www.spiff.com

REPLACING EXCEL FOR COMMISSIONS

SPIFF

This is the message Excel has been telling those of us who have been trying to force it to do commissions.

This post is a call-to-arms for forward-thinking finance and sales ops professionals. It's how you can make sure your sales and finance organizations move arm-in-arm into the next epoch of commission software–and business software in general.

It's job security for the next decade. I'm talking about replacing Excel / Google sheets for your commissions.

This is a big part of our "why" at Spiff. Think of it as the technical reason Spiff exists. Excel just doesn't cut it for commissions. Should your job be wading through 13 levels of nested ifs? Commissions are about driving behavior, growing top line, and making the finance team look like heroes to the entire organization.

Before you freak out, I know what you're thinking...

But I'm familiar with Excel. Here's the good news. So are we at Spiff. We included nearly every Excel function in our system. But you need more than Excel. Excel just doesn't work well at scale. The future of commissions will be visual logic builders that automate every tedious commission process.

HERE'S THE BIG SECRET:

Your Excel spreadsheets have turned into mini-computer programs. Those mammoth commission spreadsheets you're building and holding together amazingly well with bubble-gum and duct-tape...yep, they've turned a complex and unmanageable system.

You need to the tools to manage all of that complexity but you want the familiarity of Excel. That's our goal at Spiff–giving you all the power of Excel with the tools to automate at scale.

Spiff limits its system to just 6 commission building blocks. Once you get the hang of using these building blocks with Excel functions, building commission logic is actually fun. And it scales.

Spiff is part of the no-code movement. We build software that gives you the power of developer tools but with a visual interface designed for business users. We believe the future of commissions won't look like Excel or learning a programming language.

We'll show how we are doing that in this post.

Here are the myths we'll examine:

THE 4 PROBLEMS WITH EXCEL FOR COMMISSIONS

There are four reasons you shouldn't use Excel for commissions:

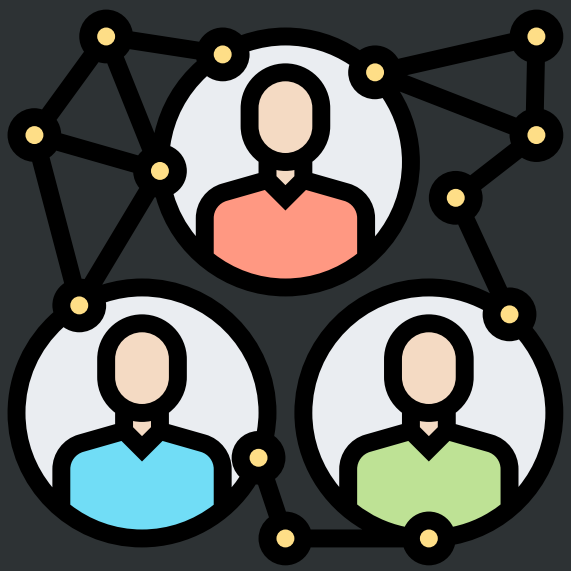
1. It's not shareable
2. It's not connected
3. It's not built for commissions
4. It's a programming language that isn't designed to scale

Let's look at these reasons in more detail.

#1

Excel Is Not Shareable

Excel is easy to share, right? All you have to do is post the file to Slack. Or you a link to an Excel Online instance.



The problem is that Excel isn't multi-tenant. Multi-tenancy means a single instance of software can serve multiple customers or users. For example, let's say you install Salesforce as your CRM. To do this Salesforce creates a company-specific version of Salesforce for your company. When you first log in you will notice it has no company-specific data for your company yet. This is called a new instance of Salesforce. Salesforce has built its software so that it can easily add new customers and each customer gets access to all of Salesforce's functionality. Let's call this customer-focused multi-tenancy. Nearly every major SaaS company offers customer-focused multi-tenancy.

Does Excel?

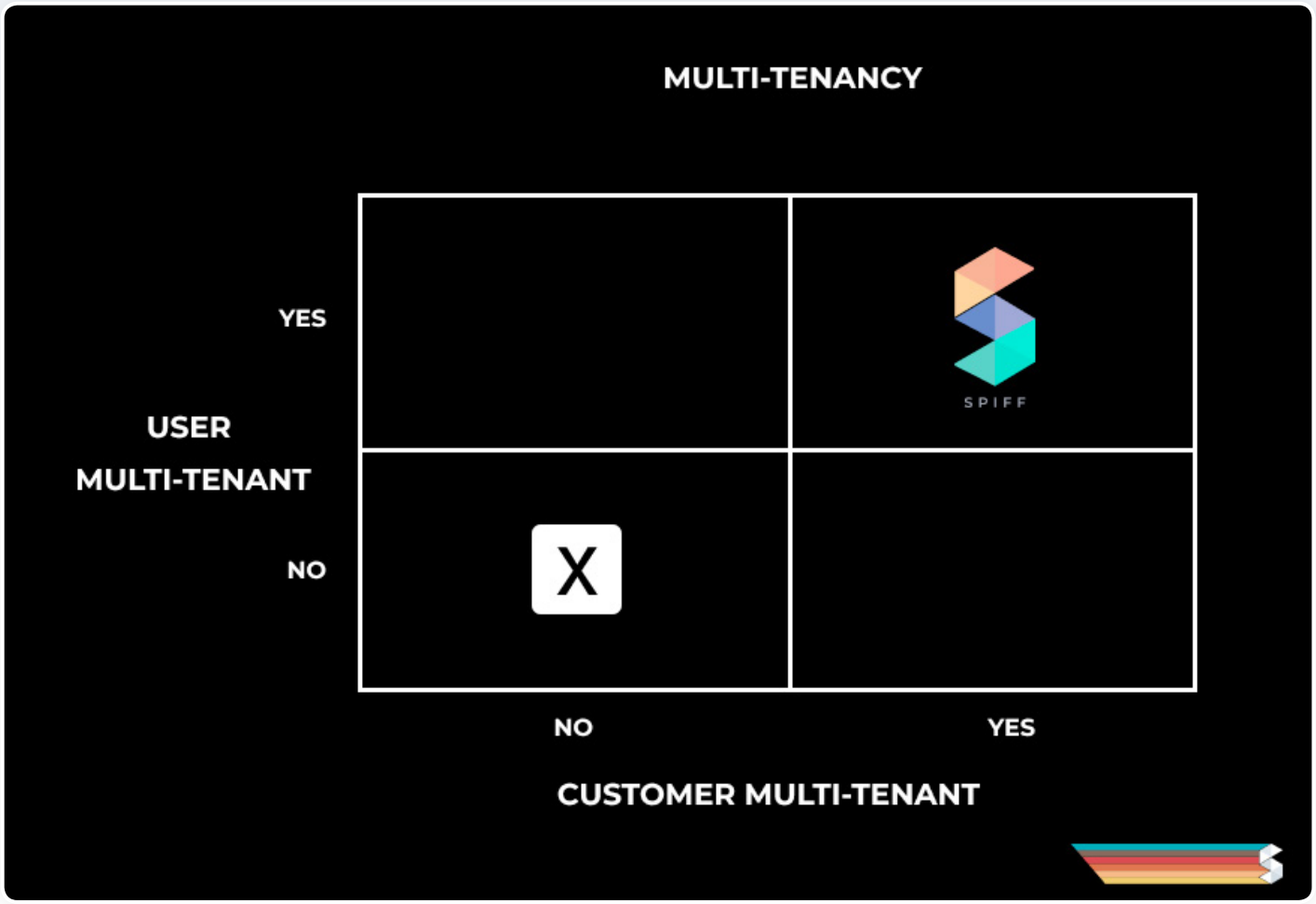
Well, when you build your commission system in Excel, it has a single customer tenant–just your company. Is that so bad? Not really, honestly. You probably don't want to share your commission system with other companies anyway. Don't get me wrong, there are big benefits that SaaS companies can offer to customers because they spread the cost of developing new features across hundreds of companies. When you build your own Excel commission spreadsheet you are 100% responsible for all of the costs of building, maintaining, updating, and improving your own commission system. On a SaaS platform you usually get most of that for free.

But it turns out there's another kind of multi-tenancy that is more important: user-focused multi-tenancy. User-focused multi-tenancy allows many different types of users to get their own personalized instance of a piece of software.

This is where Excel breaks down. You can create a commission spreadsheet in Excel for your company but what if you want to share it with 500 reps? Now imagine those reps are grouped across 15 different commission plans. Now add in the concept of managers who should be able to see the commissions of their team members but not other team members.

You can build your Excel spreadsheet, but you can't partition and share it in a way where all of the right users can only see the information that they need to see.

So Excel is single-tenant for customers and single-tenant for users. For commissions, the first of these problems is manageable but the second is not.



#2

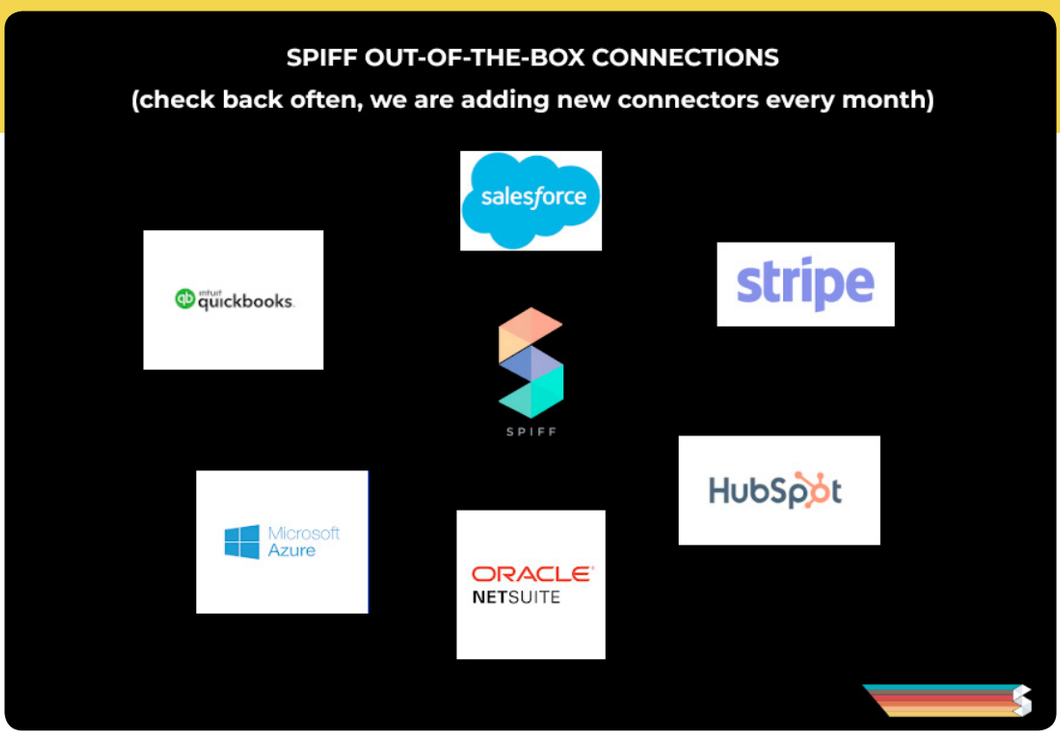
Excel Is Not Connected

Despite awesome efforts by Microsoft to make Excel (and Excel Online) more connected, it's still not a connected system. For commissioning, companies often need access to data objects directly from their CRM, ERP, and Payroll systems.

How easy is it to pull data from these systems into a central spreadsheet, manipulate the data, and keep it updated in real-time? It's virtually impossible.

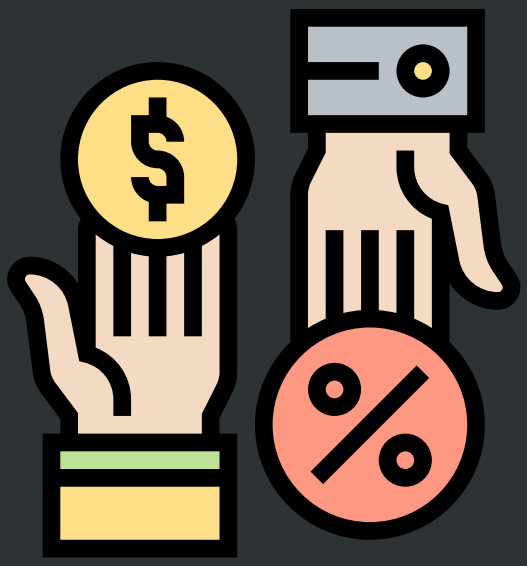
We've worked with hundreds of clients spreadsheets. 100% of the time, we see that clients accomplish this by cutting and pasting their data from these other systems into Excel. They then have to manipulate the data and normalize it. After this process, they run commissions.

This process is slow, manual, error-prone and it isn't real-time. We built Spiff using API-first development. That means Spiff connects easily to new apps. Here's a partial list but we are adding new connectors all the time.



#3

Excel Is Not Built For Commissions



In the world of software development, you'll often hear developers speak about coding languages as "opinionated" or "un-opinonated."

Opinionated languages have strong norms around the right way to do things. Sometimes they even force you to build things in only one accepted way. Un-opinionated languages don't have norms around the "right" way to build things.

Excel is the ultimate un-opinionated language for business users. Excel doesn't care what you use it for.

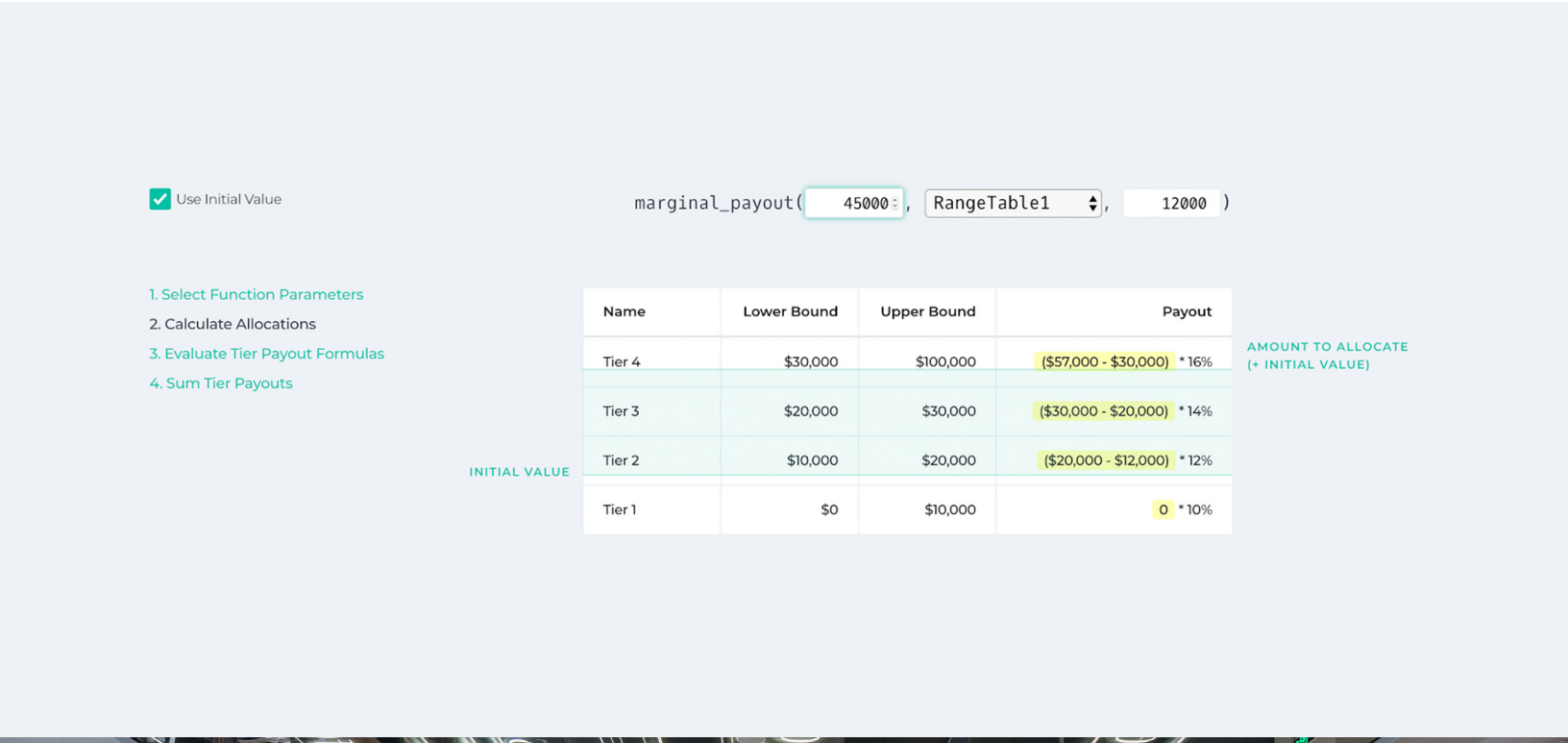
Let's take the example of commissions.

If I were building an opinionated tool for commissions, it would probably prompt me to set up Plans, Rules (for paying out commissions), and give me structured help for building things like Quotas, Teams, Accelerators, Roll-Ups, etc.

Excel won't do this for commissions (or really for anything). It's a blank canvas. For that reason, it's hard to know if what you are building is right. It's also hard to benefit from best practices.

Spiff offers a whole range of commission-specific functions. A great example is our marginal payout function. The marginal payout function allows you to easily create tax-bracket style commission accelerators.

Here's a little view of the documentation we've created to allow you to play around with the function and see how it works:



#4

Excel Is a Programming Language that Isn't Designed to Scale

Even if you could make Excel multi-tenant, connected, cloud-based and commission-specific, it's still a poor system for managing complex commission programs.

According to this awesome talk by Microsoft Research professional, Emery Berger, Excel is a "functional, reactive [programming] language." Any developer looking at the Excel function library will recognize that Excel is a full-blown, functional programming language.

The brilliance of Excel is that it combines a database, data visualization, and programming logic all in one interface.

Here's the catch...

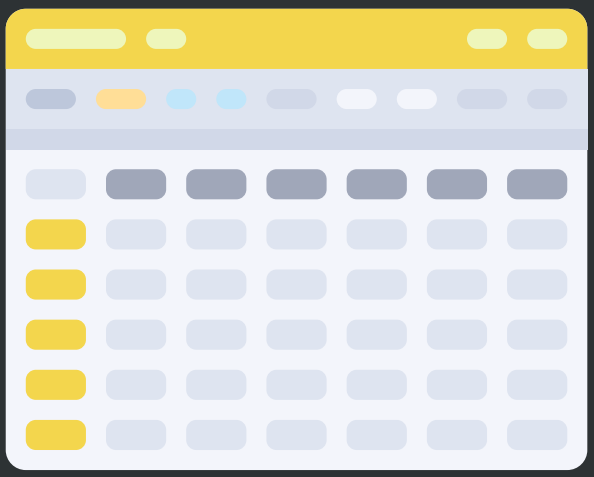
Excel works great when:

1. The data you need is relatively small (thousands of lines vs hundreds of thousands of lines)
2. You can hand-enter or cut-and-paste in the data you need
3. The logic you need isn't too complicated and is mostly mathematical

But as your Excel model becomes more and more like an actual software program, you will find that Excel lacks the features to maintain complex software at scale. You end up with what's called technical debt because your are trying to build a software program in a spreadsheet.

That's why no one uses Excel as a programming language to build anything other than spreadsheets. And spreadsheets break at scale. Emery Berger mentions that according to his research 90% of spreadsheets have major errors rendering their data inaccurate.

Excel vs Existing Commission Solutions vs Spiff



So why doesn't somebody just fix these problems with Excel? We could start by putting Excel in the cloud. Microsoft has already done that. That solves a big part of the problem. But it wouldn't change the fact that Excel is still not user shareable (multi-tenant) and connected.

Why don't we just make Excel more shareable and connected? There are several companies trying to do this:

- Airtable
- SmartSheet
- Workiva

But these companies still aren't commission-focused. Ok why not just put a commission-focused, shareable, connected version of Excel in the cloud?

Spiff's closest competitors do this. Their solutions are multi-tenant, connected, and built for commissions. But even these competitive products have two major drawbacks compared to Spiff. They aren't maintainable or scalable.

MAINTAINABILITY

There are several large competitors whose products can handle just about any commission scenario you can imagine. They've got dozens of integrations, hundreds of reports, and a big list of customers. But their customers are dissatisfied because their software is expensive to set up and maintain. These vendors sell professional services masquerading as software. They take 6-12 months to get you set up. Then every time you have a change they send an army of outsourced consultants and charge you $20K. Oh and by the way, you typically have to schedule their outsourced professional services team a quarter or two in advance. Most of their clients get onboarded a few weeks before they roll out their next year's plans and they have to start the whole process over again.

They can't provide you with a commission system you can maintain in a timely and cost-effective way. We've covered this topic in more detail in our post about why most commission software stinks.

SCALABILITY

The good news is there is a new crop of competitors that provide software you can self-maintain! The drawback is most of the systems just move Excel to the cloud.

But they still have a problem with #4 above. They are still using a functional software language that doesn't scale. They don't offer named objects, reusability, tracing, advanced error handling and testing.

They have created a commission-focused version of Excel in the cloud while Spiff has built a next-generation, Excel-replacement technology. We're trying to avoid just moving your Excel problems to the cloud.

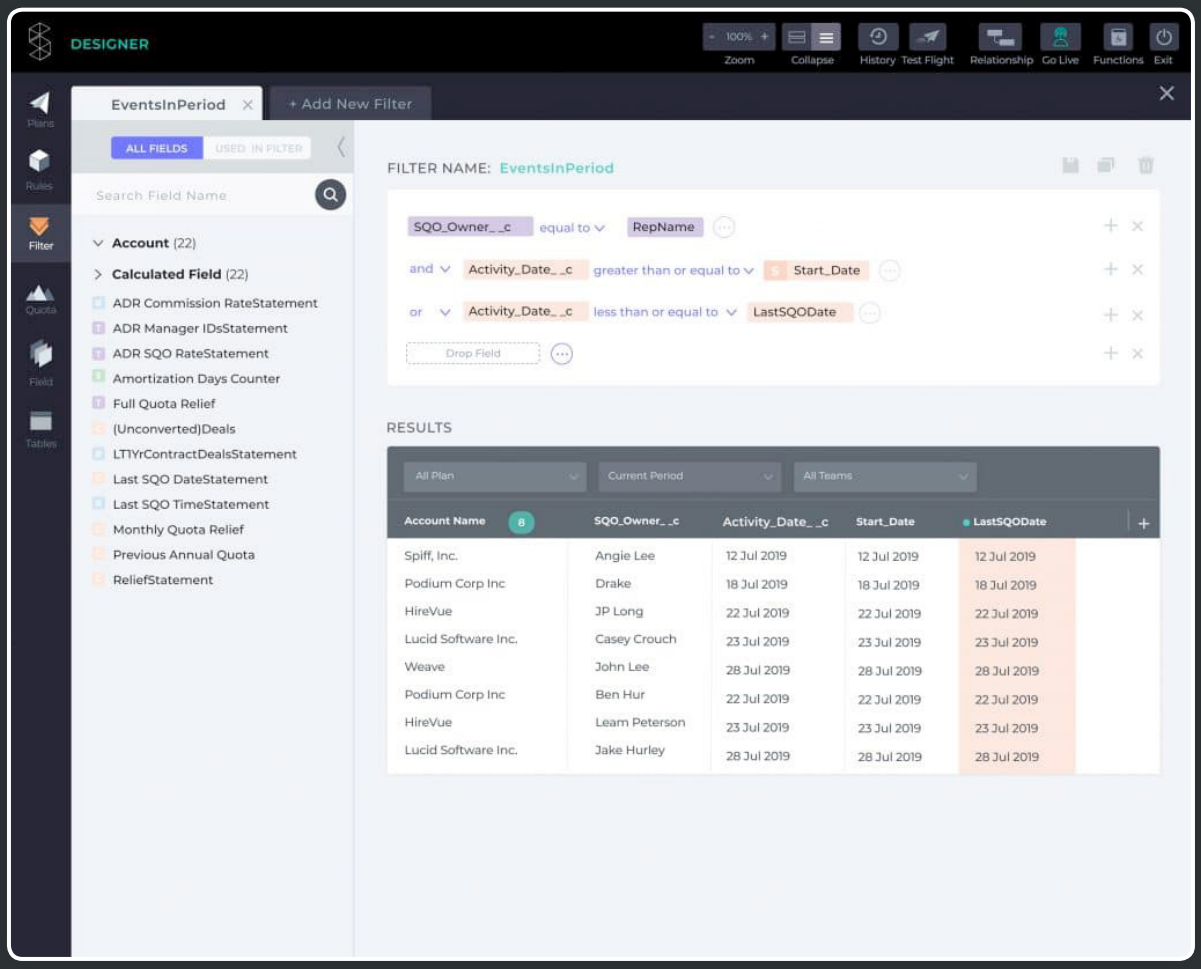
Spiff believes that you need all of the power of Excel coupled with the scalability of enterprise software. How can we do this without becoming programmers? This is the exact goal of the no-code movement. Below, we'll give you background on the no-code movement and show you exactly how this is better than Excel or Excel-like commission systems.

The No Code Movement

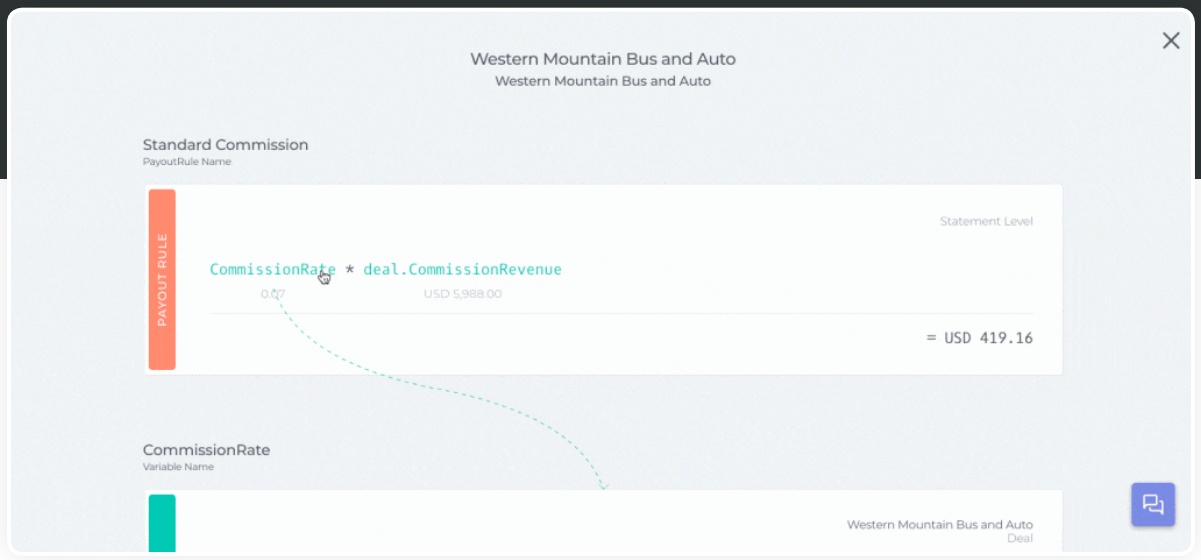
Spiff is part of the no code movement. Many large VCs and companies believe the low-code movement will become the primary way people deploy software in large enterprise applications.

The no code movement allows you to use visual interfaces to facilitate the creation of commission logic. For example, Spiff allows you to visually create filters like you would in Salesforce.

> “The Low Code/No Code movement will provide technologies with amazing DX (developer experience) systems.... No code solutions...give knowledge workers powerful functionality without requiring engineering resources. This includes easy automation, such as Zapier and UiPath, and business apps within the Salesforce and Workday ecosystem.”
>
> BESSEMER VENTURE PARTNERS, STATE OF THE CLOUD REPORT



Then you can use the results of these filters and basic Excel functions to create building blocks. You use these building blocks to break down your complex commission system into small simple calculation steps.



Interested in Spiff?

Schedule a demo with a commission specialist today.

SCHEDULE A DEMO HERE

SPIFF