



What to do about your BizTalk Server environment?

Date:
January 31st, 2023

Author:
Jim Bowyer

Do you have an existing BizTalk integration? Are you wondering about how to manage its future? Do you want to hear what others are doing? If so, read on. Let's start with the good news: (1) you are not alone, and (2) the maturing world of iPaaS tooling has given you more viable options to choose a future from. In this article, we will briefly share the challenge of BizTalk Server migration and what DMS has experienced enabling other BizTalk clients to date.

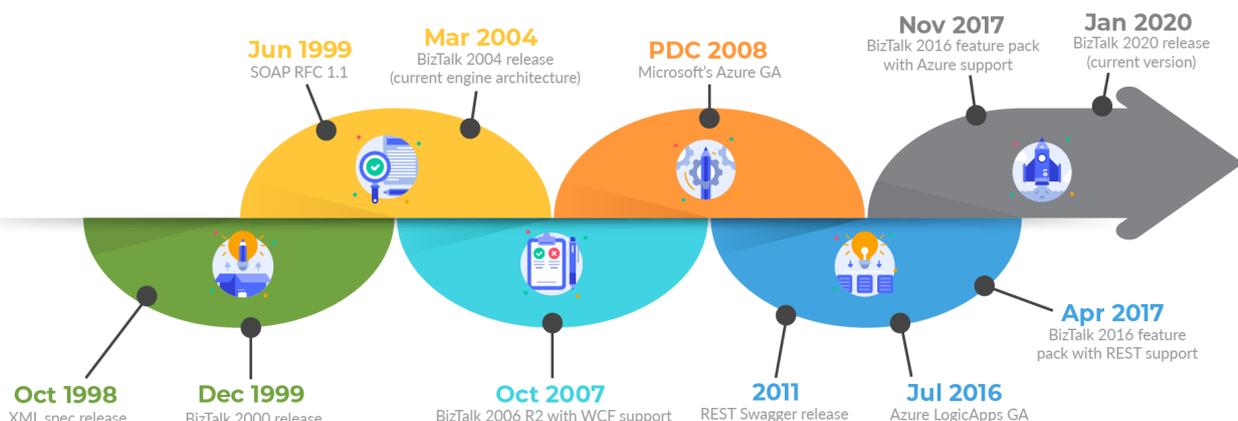
01

Why avoidance is a time limited option:

Fact is, like the rest of us, BizTalk Server is not getting any younger. Into the third decade of its product life, while it may have served many well, the reality is that BizTalk's core engine is founded on a XML/WS* era that is increasingly showing its age. This makes availability of skills harder to find and a complete rebuild of BizTalk by Microsoft less than practical and questionably desirable. Even if Microsoft extends support again, the inevitable is looking increasingly likely: platform updates will become increasingly cost prohibitive and customers (even those on the current version) are creeping closer to a forced decision between migration or eye-watering support costs. So even if the advantages of moving to modern iPaaS equivalents like Azure's LogicApps/Functions etc. don't strongly appeal (and we think that they should), you are running a platform that is facing an end time. Unless you can switch it off (presumably along with one or more of its integrated systems) then the reality is that someone in your organization is going to be confronting its migration.



Microsoft® BizTalk® Technology Timeline



Timeline created by www.redlidconsulting.com, BizTalk logo source: <https://www.royalcyber.com/technologies/biztalk/>

02

IaaS Migration falls short:

Why not just migrate the thing and be done with it? Sadly, it is unlikely to be that easy, especially if your business used BizTalk as intended. By its very nature BizTalk is not a stand-alone environment, instead it enables integration between critical business systems and processes. Often these are multiple and complex. Yes, these days you now have the option to virtualize BizTalk and potentially along with related environments, move it to IaaS ...but! ...even if this 'big bang' approach is viable, it will not solve our original concerns. We will still be operating the same server software and facing many of the same server support and operation concerns that we had before, albeit in our shiny new IaaS facility. Ugh.

03

iPaaS to the rescue:

Which takes us neatly to the decision point where many current BizTalk customers find themselves today. Why not just move our BizTalk integrations and migrate them to modern iPaaS services? Once there, our integration logic and code could operate unincumbered by server upgrades and maintenance. Such issues then become the cloud partner's problem to solve while we get to focus more on our own business. Would that not be a significant improvement? Well, DMS thinks it would, and so do many BizTalk customers we speak with.

...but how??

Sadly, the devil is in the detail of the migration versus the very real and available iPaaS destination. This is where a lot of customers get stuck and where teams like DMS can help expedite progress. Today we will outline common approaches we've taken with other BizTalk clients moving to Azure iPaaS. Certainly, there are other capable clouds outside of Microsoft Azure but within this article we are only going to focus on migration to Azure. As we'll explain, there are benefits to staying in the Microsoft house, especially if you have any significant investment in BizTalk today. Onwards...

04

Dispelling migration tool 'magic':

Before we jump in, it is important to dispel the myth of a BizTalk to iPaaS 'migration tool' - to date we have not seen any that do the job end-to-end and even if they did, our opinion is that such dreams would quickly become nightmares once put into real world use. There are too many permutations and dangerous assumptions, not to mention again the different eras of technologies. Even if you found that as-of-yet undiscovered 'quantum era migration tool', it is highly likely to wreak havoc on you by repeating the design sins of the past or at least creating new ones based in the modern era. Point in case for past design mistakes: using the wrong tool/service for the job. We've seen more than our fair share of BizTalk implementations doing data copy work that should be the domain of an ETL (Extract Load Transform) tool. Even if you got away with it to this point, repeating this in iPaaS is the equivalent of building yourself a trap. If the trap isn't triggered by increased scale loads, it will easily trip on the underlying vagaries that are inherent underneath today's disconnected cloud services. So, as for automated migration tools, our advice would be to save yourself the disappointment. Besides, you'll be glad when you've taken a clean sweep to this existing integration and put it back together in an orderly way that takes full advantage of modern iPaaS.

1.
The Start - Itinerary:



The first and critical place if you don't already have one is to start creating a catalog of your environment's integration interfaces and processes. There is no absolute wrong or right way to do this, and this can be a place where some tooling can help... for example the BizTalk documenter tool. Once listed you will need to group dependencies and assign priorities. Our top advice here is to ensure that you are driving the interface priorities of the business. We've seen clients prioritize interfaces to the top of the migration list because they are an attractive technical problem, only to find out the business rarely uses them.

2.
APIM to the rescue:



Ok, so you have a system that communicates with SOAP XML or similar, and that system absolutely cannot be touched or changed and needs to continue to operate as is, where is. This happens more than you think and we're happy to share that Azure API Manager (APIM) is fantastic service to enable this. With a few minor config tricks, you can have that old system talking to what it thinks is BizTalk while behind the scenes, APIM takes the payload and redirects it to Logic Apps bringing the result back. Your original system will have been pointed to another URL but aside from that, it will be oblivious to all changes. Neat.

3.
**Getting to/from
On-Premises:**



Yes, a common reality is that many of these migration scenarios will still need to connect the new iPaaS components back to on-premises which is likely where your BizTalk environment operates today. If your organization needs a VPN/S2S link for other cloud connectivity and has a network team that can support them go ahead and use. However, we don't think VPN/S2S links are the only suitable solution. Indeed, we've found services like Azure Hybrid Relay for data and API gateway calls to work well. If Internet level QOS and security is acceptable for your workloads then our opinion, is they are infinitely preferable to setup versus VPNs and the like and they work well. The documentation on these services may seem sparse but we believe that it's going to be one of the more pleasant surprises about your migration because these are typically one of the easier aspects to set up.

4.
Orchestrations:



There is not an XLANG engine in Azure waiting to run your orchestration and from what we hear Microsoft has no intention of creating one. Really, that's ok. LogicApps has matured into what we suspect many of the original XLANG feature team aspired orchestration would eventually become. While it's manual work to recreate these business processes in LogicApps, we think you'll appreciate the rewards. For example, once you run your first LogicApp and see how much easier it is to go examine its runtime executions - that alone will likely be enough to ensure you will not be wanting to go back.

5. **Maps/Transforms:**



If you are like many BizTalk customers, you have invested a lot in those XSLT based Mapper transformations. Being told to simply throw them away because “JSON is cool” is far from helpful. The good news: Microsoft has done the hard work for you and you can bring your existing BizTalk maps into your new logic-apps largely as-is. That is a massive time saver and will greatly reduce risk and increase reuse for your migration.

6. **Custom adapters:**



While you cannot move adapters or their code as-is, you can leverage subcomponents and snips of your .NET code. Most commonly we move these to Azure Functions. Certainly, it is an investment to learn Functions, but we think that it's a modest one and it can be done either in the familiar environment of Visual Studio or the new realm of VS Code. Both work fine and both can be used interchangeably but VS Code has the added appeal that it can open your team to a broader audience of developers from different platforms and heritage. That in turn buys you insurance on availability of skills to maintain your iPaaS investment into the future.

7. **Keeping Secrets:**



Another common requirement for integration work is the need to store passwords, connections strings and the like. BizTalk Server provides a Single-Sign-On Service that you may have utilized for such purposes. While you can store items like these in environmental variables, it is not recommended. A far better solution is provided by Azure's KeyVault. Store your secrets in there, run your iPaaS services under a Managed Identity and away you go. For the modest effort invested the payback is handsome.

8. **Testing and DevOps:**



There really is no replacement for good quality testing. If you are lucky, you may have some existing tests that you can leverage. In any event this is a good opportunity to invest. Aligned with modern DevOps practices testing and automation of your finished solution will enable a far more resilient and sustainable solution. Consider how long it takes you to stand up your existing BizTalk Server environment today. When was the last time you deployed it end-to-end? Maybe you have invested in its automation and got a partner like DMS to add DevOps to your existing BizTalk solution, but our experience is that most have not, and the thought of recreating these integration environments is painful. Besides it's not trivial and a totally unfair comparison to the automated deployment available in iPaaS enable today. If you feel even a little bit intimidated by touching your BizTalk environment today, this payback alone is worth the time and effort to modernize your integration solutions.

Interested to learn more and explore potential collaboration to enable your BizTalk migration?

We've covered a lot in this article, with much more detail that we could go into. Our hope is that this article gives you a better idea of what is possible and increased your confidence in the possibility of a BizTalk to iPaaS migration. You can certainly undertake these migrations yourself but if you desire to reduce risk and expedite the migration, we'd recommend you consider leveraging a partner like DMS. At DMS our integration practice has completed several BizTalk to Azure iPaaS migrations and has available consultants who have been working with BizTalk for over twenty years. That knowledge and experience can be invaluable so if working with DMS is of interest, we welcome your enquiries at contact@dmsgroup.ca . Either way we wish you every success in your own BizTalk migration and encourage you to make a start on these because as we described earlier, this is very unlikely to be a challenge that goes away by itself.

- BTS2020 "Mainstream support has been extended from January 9, 2024, to April 11, 2028" see <https://learn.microsoft.com/en-us/lifecycle/products/biztalk-server-2020>.
- *The views expressed in this article are DMS Group's own and do not necessarily represent the views of Microsoft or any other 3rd party.*
- *No warranties expressed or implied.*