

Blueprint Ep 7 Transcript

Matt Espley

Everyone. Thank you for joining today's episode of The Blueprint. I'm Matt Espley here with fellow CMAs Nate Weldon and Kenric Wong. Today's episode of The Blueprint focuses on helping you navigate one of the toughest platform decisions. I think we've run into, but very often run into, which is when to build, when to buy, how to manage the customizations responsibly.

Matt Espley

One of the most common strategic challenges, as I said, is we're platform teams to decide when to build custom applications, when to buy existing solutions, how far to customize ServiceNow is out of the box capabilities, and as consultants, we often have these conversations around can I do X, Y, or Z on the platform? The answer is usually yes, but should you is the big question, and the decisions that you make here ripple across the total cost of ownership.

Matt Espley

Time to value technical debt, and ultimately the long term sustainability of your platform. So in today's episode, we're trying to unpack when to leverage ServiceNow native capabilities versus custom apps, how to balance the customizations that you make with maintainability, understanding what is true cost of ownership, and how to consider true cost of ownership against the cost to do the work.

Matt Espley

Or, you know, get an out-of-box application and then maybe get into some real world examples or stories where some organizations went maybe too far down the customization rabbit hole and how they came back. So the first question, I guess, is really a general, talking discussion point. What do you guys how do you guys even consider framing these discussions and these decisions when we're talking about doing development versus, you know, going going out of box, like, what if what in your mind are the are the factors or considerations that that you start with?

Nate Weldon

So the very first thing I look at when somebody and it and it doesn't, it doesn't often start off as a buy versus build conversation. It starts off with a, hey, I'd like to do this thing on ServiceNow and it's up to us to enlighten that customer. Well, did you know that ServiceNow has a product that does this?

Nate Weldon

You know, way back when before there was App Engine? Well, I think when Scoped Apps first came out, you know ServiceNow position was if you think you can build it better be our guest. That has changed. Because, you know, there there, there are that tech that there is that technical debt. There is that, you know, I now own this code versus this is something I can make a phone call or go to support and just log a case for, you know, you own your custom code and that's that is the biggest consideration.

Nate Weldon

But, you know, when you're when you're, you know, extend that to a buy versus build conversation in HR is the

is probably the most common one we've been asked to to build is, you know, what did you know ServiceNow does this. And usually if the answer is yes, we do. You know, then we work with ServiceNow to, you know, help work with that account executive work with that customer, help them understand the full breadth of the HR product offering before any determination or any decision is made around, okay, we're just going to build you a simple case management application.

Nate Weldon

What you want to call it is your prerogative. And who you want to give it to is also you know, your decision, but know that.

Nate Weldon

A campaigns employee lifecycle and event manager hub all of those value added products that, you know, enable case management and sit on top of case management and knowledge management for that matter. You know, you're not going to have that. So is this a gateway to getting to ServiceNow product. Because you know, your H.R VPCs, you know, your your small organization in your VPCs are not used to operating out of a Q or A, you know, never operated as part of an HR call center.

Nate Weldon

You know, then you that you want to have that maturation conversation with the customer and ultimately it is a pathway to getting them into the ServiceNow product.

Kenric Wong

Yeah. And we I love how you segue into that because, it brings me back to one of our previous episodes, right? When we talked about value management. Right. We're looking at the value of the organization as a whole. So from a strategic standpoint where we're kind of laying out some of the roadmaps and the capabilities and milestones you just touch right on it.

Kenric Wong

Right. Is the the CSM is that customer service portion on your roadmap and where is that all aligned with our system. So when we get to that point. But right. So then you get like.

Nate Weldon

Why would anybody want to rebuild that external security model. I mean if, if, if external access is a requirement and I don't know if the architecture or the SKUs have changed. Right. But it used to be if you wanted that checkbox on your portal. Yeah. The same public you had on CSM. I'm not I'm not sure if that changed, but yeah.

Kenric Wong

And to your point, right. That's when that decision becomes like a build or or wait. Right. So you're thinking and it doesn't have that capability. So it is it is a segue way to maturity. But it gets them what they need for their specific milestone that they have to reach. You know, sometimes the decision is to go to the App Engine in the scope application, get them that, start that MVP as they move into that roadmap to mature that process.

Kenric Wong

So everything is up in the air, but it goes back to that value management. And you you said it too, right? It's the risk mitigation piece. Right. Is it risk averse risk transference. What portion of appetite are you at at this point? Where is it worth it to do your own custom code, where you are owning every risk now because you

have the support of yourself versus, you know, you're utilizing a capability that's there and then say, you know what?

Kenric Wong

We are moving that risk towards the platform because that's managed and, outward and supported by ServiceNow. Right. So I think that's a good point to do.

Matt Espley

Sorry, I was just going to add real quick. You know, I think the other thing that, you know, architects and CMAs should be asking is, you know, does a platform already solve this natively? What are you going to be missing out on if you customize, if you go another route. Right. And then also what business outcomes are we trying to solve here?

Matt Espley

Because sometimes I think ultimately customizations can be like a developers preference where they know we can. So we do it and we don't necessarily look at the long term picture and strategy of the platform. So a CMA or an architect should, should always be trying to turn the decision towards what are we achieving for the business and how can we do it in the most sustainable and, you know, cost effective manner, not just doing something but because we can.

Matt Espley

Right? Everything can be done. I mean, not everything, but most of the time the answer is, yeah, we can customize it, we can do it. And we have to make sure that we're that we're aligning what we're actually doing to the value. Like Kenrick was saying, not just because we can and because it would be cool and because it's it would be fun to build.

Nate Weldon

I want to finish my original thought. And one of the the good things about starting in a scoped app is the interface is the same. You're going to be in either, you know, the back end UI 16, you're going to be in a workspace, or you're going to be in portal.

Nate Weldon

But it's a stepping stone to getting that full filler group used to living inside of ServiceNow and, you know, groups that are not used to working out of a queue or working out of, you know, something that has prioritization, something that has service level agreements that are measured against it. And the that's all part of a crawl, walk, run approach.

Nate Weldon

So, you know, it's not necessarily a bad thing if they can't start off with the Ferrari of everything, maybe we need to start off with one closer to a Cadillac.

Kenric Wong

You know, I actually have a horror story to Matt's point on on when when I was a, customer, in the health care industry. And we needed to identify on car rotation because, you know, we're supporting clinicians and physicians. We need to be a 24 hour support, both our end user device computer folks and our help that and, you know, we didn't have the foresight to look ahead on even ServiceNow as a roadmap, let alone kind of what our strategic plan was at the organization level.

Kenric Wong

And we built our own custom on call application. And, you know, sure enough, it took weeks, maybe even months to get it stood up. And we had to identify, the crazy conversion for time. And ServiceNow used a very different kind of minus nine, minus ten. And you're looking at converting date time stamp 24 hours versus 12 hours.

Kenric Wong

And when it's all said that it was working, we gave them the capability. And next thing you know, you know, boom. Literally the next version we went to in ServiceNow, which was already on their roadmap, they actually had it all call capability came with ideas. And we looked at that and we were like, man, this is such a mess technical that we just got ourselves into.

Kenric Wong

And now through training that we just took and everybody's training, now we have to undo, go back to it, retrain everyone. I just I'm not against that was just not a very good use of funding, I could tell you that. Right. So it was a very bad point.

Nate Weldon

And that's a great point. The funding aspect of it now. But back then ServiceNow was not exactly as forthcoming with their roadmap. But, you know, even with even with us at that, that at that point in their, in their evolution, they had just gone public back then probably actually rotas came out before they went public. So, you know, it's just a completely different, completely different animal.

Nate Weldon

But we, we, you know, a previous organization, another partner that I worked for, you know, we we hired a bunch of developers to build a large app. And, you know, they weren't ServiceNow developers. They didn't have any ServiceNow experience. So after they, you know, and they we didn't we didn't force them to go through their CSA either.

Nate Weldon

The the fundamentals course. So we, we, they knew JavaScript, they knew how to develop, and they, you know, fumbled their way around the interface and the very, very first thing they built was an approval engine.

Nate Weldon

So, you know, and and, you know, when, when we came in and did a review and, you know, they were demoing it to everybody, it didn't look right. Right. You know, those of us that knew were like, wait, that's not how approvals normally look. And they informed us that they spent two months building one. Lesson learned, you know, when you're and it was also way earlier in that particular organization's maturation curve.

Nate Weldon

So, you know, we didn't have good development practices. We did, you know, clearly, we didn't have a good solution review practice, because one of the architects that didn't know ServiceNow would have very easily said, yeah, don't build that. That's already there to the point, you know, understanding what the platform offers. It's one of the one of the biggest and but but you know, when it comes to the platform and the

capabilities that are platform wide, sometimes you have to you have to connect some dots.

Nate Weldon

You have to say, okay, I've seen it do something similar here. Let me see if I can do like let me see if I can massage it to make my other very similar requirement work without either customizing or making a slight tweak.

Matt Espley

Yeah, that's a good point. I think sometimes it's, you know, it's probably good to call out it's not black and white, right. It's not use an out of box app or build something from the ground up completely. Right. Sometimes it's that middle ground where it's the out of box, but you don't quite have the the necessary features that you need for your business process.

Matt Espley

And you can tweak and customize, in a smart way that doesn't make your upgrades unmanageable, that are well documented and well thought out, and don't get in the way of future releases and other features on the platform.

Kenric Wong

Yeah, I think it kind of answers one of your questions. Right. Like how much do you customize it. How much do you decide that? Right. Isn't that one of the things that would be asked like, you know, you kind of have to mix in like you said, it's not black and white. It's not something you say, oh yeah, we go this out or we go this.

Kenric Wong

Right? It's more like, yeah, you know what? We don't have to customize so much because we can utilize this capability, but add on to what we need to make it work for the business. I think that's that, that, that, that answering that question that we can estimate. And I think that's a really good, good segue into other things that you guys.

Matt Espley

What would you say?

Nate Weldon

That's that's a good point, Kenric. I mean, sometimes

Nate Weldon

the requirement might need to be massaged into what it can do out of a box. Because if it if it if you can get close, maybe that's acceptable enough. Maybe that checks enough of the boxes for the business side of things to to say, you know what, we're not going to customize.

Nate Weldon

We're going to use this out of box functionality. But there they're all conversations that need to be had as part of that discovery design stage.

Matt Espley

Yeah. Sorry. That was going to be my question to you both is like for customers listening to this and they're trying to navigate this conversation with their stakeholders right now, what recommendations do you have to

help them convey properly? This, this, you know, custom versus out of box and what what tools can they show up with to help them navigate the conversation and drive it in the right direction?

Kenric Wong

I have some examples, in reality, of the client that I'm working with. So one of the components of processes that are required and these are actual required processes in their business world. One of the questions was, I have, you know, how much of this process can be modified to kind of fit certain needs in our system and how much of it has to be that strength?

Kenric Wong

And we found out that it really does have to be stricter, that it is congressionally sound. So it's more like, well, this process has been stymied not only by Congress, but, you know, this is a federally mandated process, which means there isn't much wiggle room within what we can and can't do. Right. But it doesn't neglect what Nate said earlier and what you said, earlier about what can still be utilized close enough within that box.

Kenric Wong

And then make sure we mixed in a blend of, you know, logical, common sense development of customization, but still utilize the platform. So it contains the capability to not only scale but, you know, still continue to be able to grow with the platform without having increased technical debt. And and that was one of the examples of asking where can we move the process versus.

Kenric Wong

Well, I mean, in order to go through Congress, change it, even if they agree like this is a better process, we can go up to Congress. That approval process is going to be months, days, you know, weeks, years, potentially writing, you know, business to having a time to wait for that long. So we have to think about other solutions.

Nate Weldon

Yeah, that's a good point to

Matt Espley

what is total cost of ownership? Yeah. And why is it important and how does a customer leverage it, understand it, and leverage it in order to make decisions?

Nate Weldon

So it is not just the licensing. It is not just, you know, your initial implementation project. It is the cost of your system administration team. It is the cost of your enhancement team. It is the cost of development efforts. It is the cost of third party scoped apps. So you when you when you look at the total cost of ownership, you know, from, from any system SAP, ServiceNow, workday, you know, it's not just how much you're giving that vendor.

Nate Weldon

It's it's everything you are. You are spending your company's money on, which sometimes can be a little bit hard to, to, to realize because there are sensitive topics such as salary and benefits. Right? It's not just a salary, it's a salary plus the benefits. I mean, we we we understand that regularly because our business is selling people's time.

Nate Weldon

In, in other industries that might not be so easy to or it might not be so readily for a first person's reaction is to say, okay, somebody makes X number of dollars a year, okay? Now you have to tack on benefits. Now you

have to tack on employment taxes. Now you have to tack on Social Security taxes.

Nate Weldon

And you know, in some organizations that range, I believe, is, you know, from 20 to 40%, depending on other benefits. And, and I've heard some organizations go up to 50 or 60% of the salary is what goes into that. So total cost of ownership is is very broad. But it's also very important to understand because that's how you should be prioritizing your your demands of the team.

Kenric Wong

I think it's you. You mentioned it right. I think it's it's it boils down to a discussion point, because what you mentioned is everything that covers TCO. But, you know, when you meet with your executives and you talk about what the question is, what are they truly tracking that they want to consider as a TCO? Right. Because that's what you're going to be baselining your system against to track it consistently because it marries back to your that plus your, your risk.

Kenric Wong

Right. Like it boils down to the point where, like you said, sometimes they include 60% of that and say, this is my total cost of ownership, right? Just so they can do the math. Some of the sensitive API, whatever data that they want to uncover, it's just interesting. There's a good conversation to have with executives.

Nate Weldon

But I also think looking at it like I do not like to start off with it in a TCO conversation, because because that's just what it cost. What is the potential benefit to the organization? Can we help increase revenue? Can we help decrease costs? Can we help streamline things? There's a whole other slew of conversation topics that go along side of TCO.

Nate Weldon

Yeah. But so often, you know, as consultants we start with TCO.

Matt Espley

They go together. Right. Like you're saying the what's your return on investment. What value we trying to get. Like how are we going to measure it and realize actual value, the return on the money that we're spending. But the cost of ownership is what's degrading that return on investment. Right. And you can't really understand what your ROI is going to be if you don't have your cost of ownership factored in there too.

Nate Weldon

Yeah. Or in some instances, the cost of entry is the barrier. Just getting the licenses could be prohibitive.

Kenric Wong

Yeah, yeah. Oh man. Exactly. Like because you can obtain all the licenses. And if you don't calculate the fact that you might not see return in two years because of it, that it's just one of those things, where is it worth it? It what what at what point do you calculate TCO into that factor? Or when do you start counting against it?

Kenric Wong

Because again, some of the lifts just like starting a restaurant, right? You, you know, when you build your own restaurant, you're using your own capital. You're, you're expecting to take at least a couple of years before you

start seeing any revenue generated for your organization. Right. And that's just the concept of it. If you're already starting in a negative, how you get yourself.

Matt Espley

And you could be in an opposite scenario to where I've seen, you know, there's a lot of value upfront. Yeah, right. Maybe it's a heavy lift to build something custom, but you see immediate efficiency gains. Right. But then you're looking at those technical debt risks down the road that ultimately materialize into the cost of ownership in terms of reworking, refactoring code.

Matt Espley

Yeah. And trying to pull in business processes that you didn't consider before. And now, like with your architecture, like for example, we have, a client in the, in the legal services space and, you know, they built a big custom application, for a certain part of their business with another partner. And we we've been working with them for a couple of years in helping to optimize and enhance.

Matt Espley

And I think in one case, it got to the point where it became very apparent that the entire application needed to be overhauled, basically, again, from the ground up. So now there that was a very unforeseen cost to own this application. And now we're we're in the position to where we got to start over. And that's something that I'm sure was never even a consideration down the road.

Matt Espley

Right. But I can definitely degrade your return on investment. Right. For the last two years, there may have been a pretty solid return on investment, and now we're looking at completely pulling the rug out from under it to redo it.

Nate Weldon

Yeah, yeah. The point, because I want to pause for a second. Our of caviar is able to speak, but, we do have some, some guests today. And if we want to give them the opportunity to ask any questions, how do you how would they, I don't think I see any option to raise my hand, but, that's probably because I'm a presenter.

Matt Espley

There is a Q&A session section where I believe people can just plug into a question in there, and it'll pop up on our end.

Nate Weldon

Perfect.

Matt Espley

Next question.

Nate Weldon

So for the live audience, if you do have any questions, please submit them in. There should be, a question mark box on your right hand side of the screen. If you have any questions, please ask this. This is an interactive, session, and we are here to answer your questions. So please ask if you'd like any more information on any of the topics we're talking about today.

Nate Weldon

Sorry about that. Had to do a little a little PSA, you know.

Matt Espley

Good. I should probably start with that in the beginning. Right.

Nate Weldon

Maybe.

Matt Espley

We talked about total cost of ownership, kind of understanding ROI. I think we kind of we implied a lot of governance conversation, but we haven't necessarily addressed it outright. What is as CMAs, how do you guys see our roles in helping to protect the platform from, you know, bad customization or, you know, hurdles down the road? And unfortunately.

Nate Weldon

I think it starts in two different it starts at the top of the the governance layer in the executive governance sessions, in should you buy versus build conversations. And then from the the bottom layer, the technical governance up, you know, that's where things start with, with you know, you've already made a decision to build an app.

Nate Weldon

How you construct that app, do you want just a bunch of JavaScript, or do you actually want someone to go into App Engine Studio and, you know, use table configurator, use UI builder, use those things? You know, my position is if it can be configured, always configure it no matter what, even if it's faster to just write the code, you're leaving the customer with less technical debt because somebody who understands how to just go in and click around and may not have years and years of development experience, you're at least leaving them with an application that they can go in and take ownership of and configure without having to learn a bunch of JavaScript and the and the glide APIs. It's two pronged in my opinion.

Kenric Wong

Yeah. I think as a CMA. Right? Matt? I think that's where we kind of connect the dots with the enterprise architecture of that organization in a sense, because they're going to look at that in a very, a more holistic approach from an organization standpoint. So they'll have all their applications ready to go, already aligned to certain services and business models.

Kenric Wong

And I think as a CMA, we we kind of, you know, work with them to let them know, hey, you can leverage X as well as the scoped application within a tool and a combination of both the capabilities out of box as well as deliberate customizations against this module to provide you this same business service alignment, which then again goes back to your return investment.

Kenric Wong

Right. What's that like testing cost on that application you're using right now. Can it achieve the same business outcomes your private business stakeholders are using? And look at the platform as a holistic approach to Nate's point at that executive governance. All right. So hey, you're planning on this app. You know purchase versus you know, Bill, why don't you consider what you already have and leverage what you

capability can do and see how much that return on your total cost of ownership is from a platform perspective, and then decide whether or not this meets your need.

Kenric Wong

Right. So I think as a CMA, that's what our role really is at an organization level, right. Meet with the C as with the EA's, meet with their own solution architect of that application. See if it works for them. See if it can be married into the platform already has their own. Become stakeholders of the platform as well.

Matt Espley

Yeah. And what you were talking about, Nate, as far as you know, just beyond customize versus out of box, there's also that discussion of configure versus script. And I think that's a really important one because let's say you have an in-house developer and he's really strong with scripting. And he can or she can do it well with with script.

Matt Espley

But you're missing out on the, the configuration option. And although it, you know, the argument may be I can do it a lot faster is your your gaining temporary speed and, you know, a faster time to value technically. But you're really introducing a lot of risk to degrade your ROI in the future and to increase your total cost of ownership.

Matt Espley

And I've seen it a lot where, you know, we we get into conversations on the operations side with, you know, potential clients or current clients where we're trying to help them unpack things that have been done on the platform and find ways to help support them. And one of the common things we see or we hear from from them is like a hesitancy when and trying to, you know, start a new partnership with them is, you know, I'm really worried about.

Matt Espley

Not just having your people, but like, you know, we didn't really see what they were doing, didn't really have governance in place to make sure that what they were doing was sustainable for us. And now we have our own admins who are trying to support this thing, and we don't know how to support it because it's not in the ServiceNow documentation.

Matt Espley

Like it's hard enough to go dig through documentation on on the ServiceNow site and find exactly the answer you need, let alone a customized scripted feature that's, you know, who knows where it lives. And if you don't have the skill set to really reverse engineer something, they really feel like they're painted into a corner and helpless. And I think that's a really important thing for platform owners to consider is just because you can doesn't mean you should, right?

Matt Espley

It's like do yourself the favor upfront and configure as much as possible versus script and document. Ensure documentation because you don't. You never know where you're going to be down the road with your platform support model. And it may be your own admins who know how to do it. Or maybe they don't know how to do it, and if it's another partner, they're going to have to spend time unpacking what was done in the more complex and complex, complicated and, you know, if it doesn't have the documentation, you're really, really in a hard place.

Kenric Wong

And I will knowledge, right, that you just talked about like if you have a script that's super awesome and they do the scripting, that's the only processor. That's the point of failure right there. And you can't even expand.

Matt Espley

Are you guaranteeing that person's going to be around forever? Like we talked about the war on ServiceNow talent in previous episodes. Right? Right. What is your ability to retain a really strong developer like that who is technical and can build all that stuff? Right. It's it's a really risky position to be in. So yeah, I just wanted to harp on that for a minute.

Matt Espley

You know, configuration feels slower. It feels like you're wasting time, but you're really setting yourself up for success. The more you can configure versus script.

Nate Weldon

It is easier to reverse engineer configuration versus code, for sure. Yeah. In my opinion, yeah. Since you brought documentation, it's a great segue. You know, how are you guys documenting your apps these days? You know, I've. I've kind of sort of pivoted to using AI as much as possible. You know, we're we are we are building our own agents.

Nate Weldon

We're doing documentation. We have a couple of products that help out with documentation. But even so much as downloading stories as a PDF or small, if you want to, you know, that's a really simple way to create some release notes or create some, you know, some high level documentation and even even in-depth documentation. The one thing I've I've found is like click here, click here.

Nate Weldon

Style test scripts. It's not I haven't seen a product that's there yet. So I still use a lot of the, the legacy stuff like tango or Snagit or anything else when I'm doing like training documentation. But seems like a great opportunity for some AI capabilities.

Matt Espley

100%. Yeah. Generally, you know, as a as a platform owner, make sure that your development team is documenting everything in a place that you know where it is, whether it's like in a runbook, it's, SharePoint. I, I'm the advocate of keeping everything documented within the story. Right? So you have your business case, you have your technical requirements, you have attachments in there.

Matt Espley

You have a history in the activity stream of what was done, why it was done, decisions all that stuff can exist in a story. And like to your point Nate, you can export those to PDF. They can live in your agile module. It's I think it's that that it's like yeah baseline what everybody should be doing. And then yeah, the AI tools definitely help to make it a lot more efficient.

Matt Espley

So you're spending less time on documentation. But again, you know, trust but verify when you're, when you're using notes.

Kenric Wong

So I like both y'all's approaches. In fact we do a little bit of that mixed in with, standardizing, because we are

doing a managed service for this client. Right? We, we want to have customer facing easy, accessible kind of their own design business case documentation. So we are mixing both of both worlds right in the back end.

Kenric Wong

Stories are great for the technical people to know kind of the details of how it's built. But for the customer this so many of these requirements that what enhancements and capabilities being enabled. We have a knowledge based customer facing portion of that where it shows you the full breakouts of architecture, right. You know, your business case when your business architecture, your data models, your your architectural, you know, wireframe diagrams for how we connected in the system, in the platform.

Kenric Wong

And then it's basically utilizing AI in a sense to structure a consistent, always consistent. When they look at what they ask for, it's always going to be that kV template that has the correct format of that have the exact same thing. So the the client themselves that we're managing the servers for knows that they just ask for something.

Kenric Wong

We just released it. And these are the documentation that show you the processes, the flows, the technical, the overview, the technicality. Right. But yes, we have stories back to that same article, but it's more like, you know, hey, for your technical people, this is where they can go for the details. Let's build the story. Yeah. There's the, the the wireframe of how it connects.

Kenric Wong

So we're doing a little bit of both right. To help consistently keep track of what it is we're releasing every single time platform.

Nate Weldon

Yeah. So we touched on knowledge transfer and documentation goes into that. You know, for me when I'm transitioning a skill gap, I, I step through everything that's an app engine and or App Engine studio. And you know, that which is, which is the high level, you know, fairly citizen heavy type of stuff. And then any script includes that I have created, you know, I will step through each function with those script includes, you know, any, any specific coding that I've done?

Nate Weldon

I always call that out and make sure that the customers admin team knows where all of these things are. And I try to button it all up into a document. Sometimes I struggle with,

Nate Weldon

When I was a customer, I went through a lot of rigor doing a lot of process guides where incident management, problem management, change management, and I spent weeks upon weeks writing 30 to 40 page process guides.

Matt Espley

That nobody used.

Nate Weldon

To publish that process guide once. Do you think anybody ever read it again?

Kenric Wong

You know, and that is the challenge.

Kenric Wong

I not. But I see the struggle in that. And I think that's why the, becomes more of our day to day kind of a toolset for us to help him.

Nate Weldon

Well, it's just as important to consider where you're putting that information, how you're chunking that information up. There's nobody reading the process guide, because it's a 50 page document where only four pages are actually valuable to the person reading that step of the process. You know how how you break that documentation up to focus on the right audiences, right, Matt?

Nate Weldon

You you you hit the nail on the head. Your your developers need more information then, you know, your your functional owners or your process owners. Your process owners need to understand that it works, that my state flows are correct, that my notifications are correct, that my form layouts are correct, but the how the sausage is made a problem, you know that that is, you know, also important to you, to the customers dev team and, and the people that own these things after we, you know, walk away in project

Matt Espley

I don't even know if I want to ask this question because I feel like it will go down a rabbit hole. But how do you guys see the captaincy? Jen, I impacting this discussion and decision around when to build versus a unit versus by given the tools that are available for people who maybe weren't previously. Able to develop.

Nate Weldon

That is both an intriguing and frightening question, because I will tell you, I recently had to solve a problem not related to service. Now with PowerShell, which I haven't touched in 12 years. So I immediately, you know, fired up one of my chat engines and and started hammering away on PowerShell scripts. I was able to get something functional.

Nate Weldon

I was able to get something that fixed my current problem.

Nate Weldon

But I still I still had concerns about the way that code was running, because I no longer understand or am experienced in all the latest and greatest PowerShell like PowerShell libraries. Especially the SharePoint ones. But that that's a great question. I don't think I have an answer right now, because there's nothing that stops somebody that has the right level of access from going to Claude or going to ChatGPT, or using cursor or using, you know, any copilot, any of these AI products to create a script include.

Nate Weldon

And maybe it works in there, one specific use case, but maybe it breaks a bunch of other things. Code review, peer review, all of those things is important. Because when I showed somebody the PowerShell script that Claude gave me, it was a lot more than they ever thought they would ever see in a single PowerShell document.

Nate Weldon

I mean, there were 400 lines of code in this, and it was achieving three things, but so understanding that

code, knowing how to read it, knowing, knowing what everything does always is paramount. Yeah. And that's a that's a big risk. It's a great point. Yeah.

Kenric Wong

Wide culling even now even if you I mean, the question you just asked Matt and what just Nate just said, it makes it even more amplified in the terrifying manner. Because if we decide to build versus buy and we now build to delivery, customize. So we're building a custom scope application. Then we ventured into the AI on a custom application that doesn't have a capability.

Kenric Wong

What are we cross-referencing as best practice in that coding realm of a new application? Like you're looking at a completely new capability that you're building, right? So now you're coding for something that probably doesn't exist in a sense, and now you're saying, well, it works. Is it good, though? You know, that's what's so scary about it, because it will work.

Kenric Wong

Right now we're thinking, well, in the best practice.

Nate Weldon

I'm sure it has improved since now, but when when some of the analysis stuff first started coming out and I, I, you know, the first thing, the first thing I always try is a UI action because it's, it's something, you know, people you can click on it, it does a specific thing. You can have a UI action with two lines of code in it, and it's perfectly acceptable.

Nate Weldon

It does what it needed. It needs to do. I asked it for a UI action like that, and I got a UI action with 30 lines of code in it. Like I said, I'm sure it is way better now, but but because I knew better because I knew that that's not right. And oh by the way, there were suicides in it.

Nate Weldon

You know like in it, it was what's that anecdote about, you know, the guy that shows up to fix the big piece of, oil rig gear, and he bangs on it with a hammer and it starts running, right? And it's like, that'll be ten grand. It's like, what do you mean? You didn't pay for me to hit it with the hammer you paid for?

Nate Weldon

My experience and experience is key. Yeah. I knew where to hit it. Nobody else knew where. But is so that that's part of it, right? You have to know. You have to know how to read the code. You have to understand how the code works. It's great for stubbing out code and authoring code and getting your starting point, but I've yet to have anything develop me something that was 100% of the way there.

Kenric Wong

In our governance call for our technical architects. This actually topic actually got brought up. The developers were asking the architects, can they use AI to help them code some of the development work that we were doing for our client, and we reviewed it as a whole because, you know, in one hand, somebody was saying, I think they can use it, but we have to be responsible for looking at it and reviewing it before we implement it.

Kenric Wong

Others are like, well, now I'm going to be reading code that I know I would never build, but now I'm taking a lot of time to review something that I could easily tell them. Well, here's the right way to do it. Do not only

simplify the code, but, you know, just we decided on the fact that after as a whole, that it wasn't good for the program because one, if you type that same ask in a different way on any of the AI engines, it's going to spit you out different codes every single time.

Kenric Wong

And now you can really do it very well because you have.

Nate Weldon

with ServiceNow. And I am getting very familiar with Microsoft's technology. Again, we're on my way later in my career than I expected to. There's six different ways to do any one thing. This is, are there more efficient ways? You know, we should always be striving for the most efficient code that eats up the least amount of memory, right?

Kenric Wong

Yep, yep.

Nate Weldon

Or causes the least amount of database calls or, you know, everything related to, you know, on the doc site for performance management and coding.

Nate Weldon

Perfect is the enemy of good enough sometimes.

Matt Espley

And I think that's a good segue into kind of a summary, right? We've talked a lot about the do's and don'ts of customizing, but to kind of take it back full circle, you know, we're considering customize versus buy, right. Like what are the considerations now that we've kind of hammered on this for 45 minutes? Well what's the takeaway in terms of deciding okay, now should I buy or should I customize.

Matt Espley

Right. Like what is it like an 80/20 rule?

Nate Weldon

It's the TKO rule, right? If it's going to take me a year of a dev team, you know, a 3 to 4 person dev team to build that functionality that I could get on day one from ServiceNow. That's a no brainer to me because I mean, even if you're using offshore development, there's still a.

Nate Weldon

That app is still going to cost you a few hundred thousand dollars over the course of a year in development. At a minimum.

Matt Espley

So unknown down the road in, in risks of sustainability and maintenance.

Kenric Wong

Exactly recapping your your question, Matt, is really the the value management plus governance plus the use case because what you mentioned, Nate, exists in almost any business situation. It could technically be a year

in development. It could be six. It could be three, but it could be two. And I think what we need to come down to is the value and business outcome.

Kenric Wong

It is what you're trying to achieve is a return on investment regarding a specific total cost of ownership of an actual application business function. I think holistically, when you mentioned resourcing, management, maintaining your your people, right. Your, your your side of us is the the lemon worth the squeeze at the end for you to customize and does it meet that requirement?

Kenric Wong

And if it does, then you're right. Maybe customization is the way to go, but if you are seeing a trending downwards of not just increased risk, but overhead or maintenance, then perhaps let's see what we can leverage from an out of box with ServiceNow is allowing. And that likely will be your best bet because you are now either you know, status quo or increasing your return on investments just on that that decision alone.

Kenric Wong

Right. So that's I think at the end of the day, is your value management with your total cost of ownership.

Matt Espley

All right. Well, thank you, everybody for tuning in to this episode of The Blueprint. I hope this gave you valuable insights into, you know, how to navigate the build versus buy decision. And, please join us next time as we continue to break down the what, why and when of ServiceNow capabilities. And until then, take care and keep building.