Coping with Technical Debt

By

Heather A. Smith
James D. McKeen

The IT Forum …
Is a focus group of senior IT managers from a variety of different industries convened regularly by the authors to address key management issues in IT. This report highlights a recent discussion.

— See back page for details of the IT Forum and other reports.
Introduction

Technical debt is not a very captivating topic. Talking about it is akin to going into vast amounts of detail about the planned maintenance for your car -- both the regular stuff and the surprises you didn’t know about until someone knowledgeable looked under the hood. Most of us, if we’re honest, just want the whole thing to go away, resent the time and cost involved, are shocked when the mechanic gives us unpleasant news, and (sometimes) curse the car companies for not making cars that are more durable. Yet we all accept car maintenance as a necessary evil and though we may put it off for a short while, know that we are asking for trouble if we aren’t responsible about doing it. This analogy isn’t perfect because technical debt is immensely more complex and expensive than car maintenance, but it is one that gives a non-technologist a glimpse of what we mean when we speak about how systems and technology need regular maintenance and upkeep and sometimes bigger and more expensive changes.

When one considers that most organizations operate with multiple generations of technology from different vendors, ranging from the newest cloud and mobile systems, to a variety of off-the-shelf software, such as ERPs, CRMs, and data stores, to home grown systems written in several programming languages, it should not be surprising that one or more of them may need maintenance or upgrading at any one time. Furthermore, as new layers of technology are added, IT has to ensure they all work together. What’s still running many businesses are the large, complex legacy systems and technologies from thirty or more years ago that now need to integrate with a wide variety of newer ones in ways that were never intended. One CIO has compared it to operating a 1970s car while adding all of the features of a modern car – everything from new cabin features (e.g., GPS, Bluetooth, heated seats, and cup holders) to what’s under the hood (e.g., improved fuel economy, emissions reductions, and antilock brakes) and expecting it all to work together smoothly.

Whenever replacing older systems and technologies with something more adaptable and effective is discussed, senior leadership typically rejects this as too expensive with too little ROI. So, IT carries on – deconstructing these systems into layers, hiring programmers with older system skills, and patching over problems as best they can – while the problems grow bigger and more complicated. The cost of deferring the work to streamline these technologies, make them easier to work with, or to replace them is known as “technical debt”. And until this debt can be better explained and quantified in business terms, it is unlikely that business leaders will give it the attention and respect IT leaders know it deserves.
This paper explores the nature of technical debt in IT today. It first articulates its growing scope and multiple dimensions and then examines its underlying causes. Next it describes some of the risks and opportunities technical debt presents to organizations and the challenges that IT leaders have in managing it. Finally, it offers some practical advice for managers in how to better cope with this difficult and complex problem.

What is Technical Debt?
The term "technical debt" is a metaphor used to describe the cost to an organization of supporting imperfect or aging technology. Like a financial debt, where money is borrowed now to achieve a goal (e.g., buying a house or a car) and paid back with interest at some point in the future, technical debt refers to imperfect choices that are made now to achieve a goal (e.g., to save money or to create a new product more quickly) that will need to be remediated at a higher cost in the future. The term originally focused on software and how developers sometimes knowingly or unknowingly decide not to follow best practices or standards to deliver a system more quickly. Since then however, the term has been more broadly and loosely used to encompass a large number of technology challenges related to aging hardware, software, tools, and practices that don't meet current needs or standards or which are not supported. For example, the focus group noted that coding business rules into an application is not considered best practice today but there are still older systems in their companies that have these in place. The cost of remediating this issue is considered technical debt.

There is broad agreement that limiting the concept of technical debt to software quality is inappropriate today because the scope of the technology landscape is much broader and more complex than in the past (Buchanan et al. 2017). Thus, the concept needs to be reworked and expanded to cover the many issues associated with technology development and change that can create a variety of remediation needs and costs for an organization (Kyte et al. 2018, Raskino 2018, Wisner et al. 2018, Dawson and Cappucio 2017). The current lack of a clear definition of technical debt only increases the challenges involved in identifying and managing it (Kirsch 2017). What is needed therefore is a more comprehensive taxonomy of what technology debt is and the factors that affect it both positively and negatively (Burden et al. 2018).

The focus group discussed and described the many ways technical debt can be manifested in their organizations. As shown in Figure 1, technical debt is an integral part of the creation and operation of many different IT products and services but it is also be greatly influenced by three larger factors that can exacerbate or mitigate the creation of technical debt over and above decisions made about any individual IT product and service. Each of these is discussed below.
Sources of technical debt. Technical debt can be incurred in a number of areas of an organization's technology landscape, such as:

- **User experience.** This is one area where technical debt is most visible because poor design, availability, service quality, and system response all have an impact on how users perceive the quality of their experience with technology (First Mark 2018). "We finally gave all our end users tablets for their work but because management wouldn't pay for increased bandwidth, they can't be used," said a member in frustration.

- **Data.** Difficulties accessing data, low quality data, duplicate data, and data that no one owns are all indicators of technical debt which can cause serious problems for an organization. "We have 20,000 shared folders of 'stuff' that no one owns and which someone must go through," said a member. "We can't get rid of them because they may contain information that must be retained for regulatory reasons or they may break applications."

- **Integration.** Multiple technologies, failure to upgrade to newer versions of software, and lack of adherence to current standards all create a heterogeneous technical environment that can grow increasingly challenging to integrate and require more and more time to maintain (Entertainment Close Up 2018). "Integration is a huge anchor for us," said a member. "It's extremely expensive to keep ensuring everything works together."

- **Architecture.** Both insufficient and over-designed architecture can limit an organization's ability to change in a timely way (Wikipedia 2019). Decisions (or lack of decisions) that create duplication, choices of non-standard technology, and deviations from the accepted roadmap for the future can create an environment which is highly complex, containing multiple versions of software, or which has problems synchronizing data and accommodating upgrades (Blosch et al. 2016). "We now have 14 different analytics programs," said a member. "Why??"

- **The Development Process.** Development practices, standards, and tools have changed over time so many companies still have a variety of artifacts that do not conform to what is expected from developers in the present day. Almost every aspect of the development process from requirements, to design, documentation, and the tools used can create gaps in expectations, understanding, and knowledge and in turn can create technical debt that will cause friction for future developers (Buchanan et al. 2017).
Software. In their haste to develop products quickly developers can create software that meets all functional requirements but which can have usability, reliability, maintainability, reusability, portability, security, and performance challenges which are costly to repair retroactively (Ramasubbu et al. 2015, Conroy 2012).

Testing. Inadequate testing tools or scripts can create a need to manually retest after every release of software or can result in insufficient test coverage, unexpected down time, and challenging recovery activities (Kirsch 2017).

Operations. As with software, not adopting infrastructure and operations standards or deferring upgrades can create a highly complex environment that can put a company at real risk of reliability, security, and performance issues (Kirsch 2017).
Infrastructure. As infrastructure ages, and without upgrades and maintenance, many things can happen. Performance can drop off, components may be unavailable, or users may simply experience "busy" signals (Stephanie W 2015). These issues all impair a company’s ability to recover from a range of difficulties. Furthermore, with companies now moving parts of their operations to the cloud, new layers of infrastructure complexity are being added over which a company has no control. And with the reduced operations staff that result, firms can now lack depth of knowledge or have single points of knowledge that can leave them vulnerable in managing their in-house infrastructure.

Factors that Influence the nature and amount of technical debt. In addition to these specific sources, the focus group pointed out that three broader factors can greatly multiply or reduce the amount of technical debt in an organization.

1. **Technology evolution.** The more different types of hardware and software a company has to manage and the faster it changes, the more likely it will incur technical debt. The larger number of software packages and operating systems that are commonly used in companies these days, the many new types of technologies that are being introduced in organizations (e.g., various forms of AI), and the speed of technology evolution can greatly affect many of the types of technology debt listed above.

2. **The current technical environment.** The less standard and more diverse a technology environment a company has, the greater the likelihood it will incur technical debt. One focus group member noted, "We now have so many APIs, just diagramming them looks like a hairball. They’re impossible to manage well!" Many factors that increase technical debt are not of IT’s choosing. Technology can be inherited from acquisitions that is difficult to integrate, doesn’t conform to company standards, and is expensive to manage, or required by the geographic locations in which a company operates. An organization’s current technical environment therefore not only forms a baseline from which IT must tackle technical debt but this environment can change fairly quickly as a company itself evolves and takes on new challenges.

3. **Organizational culture.** Focus group members stressed that culture is a significant factor in how an organization accumulates and deals with technical debt. "Culture affects the decisions we make about the number and types of systems we buy, how we set up roles and responsibilities for managing our technology, creates shadow organizations, and affects how IT is perceived in the organization," said one member. For example, one manager’s team was told by a senior business leader that they had to work five times faster and that “he didn’t
care how they did it”. Such attitudes, combined with budget pressures, can both create more technical debt and prevent existing debt from being dealt with.

What Causes Technical Debt?
Some degree of technical debt is inevitable simply due to age and change (Dawson and Cappucio 2017). Known to professionals as "bit rot" it happens slowly over time through incremental changes that gradually increase complexity and decrease performance (Herschmann 2018). At its worst, software can become so patched and corroded that it becomes too fragile to change (Buschmann 2011). Many legacy systems have unfortunately deteriorated to this point. Technical debt also grows because the business evolves, new technology emerges, and standards and practices change (Blosch et al. 2016). IT’s challenge is to ensure that this type of technical debt does not become so unmanageable that it introduces risks and vulnerabilities into the organization (Dawson and Cappucio 2017). "We’re incurring technical debt in our innovation and emerging technology areas as well," said a focus group member. "Debt introduced by our digital programs is so new we can’t even estimate it or plan for it. As a result, our technology leadership may not see it as a big issue – yet."

Other causes of technical debt are more deliberate. Pressure to work fast leads to cutting corners and thus increased debt (Wisner et al. 2016, Burden et al. 2018). One member told of a vice president who wanted to "see new stuff every day". Pressures to maintain leading edge IT and produce new products and services quickly naturally lead technologists to prioritize visible features such as functionality, a minimum viable product, new technologies, or speed of service, over invisible ones such a quality, performance, flexibility, and ease of maintainability (Conroy 2012, Buchanan et al. 2017). With IT staff under tremendous pressure to deliver, there is no time to remediate short term fixes or architectural short cuts leading organizations to embed these problems permanently and causing future technical debt.

Another cause of debt is diversity of technology both in terms of types of hardware and software and different versions of these. Focus group members told of applications running critical functions that weren’t supported by their vendors anymore. Others noted that end user computing technology, which has exploded in recent years, is not always kept up to date. "We have hundreds of applications built in SharePoint that hasn’t been upgraded since 2010; if we upgrade it all these applications won’t work anymore," he said. An easy measure of this form of technical debt is to ask: How many different versions of the same software do we use? How many applications are we running that are no longer supported or updated regularly? and How many different standard desktops do we have? This diversity is compounded when technology from
multiple vendors is introduced requiring different types of specialized expertise. Even something as simple as BYOD can cause havoc when a new version of a cell phone operating system is introduced. “We have to warn people not to upgrade their phones until we tell them it’s safe or they won’t be able to access their apps,” said a member. “The business has no idea of how much work is caused when something like this happens, and it happens all the time.”

Other causes of technical debt come from both business and technical leadership. Business leaders who don’t understand technical debt often fail to follow standard practices in their own business unit computing and make business decisions without reference to technical considerations and advice because they feel they need to move more quickly (Winser et al. 2018). However, “When we move too fast, things don’t quite work well,” explained a member. “For example, one business unit acquired a product without reference to our architectural vetting process and it affected our financials.” “Our business leaders don’t want to spend the money to upgrade our old technologies so we are constantly putting band aids on it,” said another. Last minute changes, poor communication, and business uncertainty about what it wants can also create technical debt (Winser et al. 2018).

IT leadership can also be responsible for creating technical debt. Parallel development or outsourcing some or all development can cause problems when different components are put together (Burden et al. 2018). “When we outsource code, we always have to check it against our standards, for complexity, and for the cost to maintain as we’ve had problems,” explained a member. Leaders who under-invest in skills development or training can end up with staff who are unequipped to keep up with new technology developments or which have Single Points of Knowledge (i.e., where there is only one person who understands a technology). “Sometimes we have older technology but no skills to change or support it,” said another member. “Then the organization is truly at risk.” IT leaders too must balance the cost of remediating technical debt with all of the other demands on IT services and often choose to increase debt as a result. "Our IT leaders rejected an API management system because of the cost and now we have a mess!," said one member. "They may not see it as a big issue but we do." Similarly, the movement of applications and infrastructure to the cloud may have reduced direct costs, but with the cadence of cloud upgrades now out of IT’s control, it is causing considerable work in-house as staff struggle to keep diverse hardware and software working together.

Technical debt and its causes can be invisible to all but a few specialized technologists but business and technical leaders can begin to understand the amount and causes of technical debt in their area by keeping two rules of thumb to keep in mind (Conroy 2012):
1. If not proactively addressed, technical debt will get worse over time.
2. The rate of acquisition of technical debt is proportional to the rate of change in an area.

The Risks and Opportunities of Technical Debt

The lack of clarity about the nature, scope and causes of technical debt means that it is extremely difficult to explain it to the business (and even technology) leaders who will ultimately fund and prioritize its remediation. Its invisibility to the lay eye makes it easier to ignore. But there are real costs involved with this approach that can lead to significant problems for organizations if they fail to properly manage their debt and some real benefits that can be derived if they do.

Although the dollar cost involved in identifying, managing, mitigating and working around technical debt is what first comes to mind when the subject is raised, these represent just the tip of the iceberg of risk to an organization from technical debt. Clearly, it costs time and money to design and implement technology well and while there are often immediate business benefits to be gained from taking short cuts, the longer they are allowed to be part of the organization's technology landscape, the more likely it is that they will contribute to these risks.

The focus group identified several significant risks arising from technical debt:

- Difficulty Innovating. The challenge involved in updating and connecting the diverse technologies present in most organizations has already been mentioned but these also make it extremely difficult to scale innovation and migrate to new technologies, thereby making digital transformation more difficult (Alves et al. 2016, Haight and Rosenstein 2014). One study found that C-suite executives are finally beginning to recognize this situation with 70% saying that technical debt inhibits IT’s ability to innovate and 69% feeling that it makes IT less responsive to market changes (Alves et al. 2016).

- Productivity. Cutting corners and taking short cuts does speed up the delivery of functionally in the short term – by up to three times faster (Ampatzoglou et al. 2015). However, if this debt is allowed to accumulate it will, in the longer term slow delivery down making technology more difficult to modify and update and requiring IT staff to spend more time managing and working around it (Kyte et al. 2018). Thus while "high debt" projects do initially deliver faster, they soon reach a plateau and are less able to deliver; in contrast "low debt" projects develop more slowly but ultimately deliver more functionality to an organization (Ampatzoglou et al. 2015).
• Reliability, Performance, and Security. "When standards are not followed, vulnerabilities are created and these are an impediment to moving forward," said a member. As weaknesses accumulate, technology can crash, go offline, or simply break down (Dawson and Cappucio 2017). Studies have found that technical debt tends to increase the risk of system failures by 62% (Ampatzoglou et al. 2015) and lead to higher maintenance and operations costs (Conroy 2012). Each of these vulnerabilities in turn can lead an organization open to financial losses (Blosch et al. 2015).

• Morale. Technical debt also affects IT staff, making them work longer hours and lowering their morale (Winser et al. 2018). Fixing problems is one of the least fun parts of IT work and can directly affect employee retention, said the focus group. One member noted, "As we accumulate more and more technologies there's a growing feeling of hopelessness – that we can't solve this problem. But it's really more that we haven't done it because addressing it is at the discretion of the business." Another added, "Our new employees get to see the problems before they can even get started working. In one case it took four weeks to get someone the technology they needed and then the software conflicted with the hardware and they still couldn't do anything!"

• Customer/User Satisfaction. Ultimately, if a product has debt in some or all of the areas outlined above, it will affect customer satisfaction (Wisner et al. 2018). Problems with usability, errors, or slow response time can lead to a drop-off of up to 50% in satisfaction rates (Ampatzoglou et al. 2015).

"Overall, dealing with technical debt is really about managing risk," said a member. This risk is exacerbated when an organization fails to deal with the known knowns and the known unknowns involved and results in technical debt becoming an amorphous blob of unknown unknowns. "We need to do a better job of articulating the risks of technical debt in terms the business can understand and making sure that all decisions we make clearly articulate the trade-offs involved," said a manager.

Interestingly, there are also some benefits of effectively dealing with technical debt. Companies following GAAP or IFRS standards may gain a tax deduction if they reduce debt (Boomer 2016, Kyte et al. 2018). Greater adherence to regulatory and compliance rules may lead to fewer unpleasant surprises at audit time, said the focus group. And paying down technical debt can lead to significant savings in cost and efficiency (Schwartz 2018). Finally, the group noted that it will be much easier to take advantage of new opportunities in a timely fashion, if an organization's technology is more adaptable.
The Challenges of Managing Technical Debt

The biggest challenge facing IT leaders with technical debt is how to identify it, measure it, and articulate and quantify the risks and opportunities involved to the business (Ramasubbu et al. 2015, Kirsch 2017). "We have a hard time identifying the root causes involved," said a member. There are various tools advertised for identifying technical debt and many of the focus group companies use them, but these only identify certain types of debt, such as in software quality, and do not deal with issues related to architectural design, data accessibility or integration (Buchanan et al. 2017). Furthermore, there is no guidance available for determining when technical debt is acceptable (Entertainment Close Up 2018). "We're trying to drive down our debt but we're not sure whether we are or not," said a member. "We lack metrics for this." And, as noted above, many non-functional aspects of technical debt, such as maintainability, productivity implications, and lack of agility, may not be visible to either IT or business leaders (First Mark 2018).

This vagueness about technical debt, what it is and what it costs, leads to a second challenge for IT leaders: lack of interest in dealing with it. "Technical debt is difficult to explain to the business because they don't really care," said a member. "We have no visibility on this issue." "People don't care about these issues," said another. "They just want us to keep the systems running so we keep patching them. This really prevents us from being good stewards." In many cases, business leaders have only a short term point of view and aren't interested in potential future problems (Conroy 2012). "Corporations tend to have ADD," said a member. "Executives don't stay around for long so they don't want to see this problem." "We need to develop common objectives," explained another. "Right now they only want us to deliver, but we should be seeking to deliver something good that runs well." "If we knew how much time we spend on maintenance, for example, versus transformation," said a third, "we might be able to interest the business in investing in sustainability. We need to make this part of the equation from the start because when technical debt mitigation is needed (e.g., when it shows up as a security vulnerability), we can't fund it."

IT organizations tend to address technical debt issues mainly when they become problems, but this approach is often costly once a system is in production (First Mark 2018). "Our executives don't want to deal with this so they wait and wait as the problems pile up," said a member. Studies agree with this perception showing that leaders want the benefits of having new types of technology and recognize the limitations technical debt places on their ability to innovate and grow, but they also want to keep their legacy systems running (Alves et al. 2016). "The cost of dealing with technical debt is very important," said a member. "Typically, there's no money for dealing with it until it blows up." Lack of funding is exacerbated by the fact that many of the
costs and issues of dealing with aging assets are at the enterprise level, while the budgets tend to be at the business unit level. "We need enterprise funding to deal with data warehouses and integrating technology from acquisitions, for example," said a member. "Ideally, we need a technology investment fund but where’s this money going to come from?"

With business leaders not prepared to invest in addressing technical debt because they don’t see a short term benefit in it, dealing with it has become an IT problem. Because no one in IT has the specific responsibility for dealing with it, individual IT groups shoulder the burden of the increased maintenance and operations costs involved, because "it’s the way we do things", and technical debt and its cost tends to grow and fester.

Within IT, much of the burden of managing many types of technical debt falls to the architecture group to deal with. Architects are often charged with establishing standards for new development so that technical debt will not increase and must balance rigidity with being overly flexible in enforcing them (Smith and Watson, 2015). In the focus group companies, architects are responsible for seeing that there’s no duplication of technology and ensuring that standards are met. "We try to enforce our standards but it’s not always easy. Many projects feel that they are different and that the standards shouldn’t apply to them," said a member. "But we feel that 99% should meet the standard." To deal with this conundrum, architecture review boards often issue waivers to allow projects not to meet a standard in the short term, while tracking them to ensure they are remediated in the longer term so these deviations don’t cause increasing technical debt. The problem comes when these waivers are not dealt with and the architecture group is left with literally thousands of outstanding waivers the business won’t fund to remediate.

Architecture is also responsible for managing and/or reducing interdependencies in their organizations' technical environment (Buchanan et al. 2017). Interdependencies represent a huge challenge in dealing with technical debt because of the increasing amounts of change that this environment must absorb both from within and more and more from external sources as well. "Business units are growing in their interdependency on each other, but we struggle to get them to work together and make decisions together," said a member. "The little things really matter or we'll end up with five things when we want one and then we need to integrate them because the business units couldn’t agree." Externally, vendors are increasing the cadence of their changes they make to cloud or off-the-shelf software. "The support model is changing and vendors are pushing upgrades every few months," said a member. "Every time they make a change to a piece of software, everything that uses it must be tested." In the long-term, keeping all software up-to-date is a good thing said the group, but this is a very painful issue for IT and it's highly disruptive. "The rate of change exacerbates debt and we’re always having to do
unplanned work that's not on deliverables," said another. "Sometimes, an innocuous change can take down a business unit; in other cases, such as when we’re upgrading to a new version of our internet access software, the whole company is affected and it’s a two year project for us," said a member. "Right now we have people working long hours just to keep up."

Overall, this is a situation that is leading to considerable individual and organizational burnout. Years of underinvestment and asset sweating have left many companies with their IT assets in poor condition, too fragile to change, at risk of cyber attack, and extremely costly to maintain (Buschmann 2011). Responsible leaders therefore need to better understand the issue of technical debt and develop plans for remediating or managing it more effectively.

What can be done about Technical Debt?

There is no easy solution for managing technical debt (Alves et al. 2016). It is not always convenient to deal with and is a subject that everyone would prefer to ignore. Nevertheless, it must be effectively and holistically managed since it touches most every aspect of IT work and how well technology runs in an organization (Entertainment Close Up 2018). The focus group recommended six broad strategies that will help managers better cope with their technical debt.

1. **Measure your technical debt.** Measurement creates awareness of the nature and extent of the problem and will make all other next steps easier (Buchanan et al. 2017). There are some tools available to assess the structural quality of applications (e.g., the Consortium for Software Quality’s Automated Technical Debt Measure) and where they are available, the focus group recommends using them. However, these do not address many crucial aspects of technical debt. Because of its complexity and scope, developing and collecting technical debt metrics should be the responsibility of someone in IT who can do the necessary work to ensure that the measures used capture all its dimensions and at the right level. In addition to the traditional software metrics (e.g., defects), the group suggested tracking the time spent to integrate new technology or upgrade existing ones, tracking maintenance time to add new functionality, monitoring user experience, and monitoring other non-functional aspects of systems such as performance and reliability. Gathering metrics is only the first step however, as these will then need to be analyzed as to their root causes, and interpreted, prioritized, and acted on.

2. **Partner with Risk and Security.** Metrics do not explain the "so what" question that business and the board of directors need answered (Raskino 2018). Company risk and security functions can help connect these metrics to real risks and vulnerabilities that can help answer
this question. Working together to connect technical debt directly to these can quantify the value of technical debt and also guide leaders in prioritizing which debts should be tackled/funded first. Members suggested creating heat maps showing risks/vulnerabilities associated with debt metrics in a red/yellow/green format to provide clear visuals of the degree of challenge involved. Building an aggregate risk framework can also provide a good idea of the key areas where technical debt contributes to risk. "This is a good way to help us speak to these issues," said a member.

3. Take a business point of view. Ideally, managing technical debt should be a joint responsibility with the business, said the focus group. Any efforts to develop a plan to remediate technical debt must be based on both the business context and the technical environment in which the debt occurs (Ampatzoglou et al. 2015). The group pointed out that not all technical debt needs to be dealt with. "This should be a decision based on the risks, costs, and impacts involved so that all parties can have meaningful conversations and make effective decisions about it," said a member. First, business must be educated on the nature of technical debt through conversations that connect it to their needs in the business. Unless technical debt is linked to business needs, discussion about it can quickly slide into technical considerations like design and code, and miss its business impact (Raskino 2018). One way to do this if for IT to adopt a "partner through to value creation model" rather than simply focusing on delivery of new functionality. "If we do this it will change our discussions with the business," said a member. To reinforce the importance of managing technical debt, some of the focus group companies have added technical debt reduction to business unit leaders' performance scorecards. "We have some business leaders who aren't good citizens. This is a way of holding them accountable for their decisions," explained a member.

4. Manage technical debt explicitly. Once technical debt is more visible, it needs to be managed and monitored (Buchanan et al. 2017, Tom et al. 2018). A systematic approach is needed that works at multiple levels from organizational advocacy, to business leadership, architecture, and project teams (Tom et al. 2018, Martini et al. 2015 and Wisner et al. 2018). This will involve creating an architectural roadmap, governance, and standards to prevent future debt. Where "waivers" from these plans are essential, they must also include plans and funding to remediate them, said the group. Metrics, business priorities, and risk and security vulnerabilities should guide plans to remediate existing technical debt, and some remediation should always be "part of the deal" when new development requires changes to existing technology (Dawson and Cappucio 2017). "Right now, we do remediation on the fly," said a member. "We need to ensure that we always have a "remediation swim lane" with a budget for every new project." Finally, IT leaders must seek to ensure that sprints and
other phases of development have reasonable timelines since time pressure is a leading cause of technical debt.

5. Adopt tools wherever possible. Although only a few tools explicitly identify technical debt, there are many other tools and practices that can be very helpful in managing it effectively. For example, IT Asset Management disciplines can make sure that organizations are aware of and are actively managing the full life cycle risks and costs of all technologies in their environment. Network discovery tools can be used to locate and identify all technical assets (Krutchen et al. 2012). Common management tools and processes to manage IT work can minimize the risk of any project not following a roadmap or standards (Schwartz 2018). Most focus group members already have some form of asset management as well as plans to rationalize and simplify diverse technologies. They noted that business leaders are often reluctant to standardize their particular technologies and suggested that while not a perfect strategy, even reducing nine different types of a technology to three is a step in the right direction. Finally, automated testing tools are a must because they speed up the complex types of testing that are continually required in today’s constantly changing technology environment.

6. Develop a funding model. Clearly, managing technical debt effectively requires resources and this means allocating a reasonable budget for it. The focus group noted that business leaders understand the need for budgeting for vehicle and space maintenance, but not for technical debt. "Because it’s hidden, the business doesn’t understand it. We really need to make managing debt part of our budget and make it visible." Another added, "this should be like condo fees – just a regular part of our budget". A third said, "if we are aligned on outcome, that is, if we all understand if IT runs well, the company will run well and we will be much more able to innovate, then funding should follow."

Conclusion
This paper has explored the nature and causes of technical debt in organizations and shown that it is a much broader concept than is commonly believed. Technical debt is at the root of many of the risks and vulnerabilities companies face in a world where everything is connected to everything else. It is also a huge anchor holding back rapid delivery, agile change, and innovation. Senior executives recognize these problems but to date, have done little to rectify them. IT leaders, lacking adequate resources, have also failed to make it a priority. Instead, it has been left to be handled on a piecemeal basis by the Architecture group and individual teams as best they can. What missing is getting the buy-in and support of business leaders and developing
a collaborative plan for dealing with it. IT’s first priority therefore is to find ways to measure technical debt in all its forms and to articulate its risks in ways that the business can understand. Its second is to develop a plan to reduce this debt that both business and IT can agree on and then to fund it. Awareness is important but ongoing action is what is going to reduce and manage this problem. Coping with technical debt is not a quick fix, but requires a mindset shift that recognizes that just like buildings and cars, technology needs regular maintenance. Organizations can wait until a system breaks down or disaster strikes, but as with structures and vehicles, it is far more prudent to undertake proactive maintenance, adhere to building codes, and follow the advice of professionals.

References


- Buschmann, F. To pay or not to pay technical debt, IEEE Software November/December 2011.

• Dawson, P. and D. Cappucio. Follow these five simple steps to rationalize and renovate your IT infrastructure, Gartner Research G00337818, 7 August 2017.


• First Mark. There are 3 main types of technical debt, https://hackernoon.com, January 25, 2018.

• Haight, C. and R. Rosenstein. What to tell your CIO about I&O technical debt, Gartner Research G00239144, 4 March 2104.


• Kyte, A., L. Mangi, S. Ven Der Zijden. A primer on technical debt, Gartner Research G00307777, 24 April 2018.


• Raskino, M. 10 questions CIO nonexecutives should advise boards to ask, Gartner Research G00347796, 16 February 2018.


• Winser, R., R. Glair, and M. Jaggers. Infrastructure resilience debt: pay now or pay later but manage your debt, Gartner Research G00348507, 13 April 2018.
Concept
The purpose is to bring senior IT managers together to examine topics that are of critical concern to them and their organizations. Via the IT Forum (www.itmgmtforum.ca), members share experiences, learn from their peers, establish valuable networks, and develop practical strategies for creating, implementing, and managing IT solutions.

Recent Papers
- Managing Maintenance
- Managing IT Demand
- IT in 2015
- Business Intelligence
- Improving Customer Experience
- Mobile Technology
- Redefining IT
- Innovation with Technology
- Emerging Technology Management
- Developing a Data Strategy
- Developing a Cloud Strategy
- IT in 2020
- Transforming to DevOps
- Developing Thought Leaders in IT
- What to do about Shadow IT
- New World of IT Work
- Enhancing the Employee Experience with Technology

Participating Organizations
- Aviva Canada
- Bell Canada
- BMO Financial Group
- CIBC
- Empire Financial Group
- LCBO
- Ontario Teachers Pension Plan
- Reliance
- SCI Group
- Sun Life

Membership
Membership in the IT Forum is by invitation only. The annual fee is $3,000. Please direct inquiries to James McKeen at jmckeen@business.queensu.ca.