

## WHAT WOULD YOU SAY YOU DO HERE?

- Staff engineering roles are ambiguous by definition. It's up to you to discover and decide what your role is and what it means for you.
- You're probably not a manager, but you're in a leadership role.
- You're also in a role that requires technical judgment and solid technical experience.
- Be clear about your scope: your area of responsibility and influence.
- Your time is finite. Be deliberate about choosing a primary focus that's important and that isn't wasting your skills.
- Align with your management chain. Discuss what you think your job is, see what your manager thinks it is, understand what's valued and what's actually useful, and set expectations explicitly. Not all companies need all shapes of staff engineers.
- Your job will be a weird shape sometimes, and that's OK.

## THREE MAPS

- Practice the skills of intentionally looking for a bigger picture and seeing what's happening.
- Understand your work in context: know your customers, talk with peers outside your group, understand your success metrics, and be clear on what's actually important.
- Know how your organization works and how decisions get made within it.
- Build or discover paths to allow information you need to come to you.
- Be clear about what goals everyone is aiming for.
- Think about your own work and what your journey is.

## CREATING THE BIG PICTURE

- A technical vision describes a future state. A technical strategy describes a plan of action.
- A document like this is usually a group effort. Although the core group creating it will usually be small, you'll also want information, opinions, and general goodwill from a wider group.
- Have a plan up front for how to make the document become real. That usually means having an executive as a sponsor.
- Be deliberate about agreeing on a document type and a scope for the work.

- Writing the document will involve many iterations of talking to other people, refining your ideas, making decisions, writing, and realigning. It will take time.
- Your vision or strategy is only as good as the story you can tell about it.

## FINITE TIME

- By the time you reach the staff+ level, you will be largely (but probably not entirely) responsible for choosing your own work. This includes deciding on the extent and duration of your involvement with any given project. You can't do everything, so you'll need to choose your battles.
- You are responsible for choosing work that aligns with your life and career needs as well as advancing your company's goals.
- You are responsible for managing your energy.
- Some projects will make you happier than others, or improve your quality of life more.
- Your social capital and peer credibility are "bank accounts" that you can pay into and spend from. You can lend credibility and social capital to other people, but be conscious of whom you're lending to.
- Skills come from taking on projects and from deliberately learning. Make sure you're building the skills you want to have.
- Free up your resources by giving other people opportunities to grow, including starting projects and handing them off.
- Focus means sometimes saying no. Learn how.

## LEADING BIG PROJECTS

- Staff engineers can take on problems that seem intractable and make them tractable.
- It's normal to feel overwhelmed by a huge project. The project is difficult. That's why it needs someone like you on it.
- Set up the structures that will reduce ambiguity and make it easy to share context.
- Be clear on what success on the project will look like and how you'll measure it.
- Leading a project means deliberately driving it, not just letting things happen.
- Smooth your path by building relationships and deliberately setting out to build trust.
- Write things down. Be clear and opinionated. Wrong gets corrected, vague sticks around.

- There will always be trade-offs. Be clear what you're optimizing for when you make decisions.
- Communicate frequently with your audience in mind.
- Expect problems to arise. Make plans that assume there will be changes in direction, people quitting, and unavailable dependencies.

## WHY HAVE WE STOPPED?

- As the project lead, you are responsible for understanding why your project has stopped and getting it started again.
- As a leader in your organization, you can help restart other people's projects too.
- You can unblock projects by explaining what needs to happen, reducing what other people need to do, clarifying the organizational support, escalating, or making alternative plans.
- You can bring clarity to a project that's adrift by defining your destination, agreeing on roles, and asking for help when you need it.
- Don't declare victory unless the project is truly complete. Code completeness is just one milestone.
- Whether you're ready or not, sometimes it's time for the project to end. Celebrate, retrospect, and clean up.

## YOU'RE A ROLE MODEL NOW (SORRY)

- Your words and actions carry more weight now. Be deliberate.
- Invest the time to build knowledge and expertise. Competence comes from experience.
- Be self-aware about what you know and what you don't.
- Strive to be consistent, reliable, and trustworthy.
- Get comfortable taking charge when nobody else is, including during a crisis or an ambiguous project.
- When someone needs to say something, say something.
- Create calm. Make problems smaller, not bigger.
- Be aware of your business, budgets, user needs, and the capabilities of your team.
- Help your future self by planning ahead and keeping your tools sharp.
- Write things down, even when they're "obvious."
- Expect failure and be ready for it.
- Design software that's easy to decommission.
- The metric for success is whether other people want to work with you.

## GOOD INFLUENCE AT SCALE

- You can help your colleagues by providing advice, teaching, guardrails, or opportunities. Understand what's most helpful for the specific situation.
- Think about whether you want to help one-on-one, level up your team, or influence further.
- Offer your experience and advice, but make sure it's welcome. Writing and public speaking can send your message further.
- Teach through pairing, shadowing, review, and coaching. Teaching classes or writing codelabs can scale your teaching time.
- Guardrails can let people work autonomously. Offer review, or be a project guardrail for your colleagues. Codify guardrails using processes, automation, and culture change.
- Opportunities can be much more valuable than advice. Think about who you're sponsoring and delegating to. Share the spotlight in your team.
- Plan to give away your job.

## WHAT'S NEXT

- You are responsible for your career and choices. There are a lot of options about what to optimize for. Know what's important to you. Be deliberate.
- You'll increase your access to opportunities with skills, visibility, relationships, and experiences.
- Everything is learnable if it's worth the time investment.
- Check in with yourself occasionally and make sure your role is still giving you what you need. Look at what's good as well as what's not working.
- There are excellent reasons to spend a long time with one employer. There are excellent reasons to move around too. Either way, you have several options for paths onward.
- Software has a massive influence on the lives and livelihoods of just about everyone on earth. Take the responsibility seriously.