewers to see if anyone else has concerns.

Sloth: Laziness, idleness, and wastefulness

A candidate's strong awareness of your company is good; lack of awareness should be a concern. A senior candidate ought to be familiar with the company, its public persona and finances, and its market. With today's search technologies, the only excuse for lack of familiarity is sloth.

Pay attention to the questions being asked by the candidate. They will show you how much thought and preparation were put into the interview. Many senior developers are pulling down six-figure salaries; they ought to be able to ask insightful questions of a potential employer.

Greed: An excessive quest for money and power

It is perfectly reasonable for a candidate to want more in their next job. Changing jobs is a big step and it ought to be worth the risk. That said, there is a time in the interview process for discussions about salary, bonuses, time off, *etc.*

A candidate—especially an experienced one—ought to realize that if they impress a potential employer, they will receive an offer and have an opportunity to negotiate. It shows greed, arrogance, and poor judgment in a candidate who insists on negotiating at the wrong times.

If the candidate is focused on money and perks, you should be very concerned. Greed tends to breed dissatisfaction which breeds conflict, which eventually leads to the candidate leaving to satisfy the greed.

Gluttony: The desire to consume more than you need

Gluttony doesn't always center on food. A good interview depends on building rapport between you and the candidate. You are trying to learn about the candidate's skills, talents, experiences, and most of all, ability to succeed in the job.

If the candidate never stops talking, dominates conversation, wrests control of the interview, provides excruciating detail, or liberally wanders off track, be concerned. This self-indulgent, "gluttonous" attitude will not be productive, nor will it foster collaboration.

Wrath: Feelings of hatred, revenge, or denial

It is very uncomfortable to listen to a candidate criticizing a former boss or colleagues. As I listen, several things cross my mind. What part did the candidate play in the conflict? Will the candidate take any responsibility or deny culpability? If hired, will the candidate shoot down new colleagues similar to these old colleagues? Will I be the next person to be shot down?

I am fine hearing about difficult situations that may have lead the candidate to leave, because it is part of the story. Along with that, I would prefer to hear what the candidate did to address issues. Leaving a broken situation, I would like to hear the candidate discuss what a healthy situation looks like. I want to hire people who can let go of the past and focus on today.

Lust: The desire to do what you want, not do what you should

Inappropriate dress is a concern; inappropriate behavior is very serious. I have never experienced the latter from a candidate and only rarely the former. In the software development world, dress standards vary widely. Some recommend wearing a suit to an interview, but I know of some companies that will hold this against a candidate. Because of this wild variance, do not be surprised if the candidate asks how to dress for an interview. I think this shows an awareness and sensitivity toward the market and employer. Establish reasonable dress expectations for your interview process.

A candidate can be lustful by insisting on excessive responsibility, the irrational need to work on certain technologies, or demanding a specific title. Desiring these things irrespective of the needs of the employer, is bound to breed resentment in the candidate, and frustration in fellow employees.

Envy: The desire for what you don't have now

A candidate that spends too much time talking about external material things they hope to get is a concern. Their focus is elsewhere during the all-important interview; where will it be at work on a day-to-day basis?

Often it feels like on-site candidates are window shopping. At one company, we had the architecture team do the technical interview first. At this company, the architects were the most knowledgeable and the best software developers. They also had excellent people skills and their enthusiasm for technology was contagious. It was very common for a senior candidate to come out of the technical interview desiring to be an architect, regardless of how unrealistic that desire was.

It is good to hear what a candidate's aspirations are, and they should be somewhat aggressive. You can learn a lot about a candidate by estimating how realistic his goals are and by hearing how they intend to reach them.

5.16 Closing

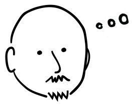
The closing is the last part of the interview and usually takes about half an hour to forty-five minutes. It is a good idea to have a mid- to senior-level technical manager perform the closing. Such a person can answer different sorts of questions, and it exposes the candidate to the management culture of your company. The best candidates care how an organization is run.

I like to cover the following in the closing:

- Allow the candidate to ask any unanswered questions.
- Provide a higher-level view of the company.
- Describe the company benefits.
- Establish candidate salary expectations.
- Confirm the candidate's availability.
- Sell, if appropriate.
- Evaluate behavior.
- Explain what to expect next.
- Send the candidate home with a positive last impression.

The candidate should have felt free to ask questions throughout the interview process, but typically there are a few final burning questions. I like to start the closing by allowing the candidate to get those out of the way. Some candidates will ask questions all night if you let them, so be sure to leave yourself time to cover the necessary components of the closing. You may have to interrupt the questioning in such cases. Try to allow time at the end of the closing, or give the candidate your business card so she can email you with her questions.

Explain your role in the organization so the candidate can gear his questions toward you. If the candidate knows you are the VP of development, he is likely to ask organizational or business questions. Expect questions about the financial standing of the company and its prospects, for example.



The closing is time to discuss salary and benefits, determine the candidate's expectations, and answer any lingering questions.

One of the necessary components of the closing is to communicate the benefits your company provides. This should be an overview and you might wish to back it up with a hardcopy that describes the benefits in more detail. Cover the medical plans, paid time off, holidays, bonus plans, and other relevant benefits. Describe any special constraints. For example, e-commerce companies in the United States often do not consider the day after Thanksgiving a holiday. *Black Friday* is one of the busiest days of the year and systems must be up and running.

For efficiency sake, be sure other interviewers understand what is covered in the closing. Let the closer focus on benefits, relocation, and salary, and let the other interviewers focus on the candidate.

It is critical to establish the candidate's salary expectations. I like to ask this after describing the benefits and bonus plans, as it provides a framework for further discussion. I find it helpful to first ask the candidate's current (or most recent) salary. I ask if there are any incentive of bonus plans in place and try to get a sense of how they work and how predictable they are.

I then ask what salary the candidate is looking for. Depending on what they answer, the discussion could go several ways. If the candidate is relocating, I ask them if they understand the relative salary ranges for the region they are leaving compared to my region.[4] If the candidate's expectations are high, I usually tell her. I will not be specific but I will let her know that her number is likely to be out of my hiring range. Depending on the quality of the candidate, I may leave it at that, or I may take the time to encourage the candidate to consider the entire package.

If the candidate's expectations are surprisingly low I do not try to take advantage. I prefer to offer within the salary ranges my company offers for similarly qualified candidates. Lowball offers have a way of doing damage later. Anyway, if you exceed the candidate's expectations, your odds of acceptance are much better!

If the candidate's expectations are reasonable, I may say so. Usually though, I will just take note and move on. I do not want the candidate second-guessing whether he should have asked for more. That may lead to frustrating salary negotiations or resentment after the candidate is hired.

A big increase from the candidate's current salary to what he is expecting, after being adjusted for different markets, should give pause. I will want to explore why the candidate feels she deserves such an increase. There may be good reasons. On the other hand, there may be unrealistic expectations driven by personal baggage. Depending on the situation, I may carefully try to recalibrate the expectations. Most often though, I find it best to simply make the fair offer and let the chips fall where they may.

What if the candidate won't share her salary information? I personally feel this is a foolish strategy and does not reflect well on the candidate. She might think she is being cagey. Shrewdness is fine, but not until some cards are on the table. I simply explain that we cannot discuss an offer without getting some sense of her expectations. She might have some discomfort about talking numbers. I make it clear we are serious about hiring her and need to begin salary discussions. The candidate may have been counseled by a recruiter not to talk about salary.[5] If you are comfortable negotiating through the recruiter, let it go. I do not work with recruiters who insist on running salary discussions. Nothing good comes of it. I think it is important

---

[4]Recruiters should cover this ground with the candidate. Regardless, I go over it too. Candidates often adjust their expectations during the hiring process. I also find that comparing the recruiter information with what the candidate later says can shine a light on process and communication problems.

[5]Some recruiters believe they should negotiate salary on behalf of the employee. They want to maximize their profit by maximizing the candidate's salary, but will settle for a lower salary since that's better than no commission at all.

to establish the candidate's expectations before she leaves the interview.

A very important step is to confirm the candidate's availability. This should have been done early in the process, but it is surprising what can come up. People are funny and some candidates will play games all the way to the end before revealing their true intentions.

There are several things I want to know. When can the candidate start? This often involves giving notice to their current employer. I think it is critical to establishing a good relationship from the start, that I allow the candidate a fair amount of time to shut things down at their current job. At a minimum, this means they should be allowed the customary two-week notice. Often though, that may not be enough. The candidate may request more time due to a critical project or situation. It builds trust to allow some flexibility here.

It may be the candidate requires some personal time. If the candidate is relocating, I first want to be sure the candidate is interested in the area and willing to move. Then I want to explore the candidate's ability to move. There are a few key questions I like to ask. "Do you own a house?" "Have you engaged a real estate agent to see what your house is worth?" "Do you have any sense of how difficult it would be to sell your house in the current market?" "Have you discussed a potential move with your family?" These types of questions give me a sense of how seriously the candidate has explored relocation. The typical job hunter makes an early decision about the scope of her search. If it includes relocation, and if she's wise, she will do some preliminary research to see what relocation would take.

If the candidate's situation is complicated, because of relocation or some other personal situation, I will ask how long the candidate needs before starting work. I weigh many factors in determining whether the timeline will work for me.

If you know you will make an offer to the candidate, then take the opportunity to sell the job. Senior management attention is a powerful selling tool, so use it. Really good candidates are likely to have other offers; you will need to use everything in your arsenal to come out on top. Often the deciding factors for taking one job over another are not financial but emotional. If a candidate feels better about one offer, all other things being close, he will take the one he feels good about.

The haunted house

For many, home ownership is a sign of success. But when it comes to relocation, selling a house is scary and there's a good chance that if your candidate owns a house, it is his first. The prospect of selling a home can be intimidating. We all expect to sell quickly and make a profit, but things don't always turn out that way. Selling a home is complex, and the homeowner has little control over the outcome.

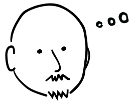
Buying a new house in a new place is also scary. When you think about the logistics of moving your belongings, searching for a home, buying a home, getting kids enrolled in school, and living in between homes. It gets overwhelming. Keep this in mind when relocating a candidate.

The house can be a nightmare for the employer too. The candidate may be trying to relocate from a depressed market or one with much lower property values than the market she is relocating to. An economic downturn makes selling a house extremely difficult. The candidate may be under water and unable to absorb the cost of selling.

I have learned the hard way not to assume the candidate has figured out his ability to relocate. Since the housing bubble burst in 2008, several hires had to quit because they were unable to sell their homes.

Some companies have the means to bridge these situations for their employees. They offer relocation packages that provide financial support. This is the exception, rather than the norm, and these packages are often reserved for executive-level employees.

Employee churn is a huge waste, and it is even harder to swallow when the mere sale of a home stands between your company and a great employee. It is a bitter pill to say goodbye to a great hire who must leave because of a haunted house.



Establish the candidate's availability, including any relocation constraints, before finishing the closing interview.

Every interviewer evaluates behaviors, but the closing provides a particular opportunity if it is performed by a senior member of staff. It creates a different dynamic because the interviewer is in a position of authority. When comparing notes later, often a big difference exists in behaviors the interviewers at a more peer level to the candidate observe, from those the senior-level closer observes. This is the difference in how the candidate responds to authority, and it can be illuminating.

Just before the closing ends, explain what is going to happen next. If your team has conducted a good interview, the candidate is going to be eager to learn how she did and whether she will receive an offer. It is so frustrating to not know when or if you will hear back from the company. You can ruin a great interview by mishandling the post-interview steps.

Tell the candidate that there will be deliberation among the interviewers to decide whether to make an offer. The message is that this is important to your company and the decision will be based on everyone's input. If you know you are going to make an offer, tell the candidate. This is a great way to finish the interview on a positive note.

Set a time limit within which the candidate will be contacted, for your sake, the sooner the better. I like to respond within twenty-four business hours and I tell the candidate that if she hasn't been contacted in forty-eight business hours, she should contact us. Be sure to provide a reliable point of contact. Consider having the closer give the candidate a business card if the candidate is likely to get an offer; this shows respect and trust and will certainly impress the candidate. Otherwise, consider some other point of contact. If the candidate is working through a recruiter, ask them to contact the recruiter.

Let the candidate know she will be contacted whether an offer will be given or not. Do not leave the candidate hanging, because what you are hanging is your company's reputation. Tell the candidate whom she will be contacted by. Use the recruiter if there is one, otherwise, consider having the closer or the hiring team lead contact the candidate. As a last resort use HR. You lose an opportunity to leave a good impression if you hand off this responsibility to someone outside the hiring team.

The closer should perform the necessary steps to conclude the interview. This involves escorting the candidate off premises; taking care of the guest badge if there is one; ensuring the pre-arranged transportation is ready, *etc*. Remind the candidate she will be contacted shortly and end the interview on a positive note with a firm handshake and a smile. If the candidate leaves with a desire to work for you, no matter how she did, you can claim success.

You may not have found a great hire, but you have improved your company's reputation, and that has value.

5.17 Evaluating the candidate

The purpose of the evaluation is to decide whether to tender an offer to the candidate. Each interviewer expresses his or her opinion in the tracking system, and should share the perceived strengths and weaknesses observed in the candidate. Express any questions you still have. If you feel strongly about it, state where you feel the candidate might fit best.

It is a good idea for the interviewers to meet soon after the candidate leaves to discuss their observations. This doesn't replace the tracking system, but provides an opportunity to "connect the dots" among the interviewers and their observations. If you are hiring a small number of people, these meetings are essential. If your team is regularly hiring, they may be able to get by with just the comments tracking system.

Salary considerations should not be part of the post-interview meeting discussion, nor are they appropriate for the tracking system. The offer (salary, signing bonus, *etc.*) should be discussed separately once the team has decided to make an offer. It is counterproductive to make assumptions about the candidate's salary expectations when deciding whether to make an offer. First decide whether to make an offer, and then decide on the compensation based on your knowledge of the market, your company's salary structure, and the perceived quality of the candidate.

As with other interview phases, define a protocol for deciding whether to make an offer. At this stage, a mere majority is insufficient. There should be strong consensus in favor of a candidate before considering an offer. Pay close attention to the concerns of the small number who are against an offer. Try to determine if they caught something that others missed. When in doubt about a candidate, consider the following options:

- If the candidate is local, there is less harm should you make a bad hire. She has lost little. An out-of-state candidate, on the other hand, has a lot to lose, and there is a big expense to bring her on-board. Insist on strong consensus for candidates that require relocation.

- Wait a day and discuss again. If there is uncertainty, time will often help establish resolve.

- Inform the candidate or their representative that you are unsure and would like time to consider. Remember not to leave the candidate hanging.
- Identify what is missing to make a decision and have a follow-up phone call or interview.

Even if the team decides not to make an offer, the candidate should be notified as soon as possible. Many companies do not inform the candidate promptly or at all. This is incredibly unprofessional and does great harm to the company image. Calling to tell a candidate an offer will not be made is a good job for the HR department.

Finally, reflect on every interview and on the interview process as a whole. The former is easy if you do it at the end of the post-interview meeting. If you decide to waive these meetings, you might have trouble reflecting on individual interviews. An alternative is to hold interview reflections when something goes wrong. For example, if a candidate comes in that is seriously unqualified, you might want to discuss how that person got through the résumé review and phone interview; you should refer to the tracking system and the résumé as you do this.

Reflecting on the overall interview process simply requires setting up a meeting once in a while. This is good to do a regular basis, but not too frequently. A good guide for frequency is rate of change. If you are trying to define your hiring process, or if you are making a significant change to it, consider meeting once a week for a while. If things are pretty stable and the results are good, meet once a month. If results are not good, consider holding more frequent meetings to decide on small incremental changes, and to monitor the effectiveness of those changes.

5.18 Tips for great on-site interviews

There are many little things that make for a good on-site interview. I cover some of them here:

Scheduling

Be sure to have the entire on-site interview scheduled a few days in advance. Book the conference rooms, send out the meeting invitations, assign the lunch interviewers, and the closer. This allows the team enough time to adjust for problems or schedule conflicts.

On the day of the interview, send out a reminder summary email that contains the whole day's schedule and participants. With this, no one should miss a beat. It is a good idea to start interviews a half hour to an hour after employees typically arrive at your office. This gives them time to see the email, review the résumé and comments, and develop a strategy for their part of the interview.

A well-scheduled interview is the basis for leaving a great impression. Executing that schedule well is guaranteed to impress your candidate. A poorly scheduled interview, on the other hand, will leave the impression your company is incompetent. Great developers don't like incompetence!

Be on time

Failure to be on time is a Cardinal Sin. You let your hiring team down, the candidate down, and your company down. It is so awkward for me, as an interviewer, to have to track down the next interview team. It is a black eye for the company and leaves an impression of incompetence. Being on time shows how much the company values the candidate. It also shows the team is not too busy and that they are able to manage themselves. It will leave an impression of professionalism.

Hand offs

Establish a protocol for handing the candidate off from one interview team to another. If you schedule one room for all interviews, insist that the next set of interviewers arrive at the room no later than the designated time, ready to start.

If you must move the candidate to another location, decide whether the current interview team takes them there, or the next team picks them up. On the one hand, the current team has built a relationship with the candidate and so the walk is likely to be more collegial. On the other hand, timing of the hand-off usually works better if the next team picks up the candidate. Experiment with it and see what works best.

Swiss precision

In 1992, my wife and I took a trip to Switzerland. Once there, our sole form of transportation would be our Swiss Rail pass. Based on my previous experiences with public transportation, I was a bit nervous. We had a tight itinerary and I had doubts about meeting it if we had to depend on trains.

After a couple of days, all my fears were gone. I have so many fond memories of the beauty of Switzerland and the kind nature of her people. But one of the more powerful memories, one that instilled great respect for the Swiss, was the fact their trains were always on time.

My impression of the Swiss is of a people who are competent and reliable. This is the kind of impression I want candidates to have of my employer and me.

Information passing

As the on-site interview progresses, information should be passed to the interviewers that have not yet met with the candidate. Typical information to pass includes:

- General impressions
- Technical evaluations
- Questions that were asked
- Questions that should be asked, or areas that should be explored
- Warning signs such as hints of strange or negative behavior
- Whether to shift to *sell mode* because the candidate is an obvious winner
- Whether to shorten the interview because the candidate is unqualified

The simplest way to pass information is to put your comments and recommendation into your tracking system immediately after your time with the candidate. Each member of the hiring team should review the tracking system just before they meet with the candidate. A good tracking system will send emails when comments are added, so it is easier to keep abreast of the candidate's progress.

If something is very important, it is best to meet with the remaining interviewers to be sure they get the information in a timely manner. It can be impractical to pass information to the interviewers that immediately follow you. Pulling them aside as you hand off the candidate is awkward and likely to leave a bad impression. You might send a text message if the information is very important, but that can backfire. The interviewer must check her mobile device, and that can be rude. Usually it is best to communicate beyond the next group.

If the interview is to be cut short, an email to the interview team should be followed with a face-to-face conversation or a phone call to every remaining interviewer. Adjust the schedule and let everyone know who is still participating. The team should be told how to handle a shortened interview. Remember that all candidates should leave your company with a desire to work there.

Sometimes it is better to avoid passing information. For example, the technical interview turns out to be very ambiguous regarding the personality of the candidate. Rather than "pollute the well," it might be better to wait and let others form opinions independent of what the technical interviewers thought they saw. Other times, you may only want to communicate partial information initially. Say the technical ability is ambiguous. You might consider entering what you do know and then suggest areas where further technical exploration should go, but you don't include a recommendation until later.

After the interview day is over, it is critical that everyone has entered all they wish to enter into the tracking system. A decision must be made quickly. There is nothing more frustrating than making an offer and then learning some critical piece of information that would have resulted in a different decision.

Comfort

Be considerate of the candidate's comfort needs. The hand-off is a convenient time to ask if he needs a restroom break, beverage, or something else. Candidates sometimes want to make phone calls. This is usually fine, but if they repeatedly make calls or check text messages, that is a concern.

The third-base coach

In baseball, the third-base coach communicates messages to the batter and the base runners through a series of prearranged signs. He might want to tell the batter to bunt, or a base runner to steal. The signs are complicated and involve hand movements, touching parts of the head, face, or body. Most of the movements are meaningless, but one is an indicator that what follows should be interpreted as the message. The goal is to communicate tactical information to a team member without the opposing team knowing it.

This is similar to the problem of communicating to the next set of interviewers when handing off the candidate. Just today, we had a candidate who appeared to be brilliant. He aced the phone interview, but he already had an offer in hand. We agreed in advance that if the candidate did very well during the technical interview, that we would communicate ahead that this was someone we wanted to go into sell mode with.

We decided that we would bring in our chief architect if we determined to go into sell mode. He would be on the third interview team and we were able to communicate that to him, and to the coordinator, verbally. We also wanted to communicate to the second interview team, so we prearranged a sign to indicate whether to sell the position or to conduct a normal interview. The sign was to give a handshake to one of the interviewers if I wanted him to go into sell mode.

As we hoped, the candidate was great, and I signaled to the next interviewer. They performed a sell-oriented interview and we made an offer feeling confident that the candidate would accept.

Tours

Consider giving candidates a quick tour of your facility. Do this especially if there is something interesting to see. At least walk through the developer areas and point out what the work environment is like.

If you don't have time for a tour, use the transitions between rooms, to and from lunch, or to and from the restroom, to do a mini-tour. Senior candidates want to know what the workspace is like, so if you don't show them, they may go home with doubts.

If you feel embarrassed to give candidates tours, put down this book and figure out why. Some salespeople can sell anything, but the best salespeople are selling products they believe in.

5.19 Conclusion

The hardest parts of hiring are now over. The candidate's résumé earned her a phone interview. She was impressive enough to be invited for an on-site interview and she did well. All that remains is to determine the offer and close the deal.

Chapter 6

Closing the Deal

The interview process is almost over. The interview team has decided to make an offer and now the goal is to get the candidate to accept it. This chapter will take you through these final steps of the hiring process, including how to deal with references, making the offer, and negotiating the offer.

6.1 Checking references

Before tendering the offer, should you check the candidate's references? Reference checking is an area where I tend to go against conventional wisdom that says you should check all provided references. I usually do not check references if my employer does not require it, because I have found that they are rarely helpful.

Although I don't usually perform reference checks, there are some reasons cited to do them:

- The importance of the position demands a reference check.
- You may get more useful information about the candidate.
- The reference check may resolve indecision.
- You can check the truthfulness of the candidate.
- You learn about the marketplace.
- Reference checking provides a recruiting opportunity.

Certain positions are so important that taking the time to do reference checks is a no-brainer. When you are filling a small number of positions and you require the very best person, not just a really good one, you should check references. Leadership positions always deserve a reference check. Very senior and influential technical positions, such as architect, should be checked. Beyond that, reference checks may not be worth pursuing.

By checking references, you may discover information about the candidate that will sway your decision. There is a chance you will learn that the candidate is not as strong in a skill as you were led to believe, or that there is a significant behavioral "quirk" that could be a problem in your environment. It is rare you will find a "smoking gun" though. Conversely, you may learn of a valuable ability that was not revealed in your interview.

A reference check might resolve indecision about the candidate. Typically if there is indecision about a candidate, especially if it is more evenly split, I recommend not making an offer. In cases where there is general agreement to make an offer but there are a small number who disagree, checking references is a way to validate their concern and resolve the issues they raise.[1]

Sometimes a reference check will reveal the candidate was not truthful in portraying their job history. Dishonesty is always a serious problem, but care must be taken to understand whether the candidate was truly dishonest, whether he was stretching the truth, or whether the interview process failed to reveal the truth. In the case of clear dishonesty, it is best to disqualify the candidate. In the other two cases, some form of follow-up with the candidate may be in order depending on the nature of the concern.

I find one of the more useful aspects of reference checks is the opportunity it provides to learn about the marketplace. I may learn about the company of the referrer, their market, their company's health, how they work, and their philosophies on hiring. I may learn about a new competitor. I may make a professional connection that will enhance my personal network, or I may identify a collaborative resource that could benefit my company. One never knows!

Finally, checking references is a recruiting opportunity. It is often the case that the candidate is not the only person leaving the referrer's company. Often things are getting bad and others may need a new job. It is common for

[1] Another way to resolve isolated concerns is with a follow-up interview, either in-person or over the phone.

a referrer to recommend others who are looking for work because of a layoff or other concern about the company. In some cases, the referrer is interested in an opportunity!

Despite these benefits, there are also problems with reference checks:

- Reference checking takes time.
- The referrer is under no obligation to be truthful.
- The referrer was hand picked by the candidate.

Reference checking takes valuable time. Your odds of losing a candidate are greatest between the on-site interview and the making of the offer. Time is of the essence. Unfortunately references are not always easy to contact. I find it typically takes two to four days to chase down four referrals. That may not seem like much time, but it can mean lost candidates.

The individual referenced does not want anything from you and so is under no obligation to be truthful about the candidate. You know very little about the referrer and it is unlikely you will be able to determine whether they are speaking truthfully. I guess that 25 to 50 of the references I have checked were not truthful based on the subsequent performance of the candidates after they were hired.

The story about Frank, Ralph, and Rebecca on pages 195 and 196 drives home a few points. The first is that you never know—and probably cannot discover—what the true situation and motivation of the reference is. The second point is that, just like interviews, you must drill down to real answers. The third is to resist letting your preconceptions and desires get the best of you. Rebecca asked somewhat careless questions and did not attack the ambiguous answers. She didn't feel she needed to because Ralph was telling her what she wanted to hear.

Finally, the candidate handpicks the references and so they are almost guaranteed to give positive reviews. To me this is the biggest problem with reference checking. I consider myself fairly skilled at teasing out the truth in a reference check, but even so, the failure rate is too high to rely on the reference information.

Only a fool would list a reference that would not give a good referral. Fools are easy to spot so I don't hold much hope of getting useful information about a candidate from a reference check.

A story

Frank has worked on Ralph's development team for nearly two years now and the situation hasn't improved. Frank is a smart developer with a strong personality. He has an ego and he is somewhat insecure. This has led to destructive dynamics on Ralph's team. Most design discussions end up with Melissa and Roy—both talented developers—giving in to Frank's opinion because he is so strong-willed and belligerent. Melissa and Roy have been coached on conflict resolution but Ralph knows that sometimes they just give up rather than fight.

Frank can lord it over the testers. Rajesh finally admitted to Ralph that the bug that brought down the site was one he had discovered in Frank's code but failed to report because he didn't want to deal with Frank.

Ralph has spoken to Frank several times in the last six months without noticeable improvement. Ralph—who is averse to confrontation—has hoped the problem would go away, but it hasn't. Ralph has had enough so he speaks with Carol, the VP of Development, about the situation.

Ralph explains, "Carol, I think we need to let Frank go. He's destroying our team and he's undermining the quality of our product." Carol asks Ralph about the warnings. "Ralph, did you specify any consequences if Frank failed to change?" Ralph responded that he hadn't. "How about an improvement plan?" Ralph shook his head. Carol said, "Ralph, I wish you'd come to me earlier. As it is, it may take a while before we can take any action with Frank. I need to talk to our lawyer."

Ralph didn't realize that you couldn't just fire someone for being difficult to work with. This was the message Carol relayed from the lawyer. If the company fired Ralph, they might be exposed to legal action.

As it turns out, Frank wasn't happy either. He was tired of everyone arguing about his brilliant design ideas. He wanted to work for a

Employment and salary verification

I think it is a good idea to verify the candidate's recent employment. If you have an HR organization, this is a perfect task for them. These verifications require calling the other company's personnel organization and do not recompany that hires smarter developers like him. As luck would have it, Frank approached Ralph asking if he would be a reference. Ralph almost said no but had a second thought: "If I give a good referral for Frank, I can get rid of him right away rather than go though a six month process to get him fired." Ralph said he would be a reference for Frank.

The following week Ralph received a call from Rebecca, who had just interviewed Frank for a senior developer position. After confirming Frank's employment and title, she asked about Ralph's relationship to Frank. Ralph said he was Frank's development team lead and that he worked closely with Frank every day.

"What are Frank's strengths?" Rebecca asked. "Frank is very intelligent. He's strong in design," replied Ralph. Rebecca then asked, "How is he to work with?" Ralph thought for a moment, not wanting to lie. "Frank is a very high-energy fellow. Sometimes we have trouble keeping up." Not a lie, thought Ralph. Rebecca spoke her thought out loud, "High-energy is not a bad thing. We detected that in our interview." Rebecca only spent forty five minutes with Frank, but like everyone else, she realized he was very intelligent and seemed like a go-getter. He had a strong personality, but it didn't seem to be a problem.

Rebecca half-asked, half-stated, "Frank seems to have a strong personality." Ralph responded, "That he does. He really attacks a problem!" No need to stretch the truth here, he thought.

By now, Rebecca was satisfied that Ralph had confirmed the team's observations. She was finally going to fill that position after two months of trying. Rebecca thanked Ralph for his time and hung up. Ralph, for his part, thought the call was a huge success. The caller was clearly impressed with Frank and hopefully she would make an offer soon. As a bonus, Ralph—who was an honest, if somewhat wimpy person—didn't have to lie.

quire technical expertise. Legally, the company is only required to answer a few questions: they must confirm employment and employment dates.

Salary verification may be worth pursuing for the most recent position. The primary value is to ensure the candidate isn't falsifying information to get a better offer. That said, most candidates realize the prospective em-

A failed reference check

There was a candidate applying for a critical senior technical position where reference checks were required by the company. The candidate interviewed well, was technically qualified, and the behavioral skills seemed to be there.

The reference checks were unanimous in praising the candidate's technical skills. We specifically asked if there were any behavioral issues and whether the candidate worked well with others. The universal response from the three references was that there were no problems.

We hired the candidate and it became clear the technical praises were justified. After a few months though, a problem emerged. The employee had anger management issues and got very defensive when others criticized his work. This was the case even in reviews where constructive criticism is necessary. The issue was so bad, the employee had to be let go.

Two issues are clear in this case. The first is even the best interviews may fail to reveal certain problems. It is unlikely the anger issue would be revealed during the interview. The second issue is that the hand-picked references were either dishonest about the candidate, or they didn't know the candidate well.

ployer will check and so are truthful, because it is easy to verify salary, and dishonesty will lose them the job.

If you do not have other resources to pursue salary verification consider that the candidate is unlikely to lie, that getting the information will slow you down, and that the information is not that useful beyond the truth test. You should have already determined a salary range for the position and your interview process ought to determine where the candidate falls in the range. Your offer should be dictated by value, not by the employee's history.

6.2 Making the offer

At this point, your hiring team has given the thumbs up to the candidate, and now you must make an offer. Your goal is to make your offer as appealing and fair as possible. Present the offer as a package with everything included:

- Title
- Salary
- Signing bonus
- Bonus program
- Incentive plans
- Paid time off
- Holidays
- Health insurance benefits
- Retirement plans
- Investment plans
- Relocation
- Work visa support
- Other perks
- Job assignment

Include everything so as to maximize the candidate's understanding of the entire package. The sophisticated candidate will look at everything when comparing offers.

The decisions you must make are with respect to title, salary, signing bonus, relocation, visa support, and the job assignment. The last three depend on company policy and the situation. The first three are the ones you have the most control over.

Many companies have a defined set of titles to choose from. In that case, your job is to determine which one fits the candidate. It may not necessarily be the advertised title. For example, you may have been advertising for a *Senior Software Engineer* but the candidate falls a little short of your standard. You still want to make an offer so you choose *Software Engineer*, which is the next title down in your system.

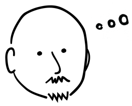
You may be unconstrained by company policy. In that case, your options are limitless. It is worth noting that employees have a wide range of feelings toward titles. Some care deeply and wish to have the best title for their résumé. Others would prefer every developer just have the same title. If you have discretion, be sensitive to this.

Salary is your blunt instrument for hiring. Despite what many say, size does matter. On the other hand, mature software professionals do put less weight into the salary and consider other aspects of your offer. It is a warning sign if your candidate is overly fixated on salary; it is a sign of immaturity and you should reflect on whether the candidate is for you.

The best way to choose the salary is to base it on market knowledge, company knowledge, and your estimation of where the candidate fits within your company. Monitor what comparable companies in your area are paying. You can learn this through various surveys, recruiter information, social networks, and by seeing enough local candidates. Your salaries, in the context of your complete offer package, must be competitive.

You need to know what your company considers appropriate salary ranges for each title. You should understand the salary distribution for current employees. Your offer should not violate company salary policy, nor should it disrupt the parity for existing employees. This is a delicate issue that I touched on earlier.

Finally, given your knowledge of the market and your company, you must estimate what this candidate is worth. If you have a twenty thousand dollar range for a senior software engineer, where does this candidate fit in that range?



Base your salary on market forces and what the candidate is worth to you. Don't base it on the candidate's expectations or his previous salary.

With salaries, the ground is always shifting. The supply, the demand, the employee salaries, *etc.*, are all in flux. You must stay on top of these changes to ensure you are making competitive offers.

A last word about salaries: never make low-ball offers. You are not buying a car, you are trying to hire and retain a potentially great employee. If an employee discovers you ripped her off, she will be full of resentment. She may eventually leave your employ, but in the meantime, she may be poison for your organization. Make offers based on value, not on bargains. If money is that big an issue, maybe you shouldn't be hiring.

The signing bonus is an optional tool you have to sweeten the deal. Often employees between jobs can use the extra cash right away. An extra few thousand dollars can make a big difference. You might choose to offer stock options or grants instead of cash. Grants have real value, but are not as enticing as cash because of delayed gratification. Candidates typically view options cynically. I would avoid them unless you are a start up and options are the only currency available.

The candidate will also consider things you didn't put in the offer:

- your work environment
- your culture and principles
- the nature of the work
- degree of autonomy
- the quality of your employees
- the stability and future prospects of your company
- opportunities for growth
- management style

These are the things you hope your employer has paid attention to; not just how things really are, but also how they are perceived. More than any of the things in the offer, these are the kinds of things that will close the deal. If your employer is lacking in these areas, you are hiring at a disadvantage.

Consider a two-step offer process. The first step is to make a verbal offer over the phone, and the second is to deliver a formal written offer. This allows you to get your offer on the table as soon as possible without any fear of unanticipated problems related to the logistics of delivering the formal written offer.

Timing

A strong candidate is likely to have other offers, so you need to move as quickly as possible. Contact candidates within two business days. One company I worked for tried to contact the candidate within twenty-four hours. We sometimes called on the evening of the interview.

This quick response really impresses the candidate and increases your odds of success. Knowing that you are wanted is a great motivator to sign. Make it a practice to respond—whether making an offer or not—as soon as possible. Nothing is a bigger turn-off for a candidate than incompetence in delivering the offer.

Who makes the call?

Who should call the candidate with the verbal offer? I've tried several approaches and they all have their advantages and disadvantages. What I've settled on as being the most practical is to have the person who did the close make the call.

Remember the closer is usually a manager at the director or vice president level. She is also someone the candidate met. The combination of influence and familiarity usually leave a positive impression. Furthermore, the fact that someone involved in the interview completes the process on behalf of the candidate seems to reflect a sense of accountability that also reflects well on your organization.

On the other hand, the impression you would leave by having an HR representative make the call might not be positive. It depersonalizes the whole process and could make the candidate feel unimportant. Additionally, an HR representative is not going to be able to answer most of the important questions a candidate might wish to ask at the time of the verbal offer. A manager involved in the interview has the context, the knowledge of the organization, and the authority to answer just about anything the candidate might dream up. Most candidates want answers before committing to a new job.

There are other people who can make an effective call and, in some cases, these options are preferable. The obvious choice is the person who the new hire would report to. This is especially effective if that person was part of the interview process. If not, it may be better to have a more senior person make the call.

If the candidate seems like she would be impressed by a call from a particular person, this can be a good option too. For example, a candidate who seems really impressed by the technical aspects of your company might respond well to a call from one of the technical interviewers. Anyone in the hiring process who developed a strong rapport, and who is qualified to make the call, could be considered. Finally, a person who championed the individual, maybe a friend or person who recommended the candidate could also make the call.

Goals of the call

The primary goal of the call is to try to get a verbal commitment from the employee. Short of that, you should try to find out where your offer stands with the candidate. Does the candidate perceive any shortcomings with the offer? Is the offer competitive? Is the candidate's perception of the market reasonable? Are there any obstacles? What questions still need answering? Is there anything you can say to encourage a decision? Or worst case; is there no point in a formal offer?

The call should improve the chances that the candidate will accept your offer. If your team did a great interview, your offer is fair, and your company is a great place to work, you stand a very good chance of getting a verbal commitment. What are the possible outcomes of the call?

1. The candidate verbally accepts your offer.
2. There are obstacles to be resolved before a decision can be made.
3. The offer doesn't meet the candidate's expectations.
4. The candidate decides not to pursue employment at your company.

In the first two cases, you will want to initiate the formal offer. In the second case, there may be things you can do to help remove the obstacles. This may require further action and approvals. In the third case, find out where the offer falls short of the candidate's expectations. If it is a perception problem, try to correct it. In the fourth case, get some feedback. Was it something your company did that could be improved? This information should definitely be discussed by the organization.

A typical example of a perception problem would be a situation where a candidate is moving from an area with high salaries, such as Silicon Valley or Boston, MA, to an area with significantly lower salaries. There are wage calculators available on the Internet; use them to explain to the candidate that salaries in your area are lower because the cost of living is lower. Suggest the candidate do the research himself.

The formal offer

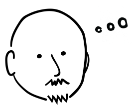
Right after the call is complete, the process of sending the formal offer should begin. The tracking system is a great way to coordinate this workflow. Creating and sending the offer letter is another good task for HR.

It is best to email the offer letter because it gets to the candidate immediately and can be acknowledged. Consider standard mail as a follow-up to ensure the candidate receives the offer. Typically you also send an application and detailed benefits information along with the offer letter. The candidate should print and sign the offer letter, print and fill out the application, and mail them both to your HR department. The application, although a nuisance, is often required for background checks. If you explain this, it might diffuse the frustration involved in filling out redundant forms. Only a small portion of the typical application form is required for a background check, so let the candidate know what parts you require.

6.3 Negotiating the offer

What if the candidate wants to negotiate a better offer? If you've done your homework, there shouldn't be much to negotiate. Most candidates want to negotiate title or salary. If you know what you are doing these are the two things you should avoid negotiating. The corporate aspects of the offer—health plans, bonus and incentive plans, investment and retirement plans and holidays—are typically not negotiable.

Surprisingly, the size of the negotiated change often isn't as important as the act of negotiating. If the candidate feels like he won a concession, he may be more likely to accept. Some cultures are steeped in negotiation and bartering and there may be a pride aspect involved. You don't need to honor this but it is something to consider. There are aspects of the offer that are less painful to concede on.



Avoid negotiating salary or title. Make a fair offer and stick to it. One-time items, such as a signing bonus or relocation, are good tools for sweetening the deal.

With regard to title, you should know where the candidate fits in your system. If the candidate is pushing for a better title, here are a couple of ways to take it off the bargaining table. If your organization does regular reviews, or better yet, ninety-day new hire reviews, you can explain that the job title can be reviewed at that time. If the candidate's ability dictates, an adjustment can be made. If your company is growing, then there are likely regular opportunities for advancement. Explain this and provide real numbers for how long it usually takes to get a promotion.

The salary offer should also have been a thoughtful decision. There may be a little room for negotiation here. For example, your salary range for a Senior Software Engineer might be from $85,000 to $105,000. If you offered at the bottom of the range, you might consider a $2,500 or $5,000 increase.

Salary negotiations could have significant long-term impacts. If you overpay someone, it strains parity among other employees.[2] Employees may become resentful if the "new guy" is clearly overpaid. This could lead to attrition, which is hugely expensive. If you are proactive about correcting parity, you still can pay big-time. That $15,000 negotiation may require you to adjust many other salaries. All those increases are paid every year from then on.

Your company may allow some discretion for senior employees regarding paid time off. Adding five days off is similar to a few thousand dollar raise, not a big deal for you. This could be a low-cost bargaining tool if you are at risk of losing a great candidate.

Relocation is a good place to negotiate. It is a one-time, fixed cost. Often the greatest barrier for a potential hire is the fear of relocation. It is perfectly normal for a candidate to fear having to cover unanticipated relocation costs. You can mitigate this fear by throwing in a few thousand more dollars.

The signing bonus is another useful negotiation tool. Often candidates will say they need a certain salary that is beyond what you would offer. You can make a candidate whole by giving a signing bonus that makes up the difference. This works especially well if you have an incentive plan that will make up the difference the following year. You can explain that if the candidate is really good, she can expect a fair raise when salaries are adjusted.

There are other creative ways you can negotiate. As a general rule, prefer one-time costs over recurring costs (such as salary increases). Avoid negotiating yourself into an organizational parity disaster.

When negotiating, always know your limits and don't exceed them. I suggest you avoid entertaining more than one round of negotiation. When you are done, make it clear by saying, "That is our offer, we believe it is fair. It is our final offer." This is a powerful negotiating tool. If your offer is competitive, you will usually get a positive response. Negotiation has a way of getting out of control; concessions are made that you will regret, simply because you didn't know your limits and did not want to lose the candidate.

Sometimes there is nothing you can do to get the candidate to accept your offer. Stick to your limits and let the candidate go. In my experience, there

---

[2] Assume everyone knows everyone else's salary... you'd be surprised!

have been several candidates who declined our offer and later came back and accepted it once they realized their expectations were unrealistic.

The bottom line, in my opinion, is that there should be little negotiation. It says you don't know what you are doing, that you didn't make a fair offer, and that you were trying to get away with a bargain. Make a fair offer from a position of knowledge; consider negotiation for special circumstances; but in general, stand behind your offer. Experience has shown that this approach works best.

Chapter 7

Parting Thoughts

This book has focused on the guts of the hiring process: résumé review, phone interviews, and on-site interviews. Throughout, I have emphasized several important themes.

The first theme is that excellent hiring is one of the most important things you and your company do. Through great hiring, you can raise your organizational IQ. With poor hiring, you can dumb down your company and make it weaker. Hire for the long term, hire for the job position, hire for behaviors, and hire for leadership. Do your best, whether your team is hiring or not, because you never know when you will be working with that new employee.

The second theme is that hiring well is hard. I've shown that the basics are hard enough, and that to really become skilled at hiring takes hard work. You must thoughtfully review many résumés before you learn to spot patterns and learn to read beyond what was written. Not until you've led dozens of phone interviews will you begin to understand and successfully accommodate the challenges intrinsic to that form of communication. It may take a few years of on-site interviews before you see how your company's reputation is riding on how well you execute every step of the process.

The third theme is that great hiring depends on the skill of the people involved, and not on their ability to follow a process or a set of rules. Hiring is an empirical process where the same inputs do not give repeatable results. The information we base our decisions upon is incomplete and inaccurate. People are funky and constantly surprise us. In the face of all this, we must bring to bear skill, experience, and adaptability if we hope to do well.

Your hiring staff will benefit from sound principles that should rarely change. No set of rules can anticipate even a modest number of situations you will encounter in hiring, but principles can provide fundamental guidance when decisions must be made.

Guidelines, sample questions, and lines of questioning provide guidance for the Shu-level participant in the hiring process. As you grow to Ha and Ri level, you realize these tools are insufficient, and you must take responsibility to find ways to extend these tools.

The fourth theme is the importance of regular feedback. Although the information you have in the hiring process is inaccurate and incomplete, it is not random or chaotic. There are clues, patterns, and combinations of information that you can learn to spot. The learning process requires that you form hypotheses and then test them, as more information is available. You can get this kind of feedback by participating in multiple phases of the hiring process, by reviewing other's comments in the tracking system, and by getting to know new hires that you helped interview.

Another kind of feedback involves, training, coaching, mentoring, and discussing. This is where you learn from more experienced interviewers. Take every opportunity to discuss candidates with other participants. Ask senior interviewers what they are seeing. Pair on résumé reviews and participate in group reflections after phone and on-site interviews.

The final theme is to be flexible and adapt as necessary. The hiring landscape is always changing: supply, demand, hiring staff, and the quality of the candidates are in constant flux. You and your hiring team must make adjustments to accommodate these changes if you wish to maintain a high degree of success.

Every candidate is different and brings new challenges to your hiring team. The team that understands principle-based hiring, the path of Shu, Ha, Ri, and the primacy of continuous reflection and improvement, is the team that will hire the best people. Good luck and hire well!

Appendix A

Valuable Behaviors

Here's a list of valuable behaviors to look for in an interview candidate:

Adaptable Demonstrates the ability to adjust readily to different situations.

Accountable Liable to account for one's actions; obliged to take responsibility.

Oral communication Demonstrates concise, complete, and clear verbal communication.

Written communication Conveys ideas clearly through written forms such as documentation, memos, and emails.

Analytical Ability to define the scope of problems, break them down into elemental parts, and reason about those parts and the whole.

Collaborative Works effectively with others to efficiently and correctly solve problems and complete tasks.

Detail-oriented Appropriately concerned with important details. Conversely, does the candidate veer onto unimportant minutia?

Decisive Able to make important decisions in a timely fashion that usually result in successful outcomes. Conversely, does the employee avoid making decisions?

Delegates Demonstrates the ability to assign tasks to others in such a way as to most often result in successful outcomes.

Energetic Powerful in action or effect.

Entrepreneurial Willing to take measured risks in order to succeed.

Independent Takes ownership of problems or tasks when appropriate.

Initiative taker Demonstrates the ability to begin or to follow through energetically with a plan or task.

Honest Honorable in principles, intentions, and actions; upright and fair; truthful.

Innovative Forward looking; being or producing something like nothing done or experienced or created before.

Leader/Influencer Able to marshal resources toward a common cause.

Listener Seeks to establish understanding when interacting with others.

Optimistic Tends to expect the most favorable outcome. Has a "can do" attitude that is based on rationale and information, rather than fantasy and guesses.

Mentor Serves as a counselor or teacher in an occupational setting.

Motivator Able to incite or impel others to positive action.

Consensus builder Finds common ground among parties with different interests, negotiates agreements, and establishes responsibilities.

Tactful Possessing or exhibiting tact; considerate and discreet, empathetic.

Planner Able to formulate a scheme or program for the accomplishment, enactment, or attainment of a goal.

Relational Develops and nurtures positive and healthy working relationships.

Team player Understands and accepts a team role; works toward the success of the group.

Pride Demonstrates unselfish pleasure and satisfaction with one's accomplishments.

Quality Produces or provides products or services of high quality or merit.

Appendix B

Hiring Principles

1. Always be respectful of the candidate.
2. Always be on time.
3. Be prepared.
4. Value the candidate's behavioral qualifications nearly as much as their technical qualifications.
5. Always consider and value leadership potential.
6. Persistently and politely pursue answers from the candidate.
7. Every interview is different; be creative.
8. Take ownership of your needs, opinions, and decisions. Don't be swayed by the opinions of others without good cause.
9. Respect the opinions of others.
10. Leave the candidate with a positive impression of your company.
11. Err on the side of patience over expedience. In other words, wait for good candidates.
12. Hire for the long term.

Appendix C

Résumé Review Checklist

Before starting a résumé review, be sure you are prepared. Know what positions you are reviewing résumés for and what constraints you must meet.

First, scan to determine:

- Are all the expected parts of the résumé present?
- What are the associated costs for the candidate: travel, relocation, immigration, *etc.*?
- Is the résumé neat or sloppy? Are there misspellings or poor grammar?
- Is the objective or summary consistent with the position(s) applied for?
- Do the job descriptions support the objective or summary?
- Does the candidate's self-perception seem accurate?
- Does the skills section mention the technologies you are interested in?
- Is the skills section supported by the rest of the résumé?
- Does the résumé contain an education section, and is it complete?
- Are the important parts of the employment history present?
- Does the employment history have gaps or overlaps? Can they be explained?
- Is job hopping a problem, especially in recent years?

Then, examine for detail:

Position match

- Does the candidate appear to have—-supported by details—the *required* skills, behaviors, and experience for the job?
- Does the candidate have—-supported by details—any of the *nice-to-have* skills, behaviors, and experience for the job?
- Are the candidate's skills and experience sufficiently current?
- If the candidate is not qualified for this job, are there other openings she might be qualified for?

Culture/Environment match

- Can you tell whether the candidate is likely to adapt to your work culture?
- Is the candidate likely to be comfortable in your work environment?
- Might the candidate be trying to *escape* an unhealthy environment?
- How will the candidate's experience with development methodology and process adapt into your company's methods and processes?
- Are there any contra-indicators, *e.g.*, non-profit vs. for profit, open-source vs. commercial, waterfall vs. agile, *etc.*
- Does the candidate's willingness to adapt match your expectations?

Experience and performance

- Does the candidate have sufficient years of experience with the technologies, *etc.*, that you value?
- Does the candidate have a healthy career trajectory?
- Or, has the candidate stagnated in position or skills?
- In cases where performance is unimpressive, are there indications that an employer may have misappropriated the candidate?

- Are there backward steps in the candidate's career? If so, are they a problem, or something to be explored?

Growth

- Are there signs the candidate can learn and grow on her own?
- Has the candidate made successful jumps in technology areas, business domains, programming languages, methodologies, *etc.*?
- Are their indications of mastery in any skills or technologies?
- Does the résumé reveal what the candidate is most interested in?
- Can you tell how the candidate learns?

Leadership

- Is there evidence of people leadership?
- Is there evidence of technical leadership?
- Are there any publications, presentations, committees, or other technical leadership items cited?
- Does the candidate appear to be averse to leadership? (Remember *one step forward, two steps back* is not necessarily bad).

Impact and Potential

- Has the candidate made significant contributions or positively impacted his employer's bottom line?
- Are the claims in the job history clearly things the candidate *did*, rather than things done around them?
- Are the accomplishments meaningful or minor? Merely listing trivial accomplishments in more senior positions is a warning sign.
- Does the candidate appear to have strong potential?

Education

- How relevant is the degree to your opening?

- What is the quality of the institution(s)?
- How heavily should education weigh against the candidate's experience?
- Does the candidate have advanced degrees with projects or theses you can review?

Information hiding

- Are there any signs of dishonesty?
- Are there patterns of information hiding, vagueness, or incompleteness?

Bibliography

[Cen] Cenedella, Marc. "Seven Deadly Sins of Interviewing." *TheLadders.com*. Available on the web at: http://sevendeadlysins.theladders.com (accessed June 2010).

[Coc06] Cockburn, Alistair. *Agile Software Development, The cooperative Game*. Addison Wesley Professional, second edition, 2006.

[Fel06] Felps, W., T. R. Mitchell, and E. Byington. "How, when, and why bad apples spoil the barrel: Negative group members and dysfunctional groups." In *Organizational Behavior*, volume 27, pages 181–230. 2006.

[Gla09] Gladwell, Malcom. *Blink: The Power of Thinking Without Thinking*. Little, Brown and Company, 2009.

[Kru99] Kruger, Justin and David Dunning. "Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments." In *Journal of Personality and Social Psychology*, volume 77. 1999.

[Lew04] Lewis, Michael. *Moneyball: The Art of Winning an Unfair Game*. W. W. Norton & Co., 2004.

[Rod05] Rodman, Jeff. "The Effect of Bandwidth on Speech Intelligibility." 2005. Available on the web at: http://www.polycom.com/global/documents/whitepapers/effect _of_bandwidth_on_speech_intelligibility_2.pdf (accessed January, 2010).

[Rot04] Rothman, Johanna. *Hiring the Best Knowledge Workers, Techies & Nerds*. Dorset House Publishing, 2004.

[Shu] "Shuhari." *Wikipedia*. Available on the web at: http://en.wikipedia.org/wiki/Shuhari (accessed April 22, 2010).

[Sig] Signatories. "Manifesto for Agile Software Development." Available on the web at: http://agilemanifesto.org/ (accessed June 2010).

[Spo08] Spolsky, Joel. "What's your average length of time at any one place?" 2008. Available on the web at: http://discuss.joelonsoftware.com/default.asp?joel.3.645141.50 (accessed April, 2010).

[Yeg] Yegge, Steve. "The Five Essential Phone-Screen Questions." Available on the web at: http://sites.google.com/site/steveyegge2/five-essential-phone-screen-questions (accesssed April, 2010).

About the Author

Sean Landis

Sean Landis, a software architect with over twenty years of experience hiring software professionals, has hired in companies with less than ten developers and ones with thousands. He successfully retooled hiring practices at three companies, leading to significant improvements in the quality and quantity of new hires. Sean is a practicing software professional who today practices agile development and has innovatively applied agile principles to hiring. Sean earned a Bachelor of Science in Computer Science from the University of Utah, and a Masters of Engineering, Computer Science, from Cornell University.

Index

Page numbers followed by an n *refer to footnotes.*

P

V

W