

The Benefits of Process Automation

Improve Engineering Through Process Automation

**Reducing time
to market and
empowering your
engineers are
strong arguments
for process
automation.**

4

Introduction

There's a lot of talk in the news today about automation.

5

The Benefits of Process Automation

Broadly speaking, process automation has a wide variety of benefits for any organization

7

The Humble Beginnings of Automation

Automation is older than you probably think -- a lot older

10

What Tasks Can Be Automated?

There are four areas to look at when looking to see if a task can be automated easily or not

11

Use Case for Automation: Testing Automation

Testing automation is an area where you can get a quick and easy win.

13

Use Case for Automation: Code Testing

"Infrastructure as code" is a growing trend in IT and for good reason.

14

Deploying Automation In Your Organization

If you're sold on an idea of automation, but not sure how to incorporate it, you're not alone.

Introduction

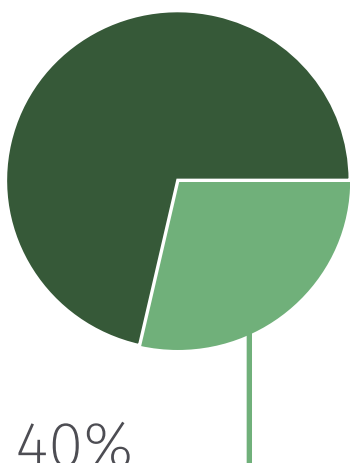
Engineering organizations can benefit immensely from automating redundant, repetitive, time consuming and just plain boring tasks out of existence

There's a lot of talk in the news today about automation. A recent report from PwC estimates that we're looking at losing just shy of four in ten jobs to automation over the next 15 years. But automation isn't just about streamlining industrial production or self-driving truck fleets. Engineering organizations can benefit immensely from automating redundant, repetitive, time consuming and just plain boring tasks out of existence. This is known as process automation.

Basically any task that can be repeated and involves following a series of steps can be automated. This means that your engineers no longer have to engage in time-consuming drudge work that costs you thousands

and lowers their morale. Instead, they can spend their time doing what you need them to do and what they want to do -- designing better products for your customers. We've found an average of a 29 percent increase in achieving product launch date targets through automation. Perhaps even more impressive than that, we've found a 27 percent increase in meeting revenue targets through automation.

Letting your engineers be engineers isn't just about designing better products, as important as that is. In today's tight labor market for top engineering talent, bogging your engineers down with a lot of boring, repetitive tasks is a surefire way to breed low morale. This, in turn, makes it far more difficult to attract and retain top talent, increasing your overhead and lowering the overall productivity of your workforce.



We're looking at losing just shy of four in ten jobs to automation over the next 15 years.

Source: PwC Report

30%
25%
20%
15%
10%
5%
0%

29%

29 percent increase in achieving product launch date targets through automation.

30%
25%
20%
15%
10%
5%
0%

27%

27 percent increase in meeting revenue targets through automation.

Source: Aberdeen Group



The Benefits of Process Automation

Broadly speaking, process automation has a wide variety of benefits for any organization, including:

- Increased productivity and increased quality of product.
- More predictable, consistent and uniform processes.
- Reduced labor overhead.
- A shorter time-to-market cycle, key in a world where rapid product updates are now standard.
- The elimination of drudge work, allowing your entire workforce to concentrate on more engaging and challenging tasks.
- An engaged and challenged team, which positively impacts retention as well as productivity.

We believe the strongest argument for implementing process automation is attracting and retaining top talent, something that pays dividends over time. Edwin Chung, SPK's Engineering Services Manager, recalls an early job he had at Stanford School of Medicine. For the most part, he was assigned repetitive and redundant tasks. While acknowledging that delegation is just part of how business is done, Chung is quick to say "I felt like my time was being wasted. No engineer wants cutting and pasting in an Excel spreadsheet to be part of their job description." So instead of delegating repetitive tasks, why not automate them?

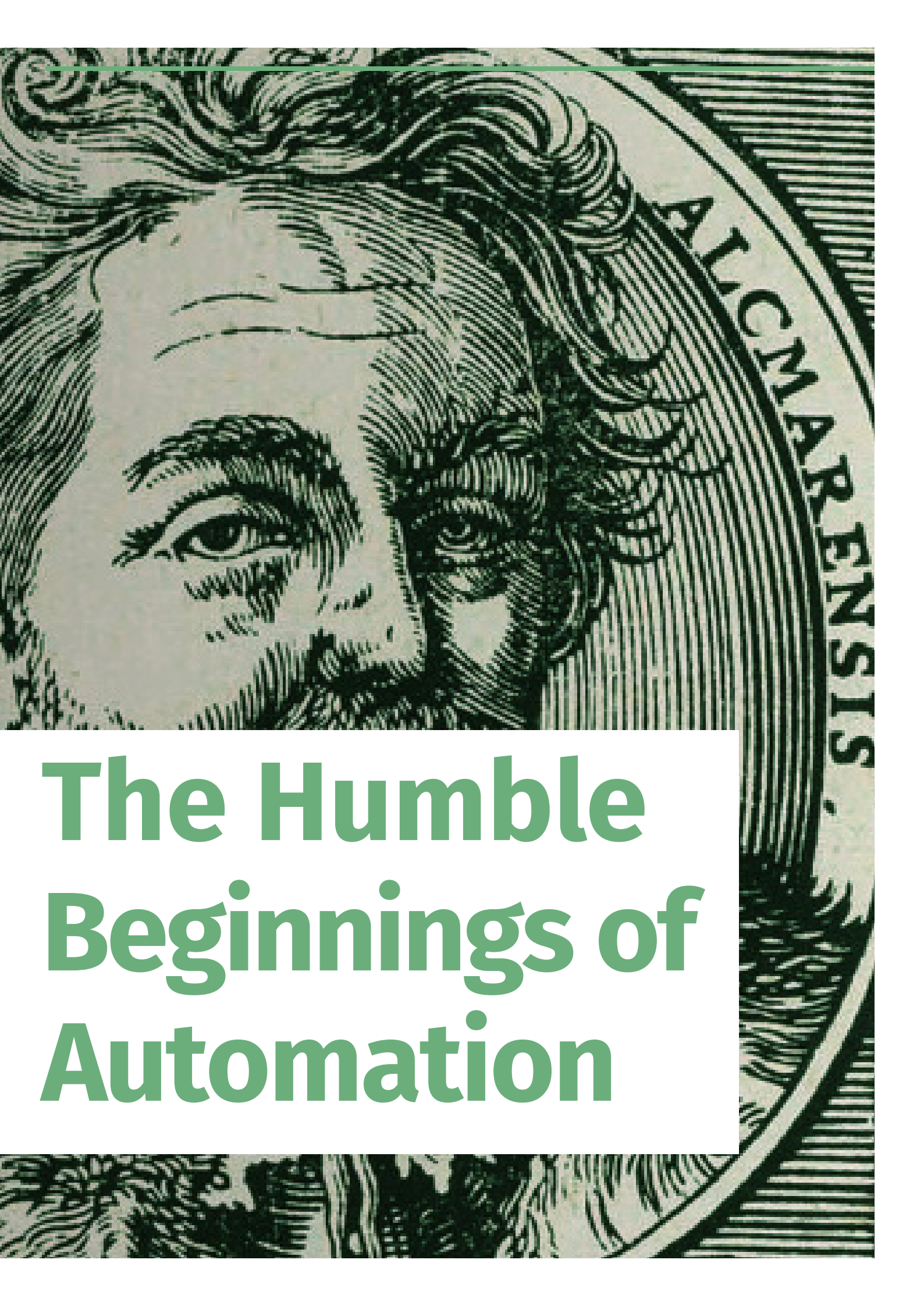
Automation promises to transform your engineer's workload from a series of low-value tasks into high-value deliverables. "Automation cannot perform original creation," says Chung, "but that works out because engineers want to perform creation." In fact, one of the best places to look for automation opportunities is simply by asking your employees what repetitive tasks they feel are wasting their time. They're likely to have a long list. This list won't be a series of gripes, but rather a list of places where you can immediately start saving time and money while boosting morale among your engineering team.

Reducing time to market is another strong argument for process automation. Today's customers expect quick revisions and an almost breakneck time to market cycle. In a worst-case scenario, your competitors could beat you to market with a similar product and enjoy the "Facebook effect," while your product is seen merely as an imitator, no matter how much better it might objectively be.

When considering automation, the key question isn't how much it will cost to automate processes. It's how much it will cost your business -- in lost talent and reduced time-to-market compared to your competitors. In fact, the up-front cost of automation can very quickly pay for itself in terms of payroll hours spent on original creation and stronger performance metrics.

"Automation cannot perform original creation," says Chung, "but that works out because engineers want to perform creation."





The Humble Beginnings of Automation

Automation is older than you probably think -- a lot older.

The roots of process automation lie in mechanical automation.

Dutch scientist Cornelius Drebbel invented the thermostat in 1620, which is the world's first automated device. By the 1930s, nearly all organizations with major electrical and systems needs had a sort of semi-automated system. Signals were sent to a control room where workers manually switched processes on or off depending on needs. Aside from balancing a variety of ons and offs, there wasn't much nuance to early automation. A process was either enabled or disabled. Digital control came in the late 1950s, with Texaco's Port Arthur chemical refinery going all digital in 1959.

The impact of automation was nearly instant in the realm of manufacturing. In 1905 the first automated glass bottling

plant opened. A two-man team produced 17,280 in 24 hours, compared to manual glass blower teams of six men producing 2,880 bottles in the same time period.

The roots of software automation as we know it lie in the phone system. The first dial telephones were introduced in 1892 and with them came automated telephone switchboards. By 1929, nearly a third of the Bell System was automated. This automation relied upon vacuum tubes consuming so much energy that it was feared that the phone system would soon require all available power in the United States, prompting the development of the transistor.





1620

Dutch scientist
Cornelius Drebbel
invented the thermostat

The first dial
telephones were
introduced in 1892
and with them came
automated telephone
switchboards.

1892

1905

First automated glass
bottling plant opened

1929

By 1929, nearly a third
of the Bell System was
automated.

By the 1930s, nearly
all organizations
with major electrical
and systems needs
had a sort of
semi-automated
system.

1930's

1950's

Digital control came in
the late 1950's

Texaco's Port Arthur
chemical refinery
went digital

1959

What Tasks Can Be Automated?

There are four areas to look at when looking to see if a task can be automated easily or not:

- **Repeatability:** Original creation cannot be automated. What's more, there's little point in automating a task that will only be done once or twice. Look for tasks that will be performed hundreds or even thousands of times. That's where you're going to get a solid return on your automation investment.
- **Stability:** Tasks that are constantly changing can be automated, however it's far more difficult and time consuming to do so.
- **Complexity:** Similarly, the quick and easy wins in automation are found in simple tasks rather than incredibly complex ones. Again, like unstable tasks, complex tasks can also be automated, however it might not be worth the time, money and energy spent to do so.
- **Wastefulness:** Some tasks are simply wasteful when performed by engineers. For example, say that you have to copy and paste a product part number from one spreadsheet to another. Now assume that this has to be done ten times for every product part. There's just no point in paying an engineer to do this, not only because it costs more, but because that engineer will soon be updating his resume and looking for more engaging work.

If you spend just a little bit of time thinking about it, you can probably come up with half a dozen tasks off the top of your head your organization could automate. Bring in the expert team at SPK and we can find dozens more.

The background of the slide is a dark teal color with a blurred pattern of code snippets in various colors (green, yellow, white). The snippets include words like 'ue', 'lse', 'R_Y', 'R_Z', 'active', 'modifier', 'ed_object', 'e].select', 'exactly', 'selected', and 'x'.

Use Case for Automation: Testing Automation

Testing automation is an area where you can get a quick and easy win. Think about it: Many testing tasks involve repetitive drudge work. Regression testing is a perfect example of this. No engineer wants to do it and you can easily automate it off of their plate. This frees them up to tackle far more engaging and rewarding tasks in the testing process, creating greater value and raising morale all at the same time. Finally, tasks like regression testing are far more accurate when automated, reducing the amount of error significantly.

Debugging and logging are two more laborious and uncreative tasks no engineer wants to perform. They are also highly repetitive and stable, meaning that they are low-hanging fruit in terms of automation. Much like regression testing, no engineers want to spend their day debugging and logging.



And, of course, during the testing period your documentation will change. Particularly for medtech organizations, this can be a highly cumbersome process. The FDA has strict regulation regarding documentation revisions and revision checking. Revision processes can easily eat up a thousand man hours every year. Add a foreign or non-alphabetic language into the mix and this number can triple or more. However, an automated revision system can save you thousands every year. Here's how it works:

1. The automated system downloads the old document.
2. The automated system downloads the new document.
3. Both documents are plugged into a comparison tool like the kind in Adobe Acrobat or Microsoft Word.
4. The result is uploaded with only the revisions highlighted, ready to be checked by the relevant engineers

That's it. And it doesn't matter if the document is in English or Japanese. The amount of time saved will be significant. The reviewing itself can't be automated, but 99 percent of the process can be automated by flagging what people need to look at and then handing it to them. Once again, the error rate will be significantly reduced through automation as well, saving you more time.

Similarly, you might have to copy and paste information from one document to another. But you can use single point of truth to store the information in one place and automatically insert it where it needs to be. One person or department has the responsibility of validating and updating the information. For everyone else this is read only.

Use Case for Automation: Code Testing

“Infrastructure as code” is a growing trend in IT and for good reason: It allows for a more modular and easily deployed infrastructure. Organizations can adopt agile operations, leverage automated testing and deploy infrastructure resources such as servers and storage. Software-defined infrastructure builds predictability into resource deployment, as standard templates can be defined and implemented for various virtualized devices.

More elastic and usage-based infrastructure means constant deployment and retiring of servers. Release and deployment can be exhausting, especially for engineers spinning up the resources for deployment. Server positioning, however, is an easily automated task offering great relief to overtaxed engineers. Still,

spun up servers need configuring and testing prior to production. Infrastructure as code, however, allows for automated tests, picking out infrastructure bugs, spotting configuration errors, identifying connection errors and other important troubleshooting tasks. This can reduce overall deployment time from days and weeks to hours and even minutes.

Automation provides a myriad of benefits from cost reduction to efficiency. It can make up for a lack of engineering skillsets, carrying out tasks that might require three engineers. Ultimately, however, automation gives time back to your Ops team. Now they can spend their time doing engaging, empowering work and overcoming important challenges rather than babysitting servers or performing unfulfilling backup tasks.



Deploying Automation In Your Organization

If you're sold on the idea of automation, but not sure how to incorporate it, you're not alone.

Many organizations aren't sure where to start. What's more, your engineering department might not have the skill set required to automate tasks. Sometimes even experienced engineers don't understand the difference between simple scripts and process automation. Scripting can often be a

useful way to carry out tasks, but can also lead to isolated solutions rather than holistic processes that integrate your entire IT environment and operating model. So where do you begin integrating automation?

1. The starting point: Tear down operational boundaries

Automation of processes often affects multiple functions and engineering groups, which requires collaboration between multiple areas. As seen in DevOps, it is becoming increasingly important to merge functions together around solutions, rather than traditional ITIL functions (e.g. release and deployment, change management, etc.). Start by holding collaborative sessions for engineering teams to explore areas of opportunity for IT automation. Weaving these teams together provides better automation solutions, which will have a greater business value.



2. Hunt for Quick Wins

Ideally, your joint engineering teams can identify some initial quick wins to concentrate on. That might mean automating smaller workflows like service ticket generation or processing approvals. Engineering and operations must carefully plan all automation strategies, ensuring newly automated workflows fit into the broader automation strategy. Stand-alone scripts and automation projects that don't fit into the larger picture tend to create confusion and competing workflows. That's going to create more headaches, not fewer.

3. Focus on the Larger Picture

After automating smaller tasks, engineers will have a better understanding of automation platforms and process. That provides them with the confidence and experience to begin automating on a larger scale. Workload automation automates scheduled jobs like provisioning new computing assets, executing backup jobs and retiring assets.

Larger automation projects like these can provide enormous benefits to engineers. The nature of their jobs will shift from monotonous tasks to more challenging, exciting and ultimately rewarding roles like development of new solutions and problem solving.

Remember: Engineers want to build and create. Refocusing their attention on doing what they love will boost morale. Engineers will now provide approval and oversight for mundane tasks like release and deployment management. The lion's share of their time will be spent on far meatier tasks.

When combined, the engineers of SPK have decades of experience both in their fields and in implementing automation at enterprise-class organizations and SMBs. When you hire us, our engineers go to work for your engineers, freeing them from the tasks they don't want to -- and shouldn't be -- performing on a daily basis.

Best of all, we can continue working with you after addressing your automation needs to manage your tools so your engineers don't have to. [Call us today](#) to talk about your organization's needs and how SPK and Associates can help you meet them.

Improve Engineering Through Process Automation



[visit \[spkaa.com/contact\]\(http://spkaa.com/contact\)](http://visit.spkaa.com/contact)



Call us 888.310.4540



Visit us at 5011 Scotts Valley Drive, Scotts Valley, CA 95066

We are an information technology services company completely focused on fulfilling the specialized needs of product development teams. SPK deeply understands the business of R&D. We live and breathe engineering. We translate that into tech solutions designed to speed up product design, software development, and release while improving your product quality. We work for your engineering team, helping to free them of mundane, repetitive and distracting tasks and let them do what they do best -- design a better product.

Our company was co-founded in 1997 by [Steve Kling](#), the Western Region Professional Services Manager for Hewlett-Packard, and [Christine \(Chris\) McHale](#), a business manager for the same HP consulting organization.

