TOP 10 TIME KILLERS

FOR INTEGRATION PROJECTS

Bits In Glass

Unlock Your Potential

INTRODUCTION

There are endless moving parts when working on an integration project. From people and systems to data and documentation there's a lot to keep track of to make sure everything stays on track.

Let's take a look at the **top 10 time killers** for integration projects so you can get ahead of them and meet your targets.

This guide is associated with our 'Beyond the Interface' webinar series.

For more helpful integration project tips, tricks, and best practices head to ***bitsinglass.com/webinars*** *to watch the full BIG Expert webinar with Eric Carsted, our Director of Technology!*

TOP 10 TIME KILLERS

10 TOO MUCH DATA VALIDATION

Data validation is the process that checks whether a given incoming piece of data is correct and complete before processing it. Trying to validate every piece of data to 100% with zero errors is a huge task that can never be accomplished. What's correct today may not be tomorrow. For the most part, validation belongs in the data source or target system, not in the integration layer.

TIME KILLER TIP:

Build enough, but ... a) Use a tool that can leverage an API spec language such as RAML as it documents and enforces data constraints. **b)** Validate required fields for existence of content. **c)** If an error is costly, such as taking a long time to come back as an error, validate that. **d)** If you see something is generating a lot of errors in the logs, validate that.

Ask yourself, what's the difference between the data failing validation, or if it fails in the target system? They both generate an error. Now, what is the cause of the error if the target system now accepts the data, but you didn't update the integration?

9 NOT DOCUMENTING DECISIONS

Documentation is essential to the success of any digital transformation project, especially when it comes to decisions made along the way.

Everyone needs to be able to refer to what has been approved, changed, or removed and why to be able to successfully move forward.

TIME KILLER TIP:

Keep a decision log! Design and implementation questions or clarifications are going to come fast and furious during integration projects. Without a decision log, a lot of time will be wasted when team members are added, changed, or if time has lapsed from the time decisions were made.

8 DEFINING “DONE”

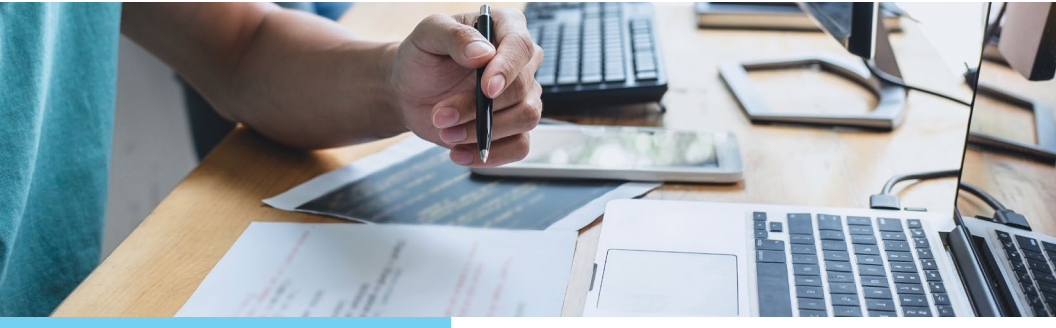
This might be the most common time killer for integration projects as items such as data validation, testing, and documentation can go on forever. It can be hard to know when to call it and say “it’s working how it should and we are confident in the quality, so we’re done!”

TIME KILLER TIP:

Define what “done” means for your project at the start. Before you begin building, make sure the entire team has a good understanding of what “done” means (number of test cases, level of documentation, etc.) so everyone is working towards a common goal.

7 NAVIGATING FAST VS. GAPS

Waiting for requirements to be “done” and signed off on can cause major delays in your project schedule and cut into development time. Developers can’t develop without requirements and can go from getting a lot done to sitting around waiting.



TIME KILLER TIP:

Get started once your interface mapping document is ~80% complete. By this point you should have a good understanding of all your data sources, connection points, and most of the mappings. This is enough to lay the foundation, the rest (and sign-off) should come within the next 1 - 2 weeks; before the first cycle of testing.

6 NOT ADDRESSING NFRs UP FRONT

NFRs, or non-functional requirements, are requirements that specify criteria that need to be met such as expected response time, maximum response time, TPS throughput, security, scalability, data privacy, maximum record size, maximum file size, etc. rather than specific behaviors.

Fixing issues after finding out that an interaction is taking five seconds at customer acceptance testing when they are expecting one second, or running out of memory when a data load file becomes a million records rather than the 200 used for unit testing is going to take A LOT of rework.

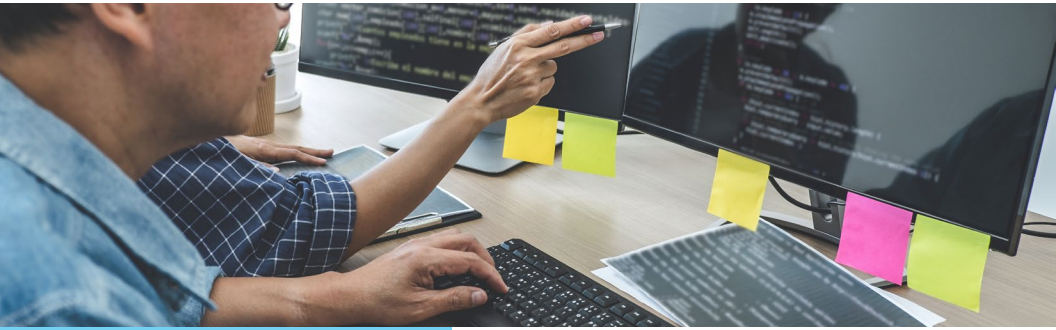
TIME KILLER TIP:

Gather and document NRFs at design time. Ensure the developers and architects are aware of these BEFORE development begins and verify they are confident in meeting them.

If you can't get definitive values, take your best guess and document that. Big misses are hard to handle within code alone, and architecture changes take time (and money).

5 NOT HAVING A DEVELOPER BACKLOG

Developers can be blocked on a task for a myriad of reasons beyond their control. Not keeping a documented backlog of work that they can switch to while others remove those roadblocks will tank their productivity. Establishing what they should work on next will waste both their time and the time of the people making that call.

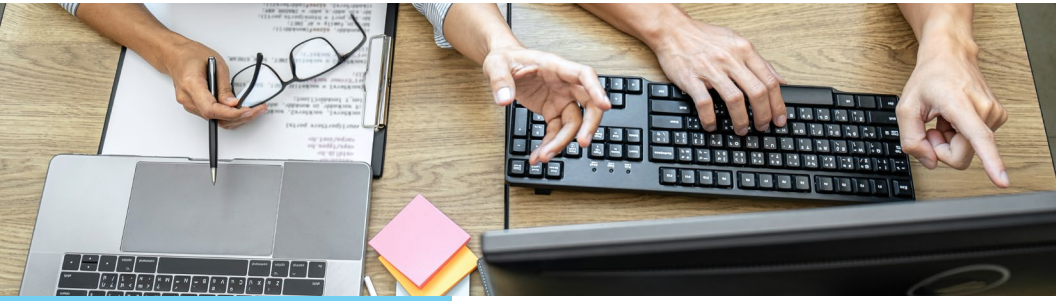


TIME KILLER TIP:

Ensure you keep a well documented and prioritized backlog. This will allow your developers to switch to the next highest priority task without delay when blocked. When unblocked, they can return to the higher priority task.

4 MAKING DIFFICULT DECISIONS WITH NO MVP

With integration projects having so many moving parts and dependencies it's easy to fall behind schedule. If the end date can't be changed, tough decisions will need to be made and committed scope may need to be deferred or cut.



TIME KILLER TIP:

Establish an MVP at the first sign of schedule slippage, or better yet, at design time. Developers should build the MVP first and additional content next. This will allow you the time to make those decisions as well as not waste time developing functionality that could have been cut. That being said, don't sacrifice quality to get stuff done. Make sure you're still testing and have proper error handling and logging in place.

3 LACKING A TEST SUITE

Testing is a HUGE part of any integration project and it's essential to its completion and success. Without a formal test suite in place, manually testing work as you go can not only cause delays but can also risk functionality.

TIME KILLER TIP:

Establish a formal regression testing suite. Automated regression testing saves an ENORMOUS amount of time and is essential when you need to change core functionality. ***Click here*** *to learn more about why regression testing is important and the value it brings to any project!*

2 MOVING GOAL POSTS

With any integration project, there's going to be a desire to add new fields, features, and functionality once there is a product. Changes after delivery will require additional building, testing, and deployment. Not having a formal sign-off will contribute to "just this one additional small thing" scope creep and can destroy your project's schedule.

TIME KILLER TIP:

Get functional sign-off before or soon after development starts.
Try and stick to your scope as much as possible. You can always add new/updated requirements to future sprint cycles.

10 UNDERVALUING TEST DATA

Data is at the heart of any integration project. After all, it's what you're integrating. To build and test integrations test systems need a sufficiently sized data pool that represents both the volume and variations to support the test cases. Every test (unit, integration, acceptance, performance) needs data to complete.

Integrations often consume data (mark it as sent or is date dependant) or source and target systems are refreshed or not in sync. Every time you pause to solve a data related issue is time that is lost and is consistently the #1 time killer on integration projects.

TIME KILLER TIP:

Constantly ask yourself, "Do I have the data to complete the next test the project needs to run to progress?" Think about all the scenarios that may change data and break your development and testing environments or yout automated regression tests.

Consider segmenting data ranges to be used by different groups so teams don't step on each other if sharing environments. Sadly, even by doing this you'll still run into data issues, but these tips can set you up for long-term success.

bitsinglass.com