BAND-AID OR WELLNESS PROGRAM?

AGILEP●INT

TABLE OF CONTENTS

BAND-AID OR WELLNESS PROGRAM?

Most companies are attracted by the idea of transformation—be it systems, process, digital or organizational. Transformation means different things to different people. There's a wide range of topics: program initiation and governance, executive engagement, process architecture, project-level business engagement, embedding customer-centricity and technology implementation.

We prefer to see transformation as a journey rather a destination. Think of it as an organizational "Wellness" program. The overall trajectory and long range objectives are usually easy to identify, as are the next few steps. But what's just around the corner and what forks will you find on the road ahead?
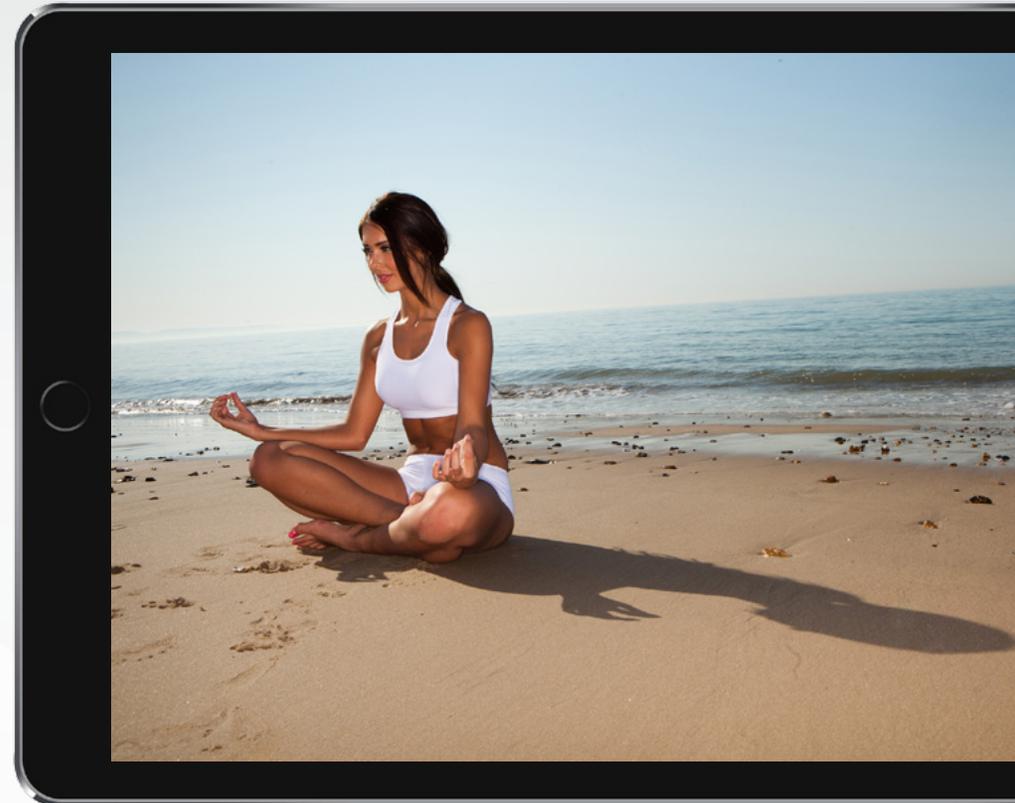
OR IS IT MORE LIKE ▲

Without the ability to continuously and easily evolve how your operations deliver value, you are destined to deliver yet another "Band-Aid" style project. Band-Aids typically focus on just one function or department, but ignore the bigger picture.

For true transformation, you need the ability to easily reconfigure your business operations—leveraging your existing strengths and combining that with the best of what's available today.

"AT THE TIME OF **PURCHASE**, EACH APPLICATION MADE SENSE AND WAS JUSTIFIED ON THE BASIS OF ITS **COSTS** AND **BENEFITS** DELIVERED. "

SO ARE YOU INVESTING IN TECHNICAL DEBT?

Every organization has invested in a set of software products and technologies over the last 20 years. At the time of purchase, each application made sense and was justified on the basis of its costs and benefits delivered. Generally, each investment focused on the needs of one or more functional areas within the business:

AGILEP●INT

"AT THE SAME TIME, THESE **MONOLITHIC** SYSTEMS REDUCED THE COSTS OF IT, AND AS A **RESULT**, THEY WERE EXTENDED INTO ALL SORTS OF OTHER **PARTS** OF THE **BUSINESS**."

## ✔ ERP SYSTEMS SUPPLIED A TRANSACTIONAL BACKBONE.

Whether it is SAP, Oracle Financials, Microsoft CRM or some other enterprise application; they have since become the system-of-record for finance, business operations or the call center. What happened along the way was that the organization then spent a fortune customizing the out-of-the-box functionality to more closely match the needs of the firm. At the same time, these monolithic systems reduced the costs of IT, and as a result, they were extended into all sorts of other parts of the business. The customization got more and more complex. But each time a new version upgrade is needed, you now find you have to rebuild that customization again—at great cost with little in terms of new value.

"HERE YOUR COLLEAGUES LEVERAGE THE EASE OF USE AND ACCESSIBILITY OF MICROSOFT'S FILE STORAGE AND TO-DO LISTS"

✔ OTHER APPLICATIONS SUCH AS SHAREPOINT HAVE TAKEN HOLD IN THE KNOWLEDGE WORK DOMAINS.

Here your colleagues leverage the ease of use and accessibility of Microsoft's file storage and to-do lists. But the processes of these high value workers still reside largely in the heads of the employees, or they are tracked elsewhere. Just think about how many Excel spreadsheets or Access databases are used to track work inside your organization; even a medium sized business has hundreds if not many thousands. All of these spreadsheets, databases and SharePoint lists represent little applications that hold you back.
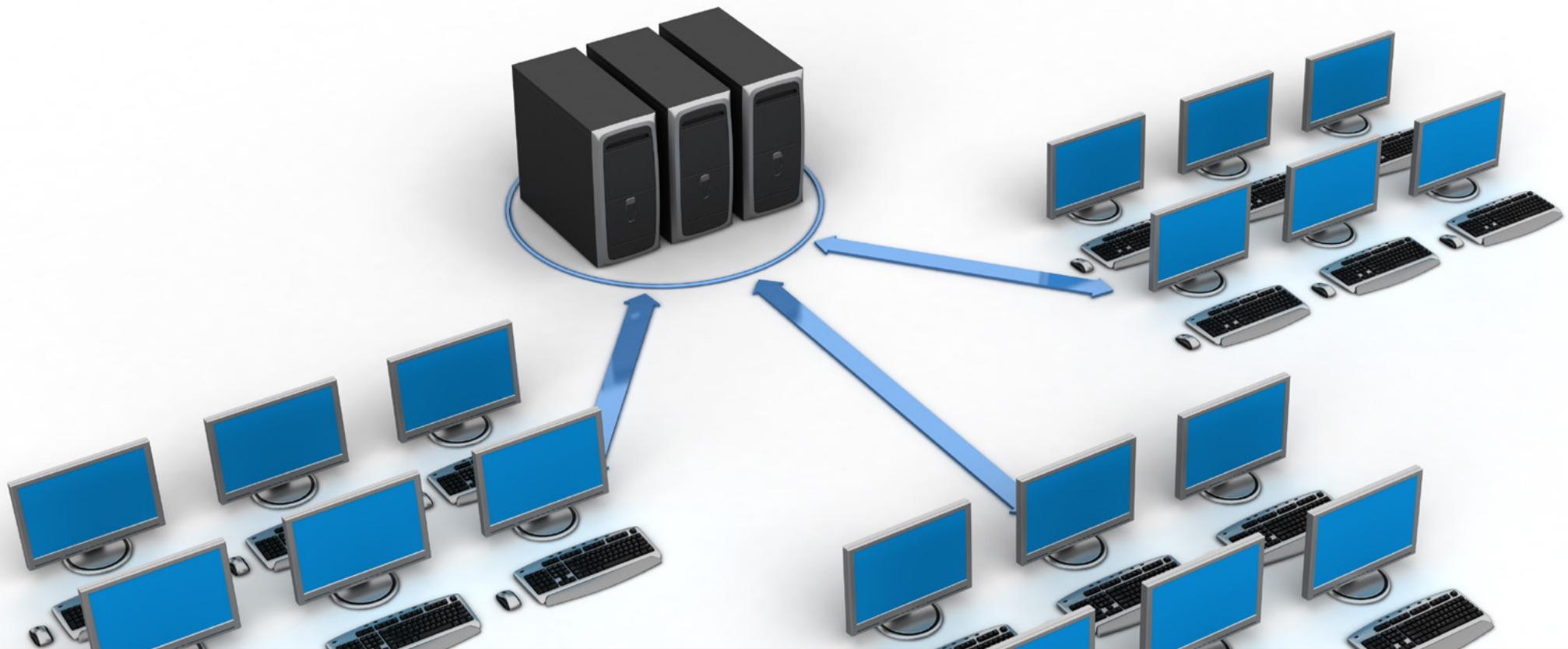
"HOW DO YOU CREATE REAL TRACEABILITY BETWEEN WHAT'S HAPPENING IN THE CLOUD AND THE ACTIONS OF EMPLOYEES AND PARTNERS POST SALE."

✔ IN THE SALES FUNCTION, THEY'VE BEEN USING SALESFORCE TO TRACK OPPORTUNITIES.

But that has created real challenges when it comes to connecting the results of those sales efforts back to the current system inside the enterprise. This is particularly a problem for firms who operate in highly regulated industries. How do you create real traceability between what's happening in the cloud and the actions of employees and partners post sale. And as other specialist cloud applications are deployed, the challenges of spanning the firewall just get bigger.

At the same time, competition marches on – new regulations emerge; new products are needed; a process reengineering initiative kicks off; employees, customers and partners – they are all clamoring for enhancements to existing systems. Then there are upgrades needed, often at great cost, to avoid an end-of-life scenario for an operating system, browser or database component. So any new application development is completely constrained by technical debt. And over time, it just gets worse – the debt interest compounds and the debt grows. The cost of development is just the tip of the iceberg.

BAND-AIDS COMPOUNDED BY BANDAGES

So on the inside, you've built point-to-point integrations between these applications to solve this problem or that; but in the end, it's got so complicated that nobody really knows for sure what impacts any changes might have. All of these things contribute to the constraints placed on your organization going forward. Collectively, all these links are more like bandages rather than Band-Aids. They have let one system talk to another, yet they have tied the organization in knots – slowing it down to the point that more agile and nimble competitors are busy stealing the future. For every step forward, you have to first take three steps backward.

YOU ARE IN THE SAME BOAT AS MANY OTHERS

And if it's any consolation, your organization is probably like most others. Every business is struggling to modernize its systems infrastructure.
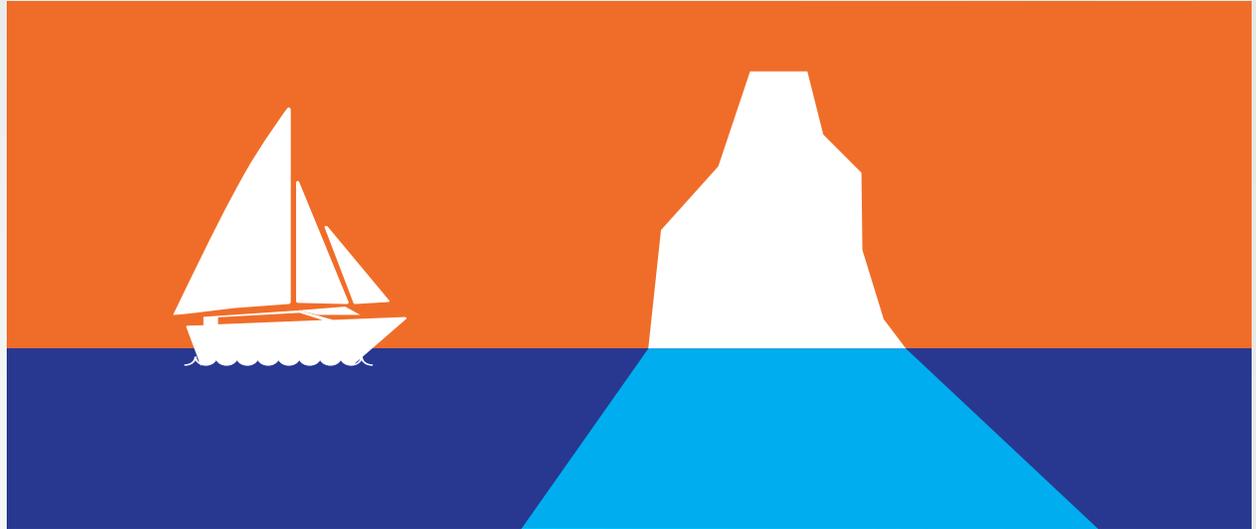
THE STARK REALITY

The stark reality is that every organization has a very messy diagram like this with arrows and lines connecting everything to everything else. Even the smallest change can have massive implications.
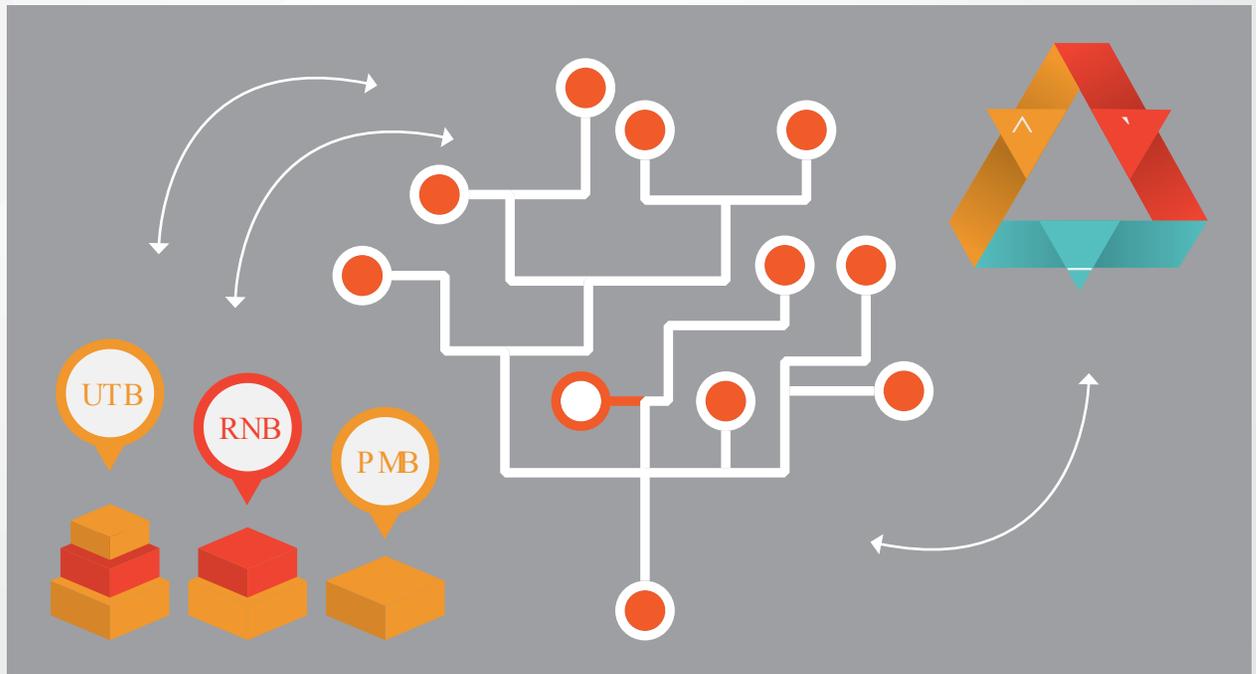
THEY ARE ALL WONDERING:

1

How can we deliver better customer outcomes and experiences (the value side of the equation) to scalable and efficient back office operations within the organization? How do we democratize the ability to change and evolve how we deliver value?

Or

2

Haven't we been here before? What did I do wrong when we purchased the last game-changing technology set? We hired the right consultants who developed new applications integrating at the API level of these big applications. But still we are considering throwing it out because we cannot change it fast enough.

3

Should we develop new applications in the cloud, with its promise of cheap elastic compute resources? But then, how can these applications leverage the resources we already have inside the firewall? So should we move some of our existing applications into the cloud infrastructure?

THEY ARE ALL WONDERING:

4

But how can we avoid putting all of our eggs in one basket—getting locked in to yet another big name vendor that's promised to do the right thing in the future? Is it really in their interests to deliver the innovation needed to get us over the plateau of this challenge? Will they disrupt themselves?

Or

5

How will we make these applications ready for an increasingly mobile world? How can we secure the combination of robust systems-of-record inside the firewall with modern system-of-engagement in the cloud?

For most CIOs and Enterprise Architects, the challenge has become how to meld these issues, using the next wave of technology to ensure an elegant and future proof solution.

IT'S TIME TO CREATE THE FOUNDATIONS FOR WELLNESS

If your intent is a "Wellness Program" rather than yet another set of "Band-Aids" and the associated technology debts that go with them, then you have some challenges to overcome. That's especially true when you factor in the promise of the cloud—with its ability to deliver great service experiences to your customers and employees alike, on a mobile device nearby. Your challenges include:

SALESFORCE, SAP, SHAREPOINT—THEY ALL LIVE IN DIFFERENT REALMS AND DON'T TALK TO EACH OTHER TOO WELL.

Therefore, moving our existing infrastructure to a virtual machine in the cloud doesn't make the problem go away. We are still left with the technical debt. The modifications made to our in-house SharePoint system are not transferable to the cloud-based version. And it's just not practicable to put our core ERP system into a cloud container. If we want anything other than the vanilla out-of-the-box opportunity tracking, then the Salesforce programming language still requires some heavy lifting at the technology level. Clearly, an approach dedicated to one or other of these environments just won't work.

For example, in the highly regulated business climate of pharmaceuticals there are real challenges associated with monitoring sales in Salesforce and then linking that to SAP for fulfillment. Consider then bringing the results of all that together in a business planning scenario supported by SharePoint. Relying on one of these systems to act as the "hub" will just increase the technical debt. We need something that sits above it all.
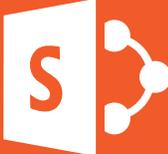
ENCAPSULATE APPLICATIONS INTO A SET OF GRANULAR AND REUSABLE INTERFACES (MICROSERVICES).

Microservices allow you to break up monolithic applications and present discrete elements of functionality separately. However, rather than a purely programmatic interface—where each is application interface is hard-coded—you really need to manage those interfaces through metadata-driven mechanisms. This is important since, should that application change, you only want to have to refactor the interface once instead of in every place it's used in every process of every application.
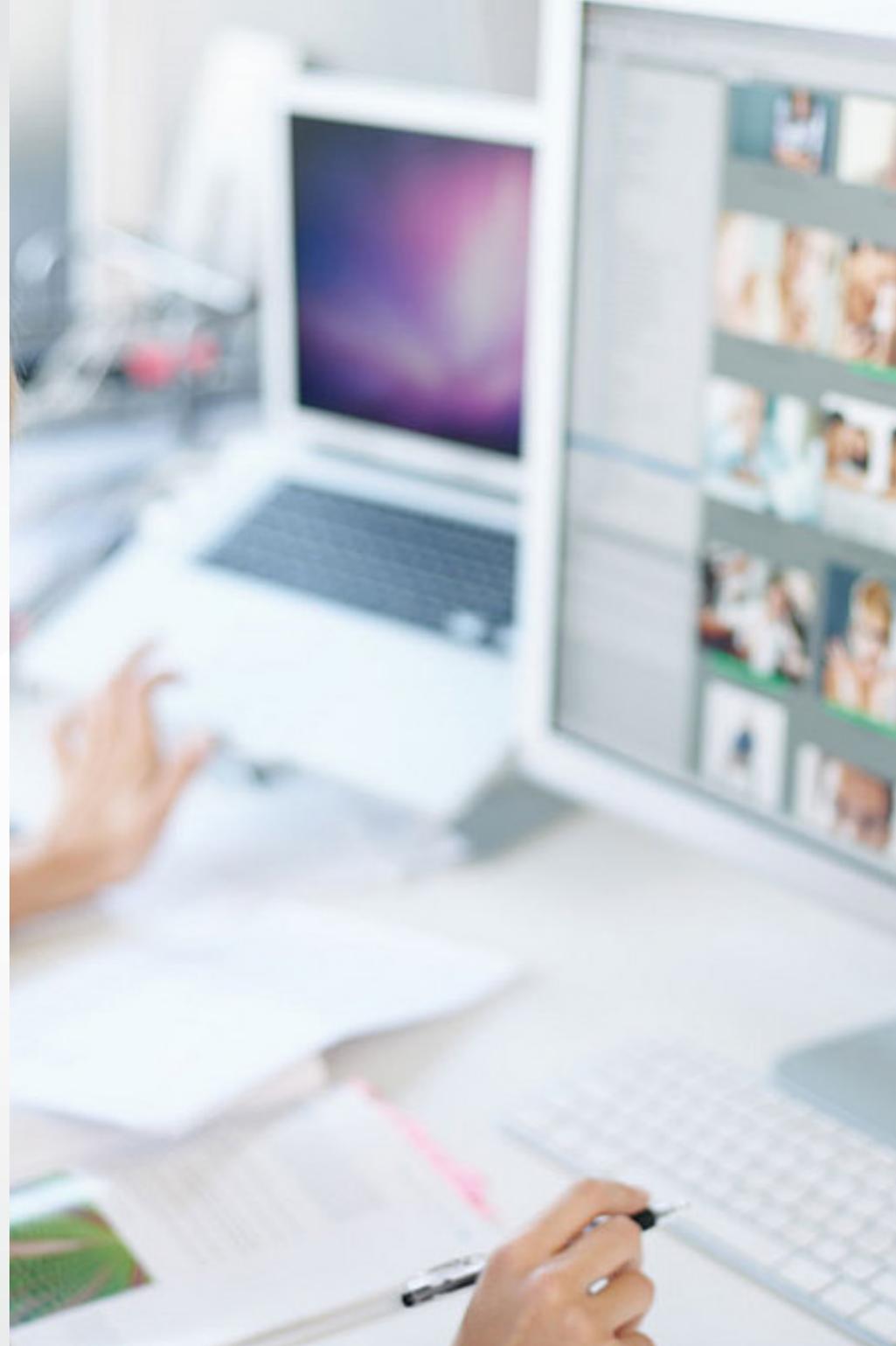
Taking our pharmaceuticals example, as the "Opportunity" record in Salesforce nears the "Closed-Won" state, it's important to ensure that the appropriate compliance carries over to business operations. So while we will no doubt leverage the APIs to SAP and Salesforce, the way in which we interact with these should enable reuse of the state information in Salesforce. Moreover, when the SAP or Salesforce application rolls on to the next version, we would only want to revalidate that connection once.
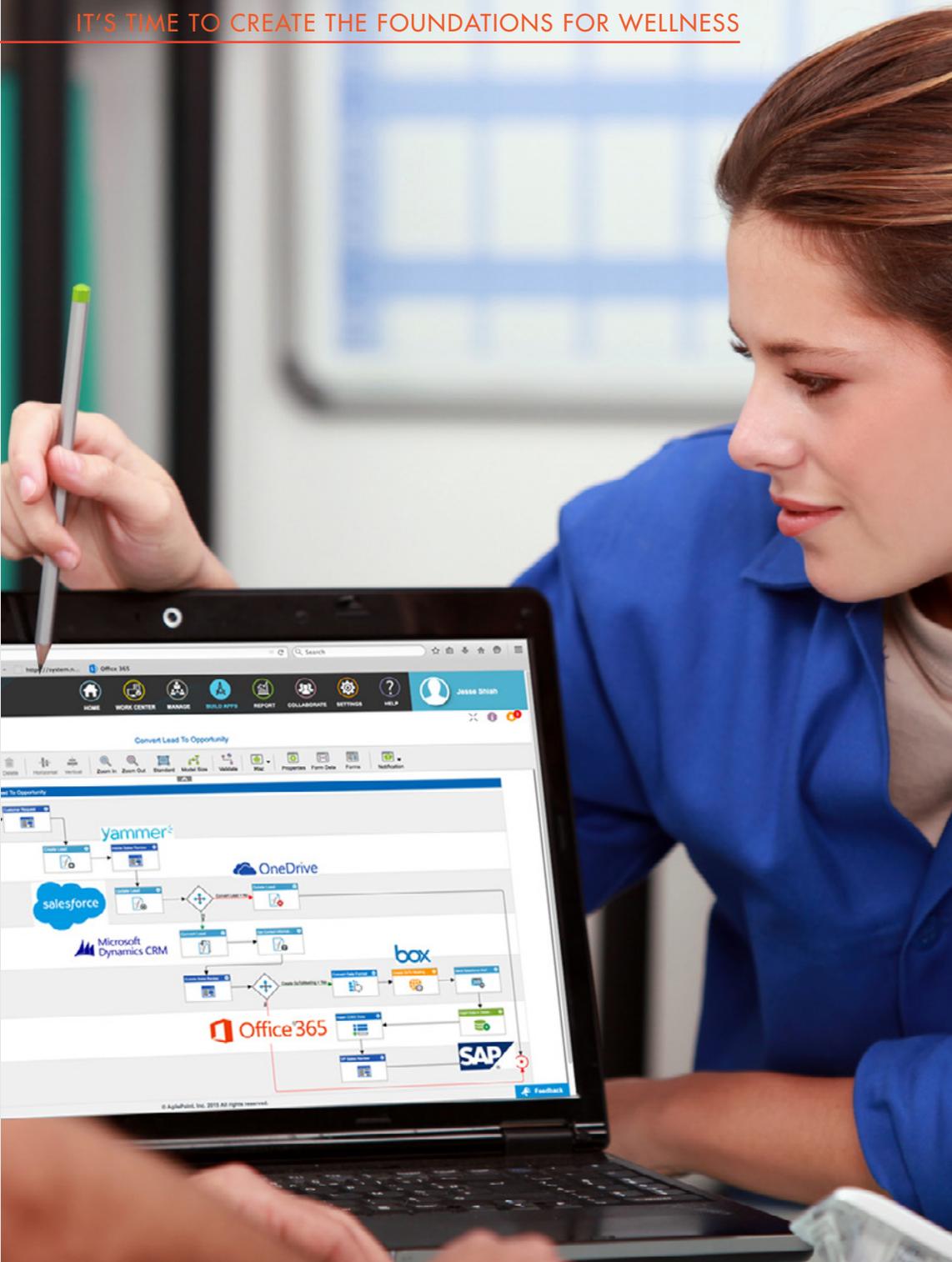
DELIVER APPLICATION FUNCTIONALITY IN A TRULY MODEL-DRIVEN WAY.

To avoid the technical debt—below the waterline of the iceberg—there are some subtle nuances to overcome in leveraging these metadata-driven interfaces. It's not a question of pushing the programming envelope by more quickly assembling snippets of code for execution. It's about assembling the right models that represent elements of business functionality, and then binding these business components together at run time to reflect the business intent, or the needs of the customer. That way, when your needs inevitably change, you can tweak the models above the waterline, rather than employing a deep sea diver (or picking up the phone to the software developer).

Let's assume that your ERP system sits behind the firewall, and the compliance team now wants to introduce an intermediate state called, for arguments sake, "Stage 2". We would want ensure that all our existing interfaces to Salesforce and SAP would still work, but when we get to Stage2, we've realized that we need to ensure a compliance check of our engagement with the client to date. We now want to kick off a review case in the SharePoint environment that the compliance team uses. Today, that's on premise too, but tomorrow it's likely that the organization will want move this to the cloud.

RETHINK YOU APPROACH TO PROCESS DESIGN.

You still need to thread all your business components together meaningfully to deliver value to a customer, or an employee. So orchestrating these components together using a BPM tool is the right way to go; but that's more than just a series of administrative workflows within a single cloud service or behind the firewall in an on-premises server. It means straddling the two environments, with components consuming and integrating the microservices of stand-alone applications in each domain. Moreover, it means developing re-usable components at the business level, with rule/policy configuration such that they deliver the desired functionality for different use cases.

So rather than a process design for this aspect of compliance in the sales process driven out of Salesforce, we want a generic business component that hoovers up the relevant context from the over-arching processes. That means taking a snapshot of the data at that point in time and delivering that to the compliance team. As you can see, have to rethink your approach to process architecture—rather than thinking of one end-to-end process for sales, with arcane code embedded in it to cover every possible path, it now becomes a set of process driven components brought together in near real time to reflect the needs of the scenario. Moreover, these components will need to seamlessly interweave from the cloud, to on premise, back to the cloud and back again on premise, ad infinitum. For many, it's a new paradigm of composite applications organized around business capabilities.

YOUR USERS TO DO THIS FOR THEMSELVES.

Yes, IT has a role, but once you componentize those business applications, embedding the metadata to enable effective reuse without code, the role becomes more about facilitation rather than control-centric policeman. Sure, you will still want governance around key changes, but the key point to grasp is that the overall cost of ownership goes up exponentially the further you get from the business need. Or putting that the other way around, the more you can help the business to self-serve, the lower the technical debt and as a result, the cost of ownership goes down by an order of magnitude.

TECHNOLOGY TO THE RESCUE!

Not surprisingly, given the articulation of these challenges, there is a solution. AgilePoint NX is what we call a Responsive Application Platform (RAP). AgilePoint NX allows you to create business apps and workflows without writing programming code. The product's genesis is from enterprise BPMS (Business Process Management Suites) but goes well beyond that genre. AgilePoint NX combines a simple, model-driven approach to app design with plain-language configuration of components and on-the-fly adaption to changing business conditions. AgilePoint NX incorporates:

1

RESPONSIVE APP DESIGNER— NO CODING REQUIRED

AgilePoint NX App Designer enables composition of applications from pre-defined components using drag and drop. These components integrate industry standard systems, such as SharePoint, Sales force, and SAP.

2

RESPONSIVE DESIGN FOR ALL DEVICES

AgilePoint NX applications run on any device, but you can design device specific visualizations. You can selectively include or exclude fields and other page elements, or resize elements for different devices.

3

RESPONSIVE FORMS OF ANY COMPLEXITY.

Using the **AgilePoint NX** cloud-based form designer, you can create anything from simple forms with just a few fields and controls to complex, multi-tabbed forms. With twenty-three different control types you can also build in business logic into the form itself, dynamically restructuring the form based on user input.
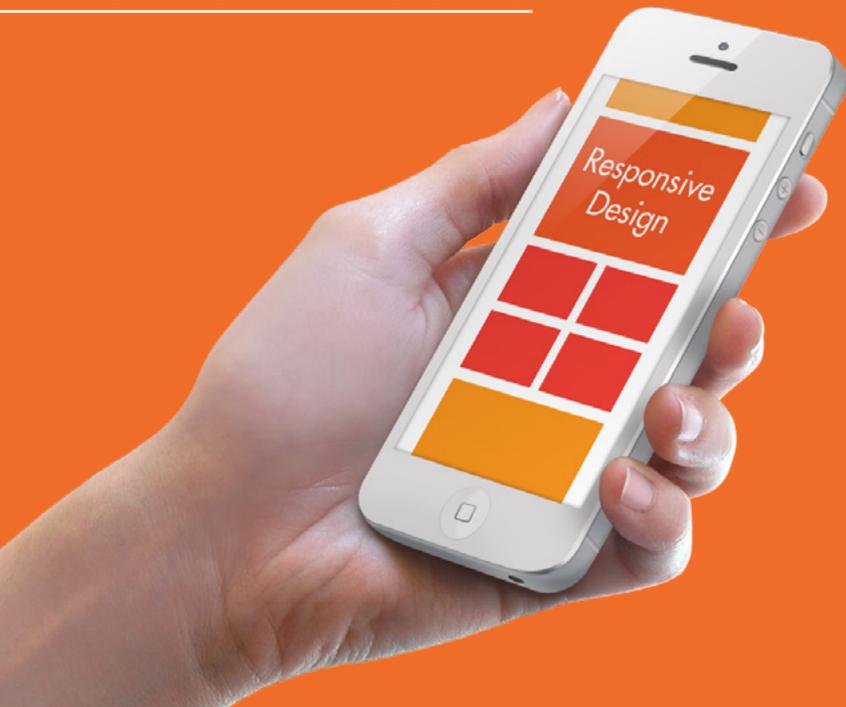
4

RESPONSIVE DATA INTEGRATION

With **AgilePoint NX**, it's easy to connect one or more primary data sources to your processes and the apps using the field mapping utility, which makes it easy to get the right data in the right places. Moreover, you get to free up data that resides in heritage systems and deliver it to the desktop and in a mobile friendly delivery app.
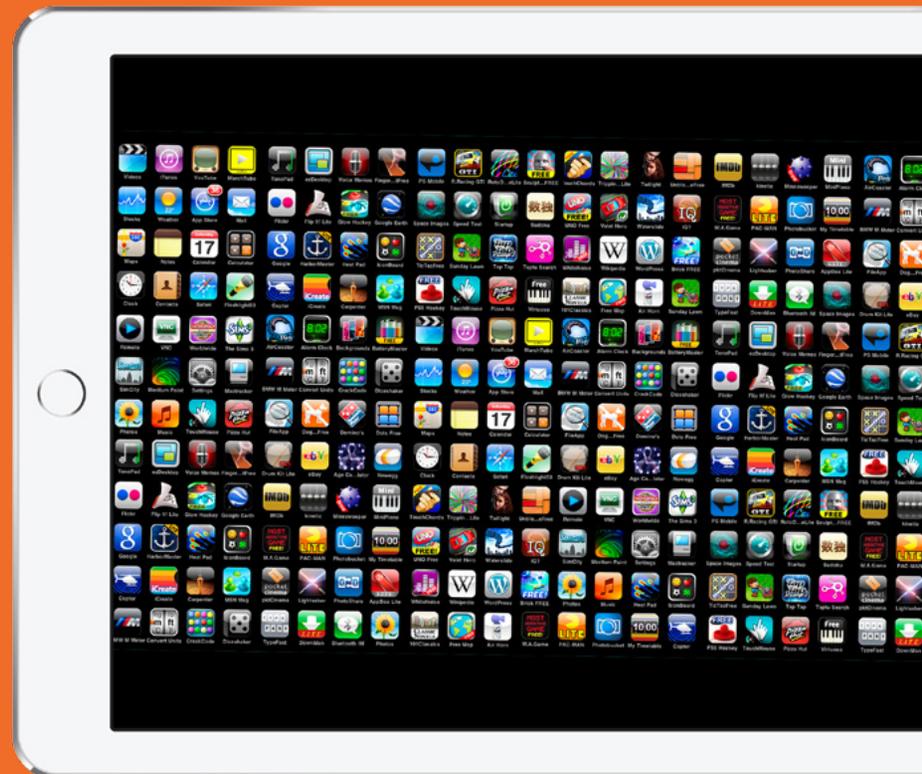
5

RESPONSIVE APP ADAPTION

AgilePoint NX allows you to create apps that can adapt on the fly to changing business conditions—you can even reconfigure individual components in an app while it's running. Or you can design an app so that components are reconfigured programmatically, based on feedback from other applications or devices.

6

RESPONSIVE APPLICATION SECURITY

Using **AgilePoint NX** Secure Link technology it becomes possible to link applications running in the cloud with those operating inside the firewall. Rather than opening up fresh security challenges, VPN tunneling between the process engines enable the passing of data and context back and forth between these domains.

WITH AGILEPOINT NX…

You can reconfigure your business operations on the fly. This revolutionary approach yields a new sort of application liquidity—loosely coupled components, on-the-fly reconfiguration, and in-flight execution of reconfigured process models. It's well suited to everything from simple SharePoint workflows to the most sophisticated Internet of Things (IoT) applications.

To find out more …

7

RESPONSIVE BUILD-ONCE TECHNOLOGY

AgilePoint NX allows you to build an app once and then create layers of configurations for other groups. This approach eliminates the need to create a variant of an app for each group that has different priorities or requirements. This ability to configure apps is a key component of scalability.

AGILEP●INT NX

THE RESPONSIVE APPLICATION PLATFORM

W W W . A G I L E P O I N T . C O M