

Project Survival Guide

Traits you need from the get-go

- 1) The ability to Communicate – you must:
 - a) Be able to read a situation and adapt
Be aware of your surroundings and the situations and environment you are in
Can assess how you fit in the environment
Be an active listener and inquire if you don't understand what you hear
If you're fresh into the working world this might be your hardest hurdle
 - b) Tailor your message to your audience
 - c) Able to clearly break down the topic at hand into building blocks that can be assembled into a presentation or argument
 - d) Explain why your idea is important
 - e) Able to enact a call to action - Your proposal as to what comes next
 - f) Ask for feedback – It helps your audience engage with your idea
Remember, especially early in your career, or even just a new job, you are looking to collaborate with your colleagues in most of what you do
- 2) The ability to ask important and probing questions in a non-threatening way
- 3) Network, network, network
Find one or more mentors. A few will shun you. Most folks are happy to pass on what they know and were taught by those who came before them.
- 4) There's an old saying: "Those who know what they are doing make things look easy, those who don't make things look tough." Be-friend the folks in the first category.
- 5) Be a team player. That means not only doing your part, but also look out for them also
- 6) Take responsibility for that entrusted to you
- 7) Always make sure that you keep your word.
This means always stay cognizant of what you commit to. If for some reason, and only when the situation is out of your hands, you fail to deliver what you are promised, be honest as to what went wrong, and apologize. Apologies often take any ill wind out of the sails of those who would come down on you.
- 8) Be a decent human being – keep caring and empathy front and center
Strive to keep a positive attitude, to be friendly, and maintain rapport with your colleagues.
- 9) Keep a sense of humor. Except where taste would dictate otherwise, look for the humor in bad situations.
- 10) Learn how to give presentation in front of others
 - a) Delivering an attention-grabbing opening for a talk
 - b) Providing a summary of what will be covered as part of the introduction to the presentation and provide context.
 - c) Use body language and eye contact to convey energy and confidence.
 - d) Pause to emphasize key points, Modulating vocal tone for emphasis. Articulating clearly and smoothly.
 - e) Interjecting humor. Speaking with enthusiasm and animation. Projecting confidence.

- f) Summarizing key points at the conclusion.
- g) Field questions to clarify points

List you need as you move up

Even if you don't end up in management, you will move up the ranks in seniority, and prestige. Let's face the fact if you don't, you will be doing something else. So now your list will grow to include the items list below.

- 1) Leadership – There are times when you follow, but you must develop the ability to sense when others are looking to you to lead. Develop the ability to encourage and inspire trust.
- 2) Motivate – Not only yourself, but others. As you gain in stature others will look to you for guidance.
- 3) Conflict happens, learn to manage it.
Learn how to mediate them to minimize the negative aspects while enhancing the positive aspects of conflict. What's positive about conflict? It's been said conflict shows that people are engaged in the issue. The goal is not only to resolve the dispute, but also to learn from it.
- 4) Learn how to give constructive criticism, which is critique, both positive and negative in an amiable way. Provide positive reinforcement. Even if you must fire someone you can usually, not always, find something positive to send the person off on.
- 5) Mentor and Counseling those coming up behind you. You were most likely helped early in your career. Pass it on.
- 6) Learn the art of Negotiation and Persuasion. Collaboration.

Why people do what they do

Rule number 1: People often do not say what they mean, and they often do not mean what they say.

Francis Bacon an English philosopher and statesmen in 1620 classified the four leading intellectual fallacies of his time, which he called idols. The term idol represented an image or belief that mislead the mind's thought process. The first was Idols of the Tribe. These are misconceptions rooted in human nature. Idols of the Cave are deceptive beliefs rooted in the individual.

The third, Idols of the Marketplace is often in the forefront today, as this refers to words used incorrectly, semantically, to convey ideas. Using deceptive verbiage to convey thoughts. These are errors in thinking that come from a misunderstanding of words and language in communications between people. This idol refers directly back to rule number 1 above.

An example: "This project is so important that we must have it done in three months." What they really meant was, "This work is so unimportant that we don't want to fund it beyond that time."

The fourth, Idols of the Theater is another often invoked meme today. This is where false belief structures are built upon false knowledge foundations. You find debates regarding this Idol in the fields of science, philosophy, and religion. These are errors in thinking that come from a blind acceptance and unquestioned acceptance of attitudes and ideas. Bacon claimed that the dogma that gains momentum and is backed by persons or groups exalted by others had created lines of reason and vernacular worthy of the stage.

People believe what they want to believe because they are not generally good listeners. They hear selectively. They only hear what they want to hear. That is because they tend to project their own biases or experiences on situations as they often get personally involved in analysis of an issue and tend to let their feelings overcome a sense of objectivity. In addition, they tend to generalize about a specific event.

A form of this is when estimating the time needed or cost to complete a project. Managers tend to hear only the most optimistic date and forget the contingencies or feature creep that they will add as the project proceeds. Managers oversimplifying a project puts pressure on the engineers to accept a shorter schedule than is reasonable. This results in increased pressure to perform at great cost, resulting in longer hours and working weekends.

People are eager to rationalize. People are often not able to distinguish what is relevant from what is irrelevant. That is because people are easily diverted from the issue at hand as they are often unwilling to explore thoroughly the ramifications of a topic and thus tend to oversimplify.

It is easier to judge from appearances. They observe something, misinterpret what they observe, and make terrible errors in judgment. The bottom line is that they rarely think carefully before they speak, but they allow their feelings, prejudices, biases, likes, dislikes, hopes, and frustrations to supersede careful thinking.

There is also a basic code of human interface - invading someone's space. I have a right to criticize my family, myself etc. But you don't have the right to do the same.

Ways to put another on the defensive - take personal attacks on ideas you support, use dogmatic statements like "How can you say that" or "you don't know what you're talking about". Become sarcastic, or patronizing.

Cultures matter, different groups around the world look upon disagreement in different ways. People in some cultures look upon their work or ideas they present as personal. If you question either you will hurt their feelings and get an unhappy response. Many Asian cultures find open disagreement in poor taste. Around the globe you find cultures that will openly disagree, some that look for group consensus, or some that will signal disagreement by indirection, or a coy,

non sequitur responses to your assertions. You will also find cultures that will use ambiguity or will simply defer to superiors. The Japanese will often appear to agree with you by serially saying Hai, or yes. Often that simply means that they are acknowledging that they hear you, not necessarily that they agree with what you are saying. You need to do your homework on the particular culture you will be dealing with. You need to find a path to respectfully discuss ideas, designs, programs, and schedules in order to make the best product possible. Throwing out ideas and using the best ones makes a product better.

Looking at it as a manger

Ideally a manager's main function is not to make people work, but to make it possible for people to work. Management's job is to get the right people, make them happy, and turn them loose. They need to mine the untapped talents of their employees. Managers can either look at themselves as extractors (of work) or service providers. The best bosses take chances and put some of their reputation and fate into their subordinates' hands, thus demonstrating trust. Your boss will know you by your fruits.

But many managers tend to treat workers as faceless resources, and management insecurity can lead to arbitrary standardization. Add the fact that many managers are risk adverse, which often means not doing anything amazing. Anything new means some level of risk.

A long-time process of managing and reviewing employee performance is known as Management by Objectives (MBO). It's five step process is straight forward:

1. Review organizational goal
2. Set worker objective
3. Monitor progress
4. Evaluation
5. Give reward

The problem is that as with many things the human implementation of MBO goes astray. The organizational goal is usually set from high and might not be based on the reality that exists below. An obsession with measuring the quantitative results of the employee means that the employees might make sure all the required boxes are checked, the quality of the result might be subpar. In addition, the monitoring process from group to group might not be consistent skewing the overall organization's goal.

Management, and their reports often concentrate on technology, when they really need to manage the sociology of the group and larger organization. This is understandable because managing a group of creative designers and engineers, that is the development process, is inherently different than manufacturing or support. Manufacturing or production concentrates on the steady state. Not how you start something up or down or change. The rigid methodologies often required under those situations will stifle innovation and tend to punish folks who try something that fails.

Many managers are threatened by reports that assert their individuality. Hard to not look at folks in a project as to how much code they can write per day or how much documentation they can produce in a given time. You need to look at how each fit into the effort as a whole. Self-esteem is tied to the quality of what we do, not usually the quantity of what we do. The strive for excellence is not facilitated by speed.

As Tom Demarco and Timothy Lester described in their book *Peopleware*, management that is subscribing to the extracting work approach are using the “Spanish Theory” of project management. Essentially this theory claims that there is only so much wealth in the world, so control as much land, resources and people as you can to extract it. In the “English Theory, “wealth can be created, a prominent example was the industrial revolution. This theory does not adhere to the zero-sum concept of outcomes. Here it is believed that a rising tide can raise all boats. Wealth can be created where it wasn’t before, and by not extracting it from somewhere else or others.

The Spanish theory encourages unpaid overtime and says that people must always be under pressure. But overtime almost always leads to undertime. People can’t keep a constant 100% pace. It is argued that people really can’t be productive more than 40 hours a week on average. An historical example is described in Tracy Kidder’s Pulitzer prize-winning book “*The Soul of a New Machine*.” He describes how Data General (DG), a major player in the “mini” computer market of the 70s and 80s fell behind DEC, the largest player in that market. DEC launched their first 32-bit machine in 1978, and while DG was working on one of their own, it was stalled in development. A second “skunkworks” project was launched by DG. A Skunkworks project is where a small group tries to do something radical outside of normal organizational oversight. Some say the skunkworks is another word for insubordination.

What was radical here was a group of folks who were officially assigned to the sustaining engineering of a current 16-bit computer, the Eclipse, became the group that came from behind and got DG’s 32-bit entry, the MV/8000, out to market ahead of the main, and sanctioned group’s project. The Eagle project, as this clandestine project was known, did this by heavily invoking the Spanish theory of overtime. Productivity was phenomenal, but at the end of the project many of the team left. Ongoing productivity plummeted.

The authors of this were involved at a company where the project manager was beat up daily by middle and upper management throughout a “bet the company” project even though it was the PM’s brilliance that derived and formulated the product when marketing had no idea what to do for the next generation product. The team worked long days, longer weeks and even a 24-hour day toward the products required ship date. The company’s CFO came into the engineering lab at one point and told the group that if the product did not ship on time, he would be laying off half the company the next day. Just what the team needed, more pressure. Besides demonstrating poor management, it also highlights the fact that groups generally do not create great concepts or products, one person often is the geneses! The old saying a camel is a horse designed by committee is true.

Companies that find themselves reverting to the Spanish theory will often suffer personnel problems right up front as the amount of required overtime changes there are employees who have life situations that simply won't allow the increase. But people will often succumb to the demand for increased overtime not so much to get the work done on time, but to shield themselves from blame when the work inevitably doesn't get done on time.

Young companies often have managers that emerge and are not promoted. They identify the task that needs to be tackled and are fit to lead in that area. They figure out how to accomplish it by doing the homework and maximize the value to everyone involved. Some charisma and a sense of humor are a plus. But they must be able to innovate. But innovation is often unwelcome because it means change. Often it must be done in a "skunkworks" fashion.

New companies and unstable ones often promote their good employees to keep them. Makes for a very top-heavy organization over time.

The problem with investments in the company: They come out of that quarter's earnings, and they won't pay out until sometime in the future. Being and getting better is a long-term concept. Workers who feel that they have been invested in by the company tend to stay. Often because they are expected to stay.

From Peopleware: An expense is money that gets used up. At the end of the month, the money is gone and so is the heat (or whatever the expense was for). An investment, on the other hand, is the use of an asset to purchase another asset. The value has not been used up, but only converted from one form to another. When you treat an expenditure as an investment instead of as an expense, you are capitalizing the expenditure. Training in people is also looked upon as an expense from an accounting perspective, while it is actually an investment. Wall Street treats investment in people strictly as an expense.

An interesting fact is that most of an organization's learning is not done at the top or the bottom of the org chart, but in the middle. But it is this middle management, which is usually targeted for downsizing. So often the major investment in employee knowledge is jettisoned.

The Environment

Another investment in people is the work environment they are in. The office layout is always a tradeoff between cost, privacy, and quiet. People's workspace can't be too enclosed or exposed. The noises they hear in their workspace shouldn't be too different from the noises they themselves make. People work better in natural light. Danish law – every worker must have a window. Buildings can be built long and narrow to give maximum window space.

What is spent on an engineer is often 20 times what the cost of their workspace is. So, tampering with the space cost can have a gigantic risk. Does the cost savings more than justify the loss of effectiveness? Often the cost/benefit analysis does not look at the benefit or result delta as the cost side is changed.

The best companies have a philosophy of piecemeal growth, where while there is a set of patterns or shared design principles governing growth, local control of design is by those who will occupy the space. A company campus might follow these goals. Organic order without overbearing central order.

Project groups should be located together with a minimum of physical separation. If the group is split up in different locations there is very little casual interaction, and little team culture develops. When team members are neighbors, they tend to all go into creative mode at the same time. Management must realize that the office layout can't be to be able to observe their reports. You can't visually supervise engineers and developers. Trying to will often lead to the Hawthorne effect, where those being observed modify their behavior in response to being observed, and often not for the better.

Organizations usually have three types of spaces: areas with outsider access, insider access, personal access. It is the personal space that affects the employee's effectiveness the most. The

Studies have found that the best engineering workspace attributes are:

Needed amount of space

Quiet

Private

Able to silence phone

Able to divert calls

Low interruptions

A study done for the IBM research center in San Jose found that each worker needed 100 square feet, and 30 square feet of work surface, surrounded by six feet tall partitions.

This leads to the question: Does the above attract better people or does it allow people to be their best.

From Peopleware, developers spend their time:

Working alone – 30% They are noise sensitive

Working with one other person – 50% They are noise generators

Working with two or more – 20% They are noise generators

That book also found that noise is a major indicator of the quality of the work environment. Programmers who tend to do zero defect work, 66% of those worked in low noise environments. A tell-tale sign that the work environment is bad: people working in conference rooms, libraries, cafeterias, etc.

Another indicator of the quality of work is interruptions. It takes about 15 minutes after each interruption to get back into the creative flow. Work accomplished is assumed to be proportional to the time devoted to it. When there are lots of interruptions the ratio of

creativity to a designer simply being present can be as low as 10%. Some companies even have do not disturb icons that they display.

Early on in their career, engineers do mainly left-brain stuff, that is analytical and methodical thinking. Later, as managers and leaders they do more right brained stuff, Holistic and Heuristic thinking. But conversely, often creative engineering breakthroughs come from the right side. So, listening to music might make a person miss an epiphany.

Again, from the Peopleware book: One generation's technology (what is here now but wasn't as a kid) is the next generation's environment. The latest generation multitasks. That actually works against the creative flow. They say it is much better to spend 2% of your workday on Facebook, than 2% of all day on Facebook. Partial attention periods must be defined as personal time off.

A great boss will shield his employees from the politics of a large company. When times are good, they will push for your advancement and when times are bad to keep your job. You must reciprocate by getting the job done and looking out for them also. If you don't your boss will look bad and the boss also wants to keep his job. Large companies tend to have more departments and bureaucracies. The worst case we have seen was Northern Telecom in the late 1970's. There was a printed circuit board designer, working in Silicon Valley, who stated that every new part an engineer wanted to add to a board took two weeks to get approved and a design cycle was two years. This is a bit extreme, but in a small company the engineer would look at what was needed and add it, done.

One of the authors worked on a project for another large telco. It was to retrofit an existing media distribution system. Every change required as part of the retrofit, no matter how small, had a three-step process. First the telco would deliver a "needs" document.

Then the company doing the project, would propose a set of documents to implement the change. It wasn't design, or really any type of engineering, it was just documentation, mainly AutoCAD drawings, and the necessary BOM (Bill of Materials), and a workplan to actually implement the change, step-by-step. Finally, a budget in money and manpower to enact it was expected.

The change would be implemented if approved. The author got a 17-page document with a requested change. After boiling down all the jargon, or vernacular, what was being requested was that a cable connected to a piece of gear in one rack be disconnected and moved to another piece of gear in the adjacent rack. With all the restrictions, requirements, and other hurdles the contracting company had in place the budget calculated to make that change was close to ten thousand dollars based on the pricing algorithm the telco wanted followed. While the author was confounded by the amount needed to do five minutes' worth of work, the change order was improved.

Everyone involved with this project appeared extremely busy, and were, but the end result of real work was small. This demonstrates that you will see companies who want engineers, and claim it is engineering that these engineers are doing, it can often end up being clerical. The moral here is that many positions claim to be offering engineering opportunities. Make sure they are before you accept.

What about the mid-size company? A little of both but here again you don't have direct access or input to people who determine your fate in the company. The middle size company has more flexibility than the large company but still very HR structured. One of us went to hire an engineering technician at a medium sized company where he was the hardware engineering manager. To do that he sat down with the HR supervisor and went through a book of titles with job descriptions and salary ranges. Once that was agreed upon, he could proceed with advertising and interviewing. The company probably paid a lot for that book. The other thing was HR wanted to hire a person below the halfway point on the salary scale. Why, so that they could get raises over some number of years. If you hired them in at the top of the range, you would have to move them to the next higher title to justify a raise and keep them happy.

In a small startup company, you will likely meet the company mom. The company mom is about function and not gender. What does the company mom do? Everything needed to keep the place running! Payroll, accounts receivable, accounts payable, insurance, W2's, I9's buying furniture, receptionist and listening to upset employees. When VCs are dolling out money, keeping the head count to a minimum is a requirement. So, the company mom is a great way to help make the cash last. Pressure in a startup can be high like other jobs in industry; you will earn your paycheck or get laid off or fired.

The other type of small company is the mom-and-pop shop. Again, the mom function helps keep the costs down. The pop function is usually someone with technical skills who loves what they do and making big money is not their game. This type of business tends to make a product that is very specialized but without a large market. Usually, the product is sold to a particular industry. People in small companies tend to have a broader expertise. Instead of a hardware engineering department with managers, engineers, and technicians you design all the hardware by yourself. You might have access to consultants with specialized expertise in some cases. There are about three software engineers to every hardware engineer in embedded systems.