WHERE THE WORLD MEETS DEVOPS

Home » Blogs » Leadership Suite » Don't Let Cultural Debt Bring Down Your Business

Don't Let Cultural Debt Bring Down Your Business

BY CONTRIBUTOR ON SEPTEMBER 12, 2017 — 0 COMMENTS

Businesses today are faced with myriad challenges to innovate and optimize. In doing so, we often need to make decisions that borrow against the future. This can come in a tangible form, such as taking money from investors to fund growth, or in more inconspicuous forms, such as making tradeoffs in technical decisions. When we make such tradeoffs, we expect them to come at the cost of additional technical work in the future; however, these decisions can also borrow against an externality that is often overlooked: culture.

Recent Posts By contributor

- How to Ensure DevOps Success in a Distributed Network Environment
- Dissecting the Role of QA

Related Posts

- Paying D
- In DevOp Minimize

Join the DevOps Community

Stay up-to-date on DevOps, including breaking news, feature stories, ebooks, webinars, events and more...

Join now

Technical Debt

Technical debt occurs in the development of software when the "easy path" is taken to solve a problem rather than using the best overall solution. The concept implies that the time saved now will need to be paid back later (with interest!) in the form of rework.

Ideally, technical debt is incurred knowingly, i.e. a decision to take a shortcut is made deliberately to increase velocity, and it is flagged to be fixed before becoming a compound problem. In calling out technical debt as such, we can better analyze our decisions, considering their net present value and future opportunity costs. By documenting it, we gain better insight into the responsiveness and flexibility of the code base. Having a strategy around managing technical debt has therefore become an essential discipline in the management of software products.

Technical decisions also have an impact on the culture of a company, though, and this facet of technical management is often overlooked. "Cultural debt" can be taken on unknowingly, and the unmanageability that flows from it can be difficult to ascertain.

Cultural Debt

Cultural debt is when we make a technical decision that borrows against the culture of the organization. Such decisions can introduce divisions between teams, deteriorate communication or even weaken the effectiveness of leadership. These types of decisions need to be made regularly throughout the software development life cycle, and it is therefore critical that we begin to identify them, call them out and manage them accordingly. If the interest goes unpaid we can end up with a toxic environment that is difficult and onerous to manage—we risk a paradox wherein attempts to innovate can stifle progress.

A Closer Look

Just as with technical debt, a business ideally will have a strategy for managing cultural debt rather than allowing it to accumulate and bog down the business. Typically, decisions that involve cultural debt will impact on areas such as:

1. **Mission:** Ensuring connection to the company mission and that there is minimal variance in how the mission is understood by different team

2. **Values:** Effective and reliable transmission consistently and ensure that they are clear

4. **Visibility:** Providing the ability for everyone to see up, down and across their organization.

5. **Attitude:** Happiness of employees, their effectiveness and retention.

6. **Leadership:** Ability for leaders to govern effectively, and influence across departments

When making technical decisions, we need to think outside the box and consider how our decisions will affect these cultural dimensions of the business.

To reinforce the concept, let's look at some current issues that the modern enterprise is being confronted with:

1. **Domain-Specific Tooling** – As roles become increasingly specialized, a best-of-breed approach provides the tools needed to perform complex tasks. Unfortunately, tools create silos, and more tools means more fragmentation. Furthermore, in a world where tools are proxies for business processes, domain-specific tooling can cut off teams from processes, reducing visibility and impeding communication. This mean the business will need to devote cycles to meetings, emails and other noisy distractions just to pay down the interest and get the visibility and coordination needed for cross-functional effective leadership. Paying down the debt requires tool integration, and without an infrastructure to reduce the cost of integrating, domain specific tools can be a heavy debt to carry.

2. **Customization** – Often we see the introduction of new tooling without the requisite investment to customize them to the business. Tools such as ServiceNow or Salesforce are highly valuable for being specialized to their use cases, but they still require customization to be effective. For example, when customer data is stored in notes fields rather than using structures data captured in the tool, visibility and reporting is gradually reduced and employee frustration abounds. The decision to introduce tooling should be made with the understanding that customization will be required and must be maintained over time to keep pace with the organization's culture as it changes.

3. **Bimodal IT** – One strategy for enabling cutting edge technologies is the use of a "bimodal" structure with essentially two IT organizations that operate under different constraints. While bimodal IT may be valuable for encouragin̲ confusion of responsibilities, cross-team riv̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ structure remains, the deeper cultural divic̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ be. The decision to introduce new technolo̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ ̲ done deliberately, cautiously and with clea̲ ̲ ̲ ̲ ̲

Join the DevOps Community

Stay up-to-date on DevOps, including breaking news, feature stories, ebooks, webinars, events and more...

Join now

to deliver on the promises: e.g. a language such as Clojure or Kotlin really will simplify development and increase velocity, or a new delivery tool does indeed remove a key bottleneck. Adopting such a technology invariably creates some cultural debt—developers unfamiliar with the new language may feel alienated, new silos will form around knowledge and expertise and tribes could be forcibly broken apart. Technology decisions that ignore cultural debt will involve more cycles and greater risk, while those that recognize it will implement new technologies with those issues in mind, providing the cross-training and collaboration needed to ensure success.

5. **DevOps** – The dirty secret about DevOps is that it involves the introduction of a number of technologies and technical practices that carry a large amount of cultural debt. This can actually tear an organization apart if that debt is not managed closely, and it is therefore why DevOps—despite being closely associated with technical practices such as automation, infrastructure-as-code and CI/CD—is regarded as primarily a cultural movement. In the enterprise, DevOps technologies threaten roles, responsibilities and livelihoods. The cultural debt actually does outweigh the benefits, and DevOps transformations need to focus on cultural issues first before attempting to implement the associated technologies. This is common sense that we learned from observing lean manufacturing transformations in the nineties, which involved a similar change situation where technical practices became a function of culture.

Cultural debt, like technical debt, can be hard to detect. If your business does not have a strategy for paying off the debt, or at least one for paying down the interest, the result will be a heavy load of cultural issues weighing down decisions and processes.

Key Measures for Managing Cultural Debt

Understand your stakeholders: In an agile world it is easy to overlook stakeholders. We can learn from the disciplines of business analysis and project management, though, where stakeholder analysis is a core componency and key practice in work management. It is not possible to do a thorough breakdown of stakeholders for every technical decision, but an actively maintained stakeholder matrix will increase your mindfulness and can be referred to for a free sanity check when needed.

Understand your culture: Your organization is filled with cultural artifacts—if you are not aware of them and taking the time to manage them, you will not be able to assess cultural impacts. These artifacts can include mission and value statements, organizational and tribal structures, meeting cadences and compositions, control systems, routines, rituals, [...] artifacts and ensure that they match intention a[...]

Obtain buy-in: Getting buy-in means giving peo[...] on people will not only result in a requirements [...]

they have negative consequences.

Embrace the 'Three Ways': We can borrow from the DevOps movement the Three Ways, which provide a solid foundation for undertaking any technical work: Systems thinking encourages considering how decisions impact the whole system, not just the local area; feedback loops allow continuous feedback and communication about impacts; and continual experimentation and learning ensures that the organization is adaptive and resilient. Leadership that embodies the Three Ways is issued debt on very good terms.

Conclusion

Peter Drucker is quoted as saying that "Culture eats strategy for breakfast," and the examples we've seen here highlight this. The best technology in the world cannot compensate for a broken culture, and a broken culture can be the result of a succession of ill-considered technology decisions. Understanding cultural debt is a key differentiator in being able to leverage Conway's Law rather than be trapped by it. A key aspect of managing technology going forward will therefore be to identify and control this cultural debt as it is being created, and ensure that the organization is able to function with full-capacity engagement.

About the Author / John Rauser

John Rauser is the IT Manager of Tasktop Technologies, a global enterprise software company. As part of his role, he is responsible for specifying, managing and administrating software lifecycle tools, witnessing first-hand the impact they have on users and the business on a broader scale. He also serves as VP Operations at the board of the Project Management Institute – Canadian West Coast Chapter, providing leadership and expertise on technology issues. He has deep passion for discussing the business impacts of technology and analyzing strategies for IT management.

Containers are popular for good reason. They let developers bundle all the necessary components of an application into one package that can easily be shared and distributed. This approach

FILED UNDER: BLOGS, LEADERSHIP SUITE
TAGGED WITH: AGILE, CULTURAL DEBT, CULTURE, DEVOPS, STRATEGY, TECHNICAL DEBT

More

« Electric Cloud and the Deployment Dashboard

Liberty Mutual: Taking an Organic Approach to DevOps »

0 Comments   DevOps.com   1 Login

♡ Recommend 1   ↱ Share   Sort by Best

Start the discussion…

LOG IN WITH   OR SIGN UP WITH DISQUS ?

Name

Be the first to comment.

✉ Subscribe   Ⓓ Add Disqus to your siteAdd DisqusAdd   🔒 Privacy

NEWSLETTER SIGN-UP

Join the DevOps Community

Stay up-to-date on DevOps, including breaking news, feature stories, ebooks, webinars, events and more…

Join now

UPCOMING WEBINARS

THU
10

Winning at Culture: The Keys to the Successful DevOps Organization

Thu, May 10, 1:00 pm - 2:00 pm EDT

TUE
15

Cloud Migration: Solving for Growth with Containers & DevOps on AWS

Tue, May 15, 1:00 pm - 2:00 pm EDT

WED
16

Bridging The Gap Between Dev and Ops

Wed, May 16, 1:00 pm - 2:00 pm EDT

Past Webinars   More Webinars

DOWNLOAD FREE EBOOK

6 WAYS TO REDUCE IT ALERT NOISE

Join the DevOps Community

Stay up-to-date on DevOps, including breaking news, feature stories, ebooks, webinars, events and more...

Join now

🔲 DEVOPS CHAT

[DOES London 2018 Preview w/ Mirco Herring, DevOps for the Modern Enterprise](#)

[Anything You Can Do With DevOps, You Can Do In A Mainframe, Chris O'Malley, Compuware](#)

[Low-Code Is Coming To An App Near You, Appian World Preview](#)

[Microservices on Google Cloud Update with Tim Hockin](#)

[DevOps Chat: Interop China Features DevOps & DevSecOps](#)

PAST WEBINARS

Join the DevOps Community

Stay up-to-date on DevOps, including breaking news, feature stories, ebooks, webinars, events and more...

Join now

DevOps.com

- Home
- Business Directory
- About DevOps.com
- Media Kit
- Sponsor Info
- Copyright
- TOS
- Privacy Policy

- Search DevOps…

-
-
-

  - ○
  - ○
  - ○
  - ○
  - ○
  - ○
  - ○
  - ○
-

Join the DevOps Community

Stay up-to-date on DevOps, including breaking news, feature stories, ebooks, webinars, events and more…

Join now

-
-
-
-
-
-
-

  -
  -
  -
  -
  -
  -

Join the DevOps Community

Stay up-to-date on DevOps, including breaking news, feature stories, ebooks, webinars, events and more...

Join now