Gradient Works

The only

LEAD DISTRIBUTION GUIDE

you'll ever need

The only lead distribution guide you'll ever need

Hello and welcome to our (extremely) comprehensive guide to lead distribution. You may also call this lead routing, or lead assignment. Maybe even lead management. They're all horses of a similar color, and we've pulled together a ton of resources for marketing and sales teams to help manage/route/assign/distribute leads.

We've included information for everyone. From beginners (what is lead distribution? lead to account matching overview), to more experienced marketing and sales ops readers (implementing advanced round robin, Salesforce lead assignment rules, etc...). Don't feel like you need to read this in order or from cover to cover. Use the table of contents on the next page to navigate to the sections you need most. Or read it all, that's great too.

Either way, please enjoy. Let us know if you have questions.

Table of Contents

What is Lead Distribution?

When a potential prospect shows interest in your product or service, you probably consider them some kind of lead. If you're in a B2B space, someone may sign up for a webinar or other marketing activity, fill out a demo request form on your website, ask for more information from your chatbot, and so on. These are **inbound leads** - someone outside your organization comes to you and expresses interest.

You may also have **outbound leads**. These are sourced from someone on your own team, like an SDR (sales development representative). SDRs may be responsible for hunting down new leads - finding prospects who haven't yet found you.

Inbound and outbound leads all need to be sent to someone on your sales or business development team so you can hopefully sell them something.

In some organizations, a new lead may need to be qualified before pursuing it as a serious sales lead. Those leads are typically distributed - or routed - to someone like a business development rep (BDR) for further qualification to decide if this lead is a good fit for a more in-depth new business conversation.

With lead distribution, companies are usually concerned with speed. You want to make contact with an interested lead as quickly as possible, because the faster you respond, the more likely that lead is to convert to a sellable opportunity. In the lead routing stage, sales leaders are typically concerned with things like speed-to-lead, SLAs, and response times. Sales leaders also want to be sure a new inbound lead isn't related to an existing customer or prospect account, so new leads need to be evaluated against existing accounts through a (hopefully automated) lead-to-account matching process. More on lead matching in a few pages.

We'll likely use the terms **lead routing** and **lead distribution** interchangeably throughout this paper. They're approximately the same thing as far as we're concerned. They both refer to the logic and rules an organization has in place to decide how to send leads through their organization. **Lead assignment** then is the next step - deciding who is assigned a new qualified lead to work in a sales process. All of this can be referred to as **lead management**, which is an umbrella term for the whole mess of things you do with leads.

Lead Qualification

Most companies will likely want to *qualify* a new lead before pursuing it as a serious sales lead. Those leads are typically routed to an initial BDR or MDR for further qualification to decide if this lead is a good fit for a more in-depth new business conversation.

Qualification is the process of determining how likely a new lead or prospect is to convert to a paying customer. This can be done automatically, with lead scoring based on fit with a set of predetermined criteria, or manually, with a person in sales or marketing determining a lead’s fit. Leads are generally qualified by marketing (MQLs), sales (SQLs), or product usage (PQLs).

B2B lead qualification criteria may include firmographics like a prospect company’s size, number of employees, industry, revenue, age, and more. You may also use the prospect’s demographics for qualification - criteria like job role or title, experience and skills, and more. Most companies have a detailed ideal customer profile (ICP) based on past sales data to evaluate against for potential fit.

In addition to a sales or marketing qualified lead based on the criteria above, companies may use product usage to qualified potential leads for upgrades or expansion. Those are called product qualified leads (PQLs).

Product Qualified Leads

A PQL involves identifying existing customers who might be primed for account expansion, typically based on customer actions in the product. PQLs may be identified manually by regular product usage reviews, or automated when a customer takes certain action in the product.

Expansion is the sum of cross-sells and upsells, and your company's exact formula dictates who you involve. Some companies will notify a customer’s AM (account manager) if they become a PQL. But other companies have a dedicated growth team whose job it is to act on these signals; the growth team receives temporary assignment of a customer during an upsell progress, and then permanent ownership goes back to the CSM (customer success manager). In yet other cases, there may be a new business rep dedicated to that line of business.

Leads vs Accounts

We’ve only been talking about leads so far. And we’re using that term *leads* pretty loosely. Generally, a lead is a new interest in your company.

Some companies define leads very specifically - a lead is an email address, for example. And Salesforce defines Leads even more specifically. So let’s talk about the difference between leads and Leads and accounts, particularly as it relates to B2B organizations.

A lead is a signal that tells you one very important thing: there is interest in your product at a particular account. The individual signaling interest matters, but less than you think.

Here’s why. People rarely go to vendor websites to request a demo form for fun. The demo request is typically the result of some internal discussion at someone’s business. During that discussion someone got assigned the task of looking for a solution to a business problem. And further, it’s unlikely the person who got assigned that task is a decision maker (except at very small companies or for very small purchases). This results in a bit of upside down logic: the lead is an incredible buying signal for the account but the individual associated with the lead is likely only tangentially relevant to successfully converting that account to a customer.

You resolve this conundrum by working the account, not the lead. When you get inbound interest, you assign the interested account to a rep so they can work that account, *starting with* the person that inbounded and then working out from there. More on that in a minute.

So let’s talk about leads and Leads. Note that in the paragraphs below we’ll use capital Lead, Contact, Account, etc to refer to the Salesforce versions of those objects. If you use a different CRM, you likely have similar terms.

A long time ago, Salesforce started using these terms: **Accounts** (businesses we can sell to), **Contacts** (people who work at businesses we can sell to) and **Opportunities** (a specific sales cycle), and **Leads**, which is kind of a combination of the other three. And then at some point in time you press a button to convert a lead and it turns into an Account, Contact and (optionally) an Opportunity. It’s kind of confusing, honestly.

Even in the simple case this wreaks havoc. A brand new person you've never interacted with before requests a demo and a Lead gets created in Salesforce. You assign that Lead to a rep to qualify it. If that interest is qualified in any way, your rep then converts the lead and now you have an Account, Contact and Opportunity. (The Lead, by the way, essentially disappears into the ether.) This means that any reps qualifying inbounds end up owning a mix of Leads, Accounts, Contacts and Opportunities. This is a recipe for disaster which we'll get to shortly.

Now consider a relatively complex case. Let's say someone requests a demo - we'll call them Barb - under the email barb@acme.com. Acme, it turns out, is already a prospect and Barb has reached out before, so she's already a contact. The lead still gets created but if we just worked it directly we'd be stepping on the toes of another rep who's already engaged in a sales cycle. That's not great.

So what do we do? We have to build in some form of "matching" that tries to connect a Lead back to an Account so we can be sure that it's actually net new. We might look at the domain name in an email address, the company name, the person's name, their location or any other attribute and try to find some existing Account, Contact and/or other Lead record that already exists in the system. If we find multiple records that seem to match, we need to rank them so we have a way of picking the best. Oh and don't forget, this all needs to happen within your SLA so the customer doesn't lose interest.

It should be clear at this point that this quickly gets complicated as we attempt to do a 3-way fuzzy match while also conforming to RoE and whatever other territory design rules we've put in place. That's before we even consider the massive data quality implications that come from missed matches potentially causing duplicates to be created. That's the bad news.

Unfortunately there's no good news (yet). Only worse news. This complex case isn't special. It's got to be done for every lead every time because we can't even do the simple case until we've verified that there's nothing to match. The technical term for all of this is a "massive pain in the ass."

B2B reps should own accounts

We believe sales reps should own accounts - and only accounts. The reason is simple. Businesses are your customers, not individuals. And accounts are businesses. This seemingly simple statement has far reaching implications for how you construct your processes and your teams.

Of course a specific buying decision is usually made by an individual (or usually a committee multiple individuals) within a specific business. So we're not saying that you shouldn't build relationships with your influencers, decision makers, technical buyers, etc. In fact, if anything we're proposing you do more of that. These relationships are critical and necessary. After all, a person has to sign the contract. That single person is not, however, sufficient in a B2B context.

The reason personal relationships aren't sufficient is another simple statement: companies outlast an individual's role. Points of contact leave jobs, decision makers get promoted and aren't as focused on your solution; the list goes on. If you're going to maximize LTV, your relationship with the customer has to be able to survive these situations. This also applies to your organization. Your reps leave, you promote your people, your leaders come and go, but your business has to retain the businesses that are your customers. That's why we call this whole thing Business-to-Business.

The only way to generate lasting value for your business is to make sure it's built to maintain relationships and deliver value to other businesses over the long term in the face of changing individual roles. And that means you need to align your organization around accounts, not individuals.

Account based marketing

The ideas above aren't new insights. They're some of the basic concepts behind Account-Based Marketing (ABM). Despite the name, ABM isn't a marketing tactic. It got its start in the marketing world as a way to target businesses (instead of individuals) with ads but it has since evolved into a fundamental way of thinking about your GTM approach that means focusing on engaging, converting and retaining accounts that fit your ideal customer profile. You may prefer the term Account Based Everything (ABX).

Let's compare ABM to a more traditional transactional approach. In a transactional model, your job is to convince someone to purchase a product or service within a single sales cycle. You either win or you lose. ABM takes a broader view. You know which businesses *should be* your customers, so you engage them over time - often over multiple sales cycles. You tailor your messages to different folks within the account (the executives, the decision makers, the influencers) to ultimately build broad-based support for your solution. If you lose an individual sales cycle, you derive information you can use in future sales cycles. For example, the account's under contract with a competitor for the next 6 months, they just hired a new leader who's going to look at this issue in Q1, or the team is just too small right now to deal with this issue. That information makes your next outreach even better, even more targeted and, at some point, even more successful. To use a sports metaphor, the transactional model is about winning an individual game, while ABM is about winning the championship over the course of a season.

This all sounds great; who doesn't want to win championships? But there's just one problem. Reps can't talk to accounts. They talk to people. When somebody goes to your website and fills out that lead form, you just need to assign that lead to a rep to go call them, right?

Sort of. There's more to it than that.

Account ownership

Even if you get your matching reasonably right, you still have to solve for ownership. Do you assign it to the owner of the matching account? Convert the lead and notify the current owner? Ignore account ownership and allow a rep to run with an individual opportunity?

There are a lot of different solutions to this ownership problem, and they're probably addressed in your company's rules of engagement (RoE). One of the most common is to focus reps on owning leads and opportunities and mostly ignore the account. This almost always means that reps end up owning leads, accounts, and opportunities, however.

With this kind of messy ownership, reps struggle to have a unified view of their follow ups, managers can't track activity, and forecasting becomes much more complex because you have to continually build a single view of the world from different places. And most importantly, you're no longer focusing on the entity that actually buys from you: the account. You're also missing out on the strategic benefits of engaging with and gathering information on an account over the long term per ABM concepts.

The solution to all this is to make accounts your unit of ownership within your sales process. This has largely been true in the enterprise world and for account management.

It should also be the case for new business teams even in higher velocity B2B sales. If you do this, you get the benefits of ABM principles while also getting the benefit of a much clearer picture of rep books. This simplifies rep follow-up, activity tracking and forecasting. And, just one more time, it ensures your reps are aligned to your actual customer: the account.

So how do you make sure that you're only dealing with accounts? Treat leads as the account buying signal they are and immediately convert them (or relate them) to accounts. Then make sure those accounts are what get assigned. So let's talk about lead to account matching.

Lead-to-Account Matching

Sales leaders need to be sure a new inbound lead isn't related to an existing customer or prospect account. Every new lead must be evaluated against existing accounts through a lead-to-account matching process.

Messy CRM data contributes to sales collisions, process delays, and even lost revenue. You need to be able to quickly match new leads to accounts and route them to the right sales rep.

Since leads tend to be the highest volume of assignments in your funnel, and every second counts, you need to automate your routing, particularly with matching.

Start simple and focus on a lead's website domain. The overwhelming majority of your potential B2B customers have websites. Since web domains are guaranteed to be unique, they're about as close to a universal identifier for a company as you'll find. Not only does it apply to an account (via the website), it also associates an individual with a company via an email address. It's also free information and there are a ton of technical tools to help you work with them (including Gradient Works, ahem). Therefore, you should build the foundation of your matching by connecting the email address of a lead to the domain of an account.

To implement domain-based matching well, your data needs to be in a good place. We recommend the following steps:

- **Ask for it.** As much as possible, restrict inbound lead generation to require company email addresses and avoid public webmail addresses (e.g. gmail, yahoo, etc). If a lead isn't interested in handing over their work email address that's often a sign that they're either a) not actually very interested, or b) not a business. In our experience this is usually a negotiation between sales and marketing about quantity vs quality. Good luck.
- **Require it in CRM.** Ensure that a website or domain is specified for an Account in your CRM using data validation rules. If you can't require it, perform regular audits to ensure that they get populated.
- **Normalize it.** One challenge with "Website" fields is they tend to get populated with all sorts of information in all sorts of formats (e.g. with and without http/https, with and without extra path information). Add some code in your CRM to extract just the domain part of the URL (e.g. "https://example.com/1234" becomes just "example.com") or email address (e.g. "bob@acme.com" becomes "acme.com") and store it in a custom field on the Lead and Account. Some data enrichment tools will help automatically do this for you.
- **Clean up subdomains.** It's *usually* a good idea to remove any subdomains when doing normalization, especially "www" (e.g. "www.example.com" because just "example.com"). However, this can cause problems with some hosted subdomains (e.g. acme.squarespace.com or retailer.shopify.com). When in doubt, only suppress certain subdomains like "www".
- **Use domain aliases.** Many companies have more than one domain. This ranges from simple aliases (microsoft.com and microsoft.net) to regional domains (microsoft.com and microsoft.co.uk) to more complex scenarios like subsidiaries (microsoft.com and xbox.com). You should consider setting up a related list on the Account with domain aliases and checking that alias list when doing matching. How you want to match these will often be intimately related to how you decide to segment accounts (e.g. by global region, separate owners per subsidiary).
- **Make it unique.** Use duplicate detection rules in your CRM to make sure that accounts don't get created that have the same domain.

With clean data, you can then use various tools provided by your marketing automation, CRM, or other products in your tech stack to match a domain associated with a lead to a domain associated with an account. If there's a match, you can associate the lead information to the existing account. If there's no match, you can be reasonably confident in creating a new account and associating the lead information with that. From there, you can assign ownership of the account to a rep per your assignment rules.

Rep Assignment

One topic that consistently comes up is rep assignment - making sure that you've got the right kind of reps assigned to the right leads (and opps and accounts) at the right stage. It's a tricky thing that most high-velocity B2B revenue organizations struggle with.

Think of revenue organizations as a factory. Our goal is to create a system that converts raw materials (prospects) into a finished product (retained, revenue-generating customers) as quickly and efficiently as possible. To continue the metaphor a little further, modern revenue orgs often operate like an assembly line; prospects move in stages from one specialized team to the next, getting increasingly "finished" along the way.

Now, you don't actually assign a prospect or customer to a team. You actually assign it to a person: an individual human BDR, AE, AM, CSM or any of the other increasingly specialized roles we have on our revenue teams. This is where the factory metaphor gets a little more foggy. Each individual is different, and while they each follow the same process (hopefully!), it's their unique skills that ultimately drive the outcome at each stage.

This means that assignment is one of the core jobs of any revenue leader. It's your job to **decide who should be responsible for any given prospect or customer at any given stage in the sales process**. This ultimately sits at the core of org structure, capacity planning, goal setting and pretty much everything else you have to do as a revenue leader.

We typically start thinking about assignments in terms of territories and customer segmentation. (Territory design is a whole other topic for another paper later, but if you want more resources, read this about territory planning and this about why you should move past geographic territories.)

When your team is small, individual territories and segments are usually synonymous with an individual: e.g. “every lead that comes from New York is going to go to Sam because she owns the Northeast” or “this customer goes to Anil because he handles enterprise accounts with more than $100k ACV.” As you scale, you typically move beyond one territory or segment owner to a team of reps responsible for a particular segment. This leads to a new problem: once you’ve figured out a segment, which rep should get a given assignment?

That’s easy, you say… we’ll just hand them out equally, round robin style. This is a good start, but in practice it gets complex very quickly. What about the rep that hoards opps and has way more accounts in their name than everyone else? What about the rep that was on vacation and lost a week of assignments? What about the rep that’s just not performing who just got assigned the best lead to come in in ages?

In reality, almost everyone’s simple round robin usually becomes a weighted round robin (to give some people more turns in the queue) with some kind of manual order adjustment (to give a rep another turn or the next slot) and some level of capacity balancing (to not overload any given rep). We’ll talk more about round robin distribution shortly.

Assignment automation

Because of the nature of a typical sales funnel, there are fewer assignments to perform at each stage (e.g. there are going to be more leads than there will be opportunities qualified by a BDR). However, each of those assignments is typically more valuable because it represents an opportunity that’s farther along in the funnel (closer to finished in our factory analogy). This inversion of volume and value means organizations typically implement automations at the top of the funnel and then rely on manual assignment later in the funnel because they feel like it’s safer. Let’s call it “Automation in the Front, Spreadsheets in the Back”.

While understandable, this approach has big drawbacks.

First, it means that assignment processes are inconsistent. Automation forces you to clearly specify your rules while a manual process allows significantly more human error (or “judgement calls”) to enter into the process which will produce varying results.

Second, you typically give up data integrity and audit capabilities in the manual process. Looking up “who’s next” in a report or a spreadsheet and then flipping their ownership in

Salesforce means you don't have a single source of truth for what exactly happened when. That makes it hard to answer the inevitable question about why one rep got an assignment and another didn't.

Finally, manual processes don't scale. If you don't plan to increase the volume of deals in your funnel, then don't worry about this one. If you're in the process of scaling now, you should really be thinking about how that manual spreadsheet is going to work when you have 2x the number of SAOs being assigned by 2x the number of BDRS to 2x the number of AEs, or 3x the number of customers to 2x the onboarding reps and so on.

So with all these issues, what should an organization do?

- **Think holistically** - Don't just think about lead routing, setting demo appointments for AEs, etc in silos. They're all instances of the same basic problem. Try to solve them in similar ways. One specific situation that causes problems is doing lead routing in your Marketing Automation platform and all other assignments in your CRM. We recommend implementing all of it in CRM - this at least keeps your logic in one place.
- **Maintain an audit trail** - You're 100% guaranteed to have times where you need to explain why a particular assignment was made and to review the chain of ownership of a particular item. Make sure that whatever you do, you maintain an audit trail. If you use Salesforce, at a minimum you should turn field history tracking on for assignment-related fields.
- **Automate** - Even if it's easy to use a spreadsheet now, it'll lead to inconsistencies now and it'll get harder in the future. Salesforce offers great tools like Flow to help you automate all kinds of tasks. This forces you to be explicit about your rules and helps drive consistent results.
- **Learn from others** - While round robin assignment eventually turns into something more complex, you can get started with a basic version fairly easily.

When your organization gets to the point that you have a) leads coming in on a regular basis and b) multiple sales reps, you quickly realize you need a way to divide up those leads among your reps. If you're using Salesforce, you may have noticed a built-in feature called Lead Assignment Rules. Sounds perfect, right? Sort of. We'll walk you through it next. Skip this section if you don't use Salesforce.

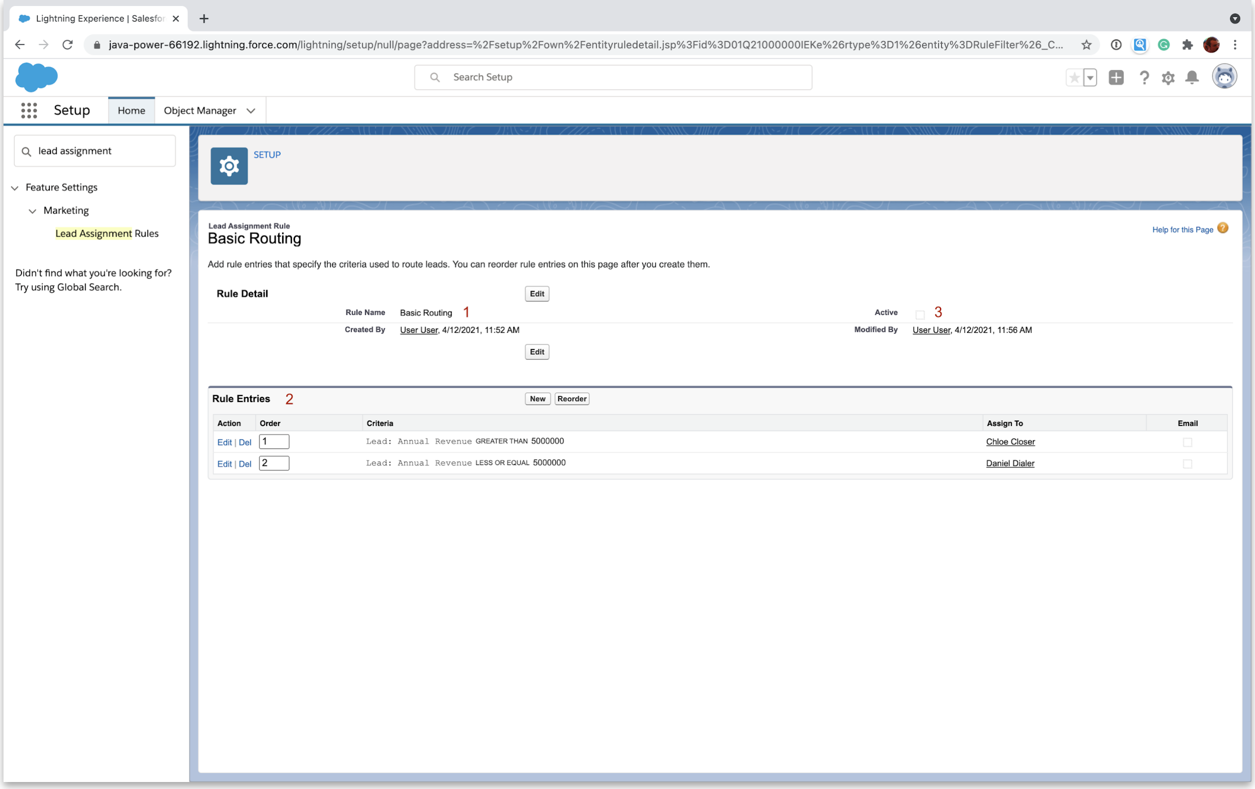
Lead Assignment Rules in Salesforce

Here's our agenda for the lead assignment section:

- The basics of lead assignment rules
- Matching and assigning with rule entries
- Issues and challenges with lead assignment rules

The basics of lead assignment rules

Let's take a look at lead assignment rules in Salesforce. Go to your Salesforce org's Setup section and search for "lead assignment" in the Quick Find box. You'll find these buried under Feature Settings > Marketing > Lead Assignment Rules. If you don't see it, make sure you have permissions for "View Setup and Configuration" and "Customize Application".


Lightning Experience | Salesfor
java-power-66192.lightning.force.com/lightning/setup/null/page?address=%2Fsetup%2Fown%2Fentityruledetail.jsp%3Fid%3D01Q21000000IEKe%26rtype%3D1%26entity%3DRuleFilter%26_C...
Search Setup
Setup
Home
Object Manager
lead assignment
Feature Settings
Marketing
Lead Assignment Rules
Didn't find what you're looking for? Try using Global Search.
SETUP
Help for this Page
Edit
New
Reorder

Lead Assignment Rule

Basic Routing

Add rule entries that specify the criteria used to route leads. You can reorder rule entries on this page after you create them.

Rule Detail

Rule Name	Basic Routing 1	Active	□ 3
Created By	User User, 4/12/2021, 11:52 AM	Modified By	User User, 4/12/2021, 11:56 AM

Rule Entries 2

Action	Order	Criteria	Assign To	Email
Edit | Del	1	Lead: Annual Revenue GREATER THAN 5000000	Chloe Closer	□
Edit | Del	2	Lead: Annual Revenue LESS OR EQUAL 5000000	Daniel Dialer	□

Let's start with some basics:

- **Unique names** - Every lead assignment rule has a unique name [1].
- **A lead assignment rule is really a list of "rule entries"** - Salesforce calls each of the individual rules in the list a "rule entry" [2]. Each rule entry allows you to say something along the lines of: "if a lead meets these criteria, assign it to this user” (or queue - more on that in a moment).
- **Rule entry order matters** - The list of rule entries [2] is processed in a specific order you define. Salesforce will process each rule entry until it finds a match. Once it finds a match, it will assign the lead based on how the rule entry is configured.
- **Only one active rule at a time** - You can only have one lead assignment rule set to active [3].

Now that we've got a decent idea about what a lead assignment rule looks like, let's dive into rule entries.

Matching and assigning with rule entries

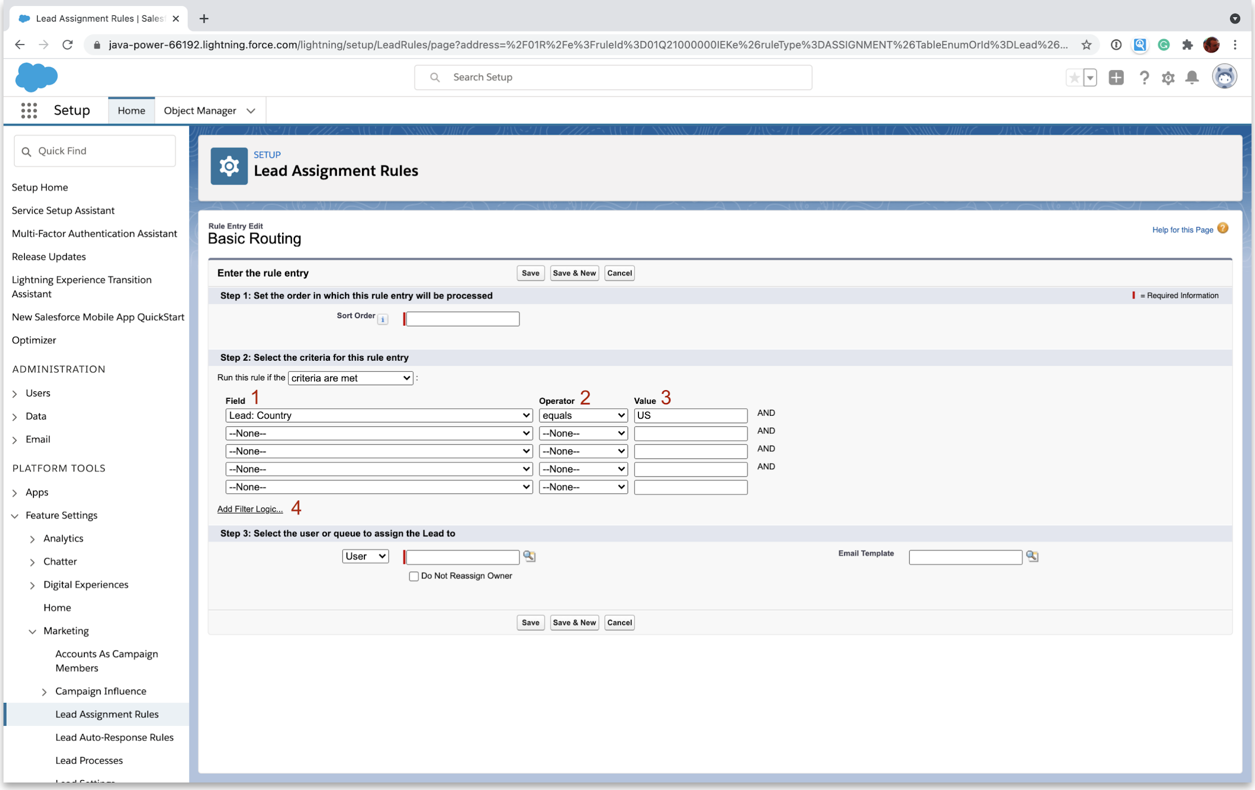
Every rule entry has three parts:

- **Order** - This is a number that indicates where the entry exists in the list. Rule entries are evaluated starting at 1 and then processing until one of the rules matches.
- **Matching Criteria** - Leads are compared to the matching criteria for each rule to determine if the lead matches. The first rule that matches is used to determine assignment.
- **Assignment** - This tells Salesforce what you want to do once a lead has matched the matching criteria. Usually, you'll tell Salesforce to assign it to a user or a queue.

Order is pretty straightforward. The complexity really lies in matching criteria and assignment. Let's spend some time on those.

Matching criteria

There are two types of matching criteria: filter criteria and formulas. You'll probably use filter criteria the most, so let's start with that one:


Lead Assignment Rules | Sales
java-power-66192.lightning.force.com/lightning/setup/LeadRules/page?address=%2F01R%2Fe%3FruleId%3D01Q21000000IEKe%26ruleType%3DASSIGNMENT%26TableEnumOrId%3DLead%26...
Search Setup
Setup
Home
Object Manager
Quick Find
Setup Home
Service Setup Assistant
Multi-Factor Authentication Assistant
Release Updates
Lightning Experience Transition Assistant
New Salesforce Mobile App QuickStart
Optimizer
ADMINISTRATION
Users
Data
Email
PLATFORM TOOLS
Apps
Feature Settings
Analytics
Chatter
Digital Experiences
Home
Marketing
Accounts As Campaign Members
Campaign Influence
Lead Assignment Rules
Lead Auto-Response Rules
Lead Processes
SETUP
Help for this Page
Save
Save & New
Cancel
1
2
3
4

Lead Assignment Rules

Rule Entry Edit

Basic Routing

Enter the rule entry

Step 1: Set the order in which this rule entry will be processed

= Required Information

Sort Order

Step 2: Select the criteria for this rule entry

Run this rule if the criteria are met :

Field	Operator	Value	
Lead: Country	equals	US	AND
--None--	--None--		AND
--None--	--None--		AND
--None--	--None--		AND
--None--	--None--		

Add Filter Logic...

Step 3: Select the user or queue to assign the Lead to

User

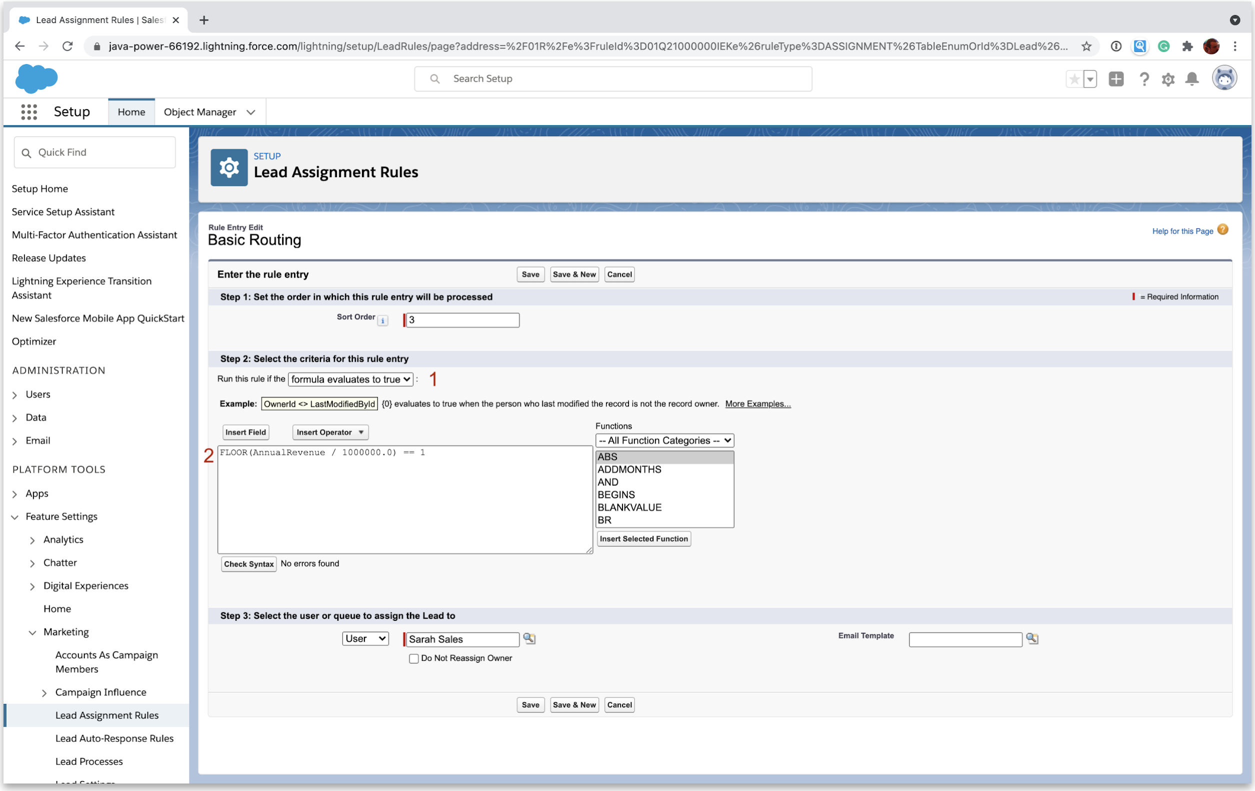
□ Do Not Reassign Owner

Email Template

Filter criteria work a lot like a classic Salesforce report. You specify a field [1], an operator [2] and a value [3] for matching purposes. Unless you use custom logic [4], all the criteria must be true before this rule will match. In the example above the "Country" field on Lead must exactly equal "US" in order to match.

One key limitation is that you can only configure criteria using fields from the lead, the current user and the campaign associated with the lead.

Here's what it looks like to create a formula rule entry:



Rule Entry Edit

Basic Routing

Enter the rule entry

Step 1: Set the order in which this rule entry will be processed

▌ = Required Information

Sort Order: 3

Step 2: Select the criteria for this rule entry

Run this rule if the formula evaluates to true:

Example: OwnerId <> LastModifiedById {0} evaluates to true when the person who last modified the record is not the record owner. More Examples...

```
FLOOR(AnnualRevenue / 1000000.0) == 1
```

No errors found

Step 3: Select the user or queue to assign the Lead to

User Sarah Sales

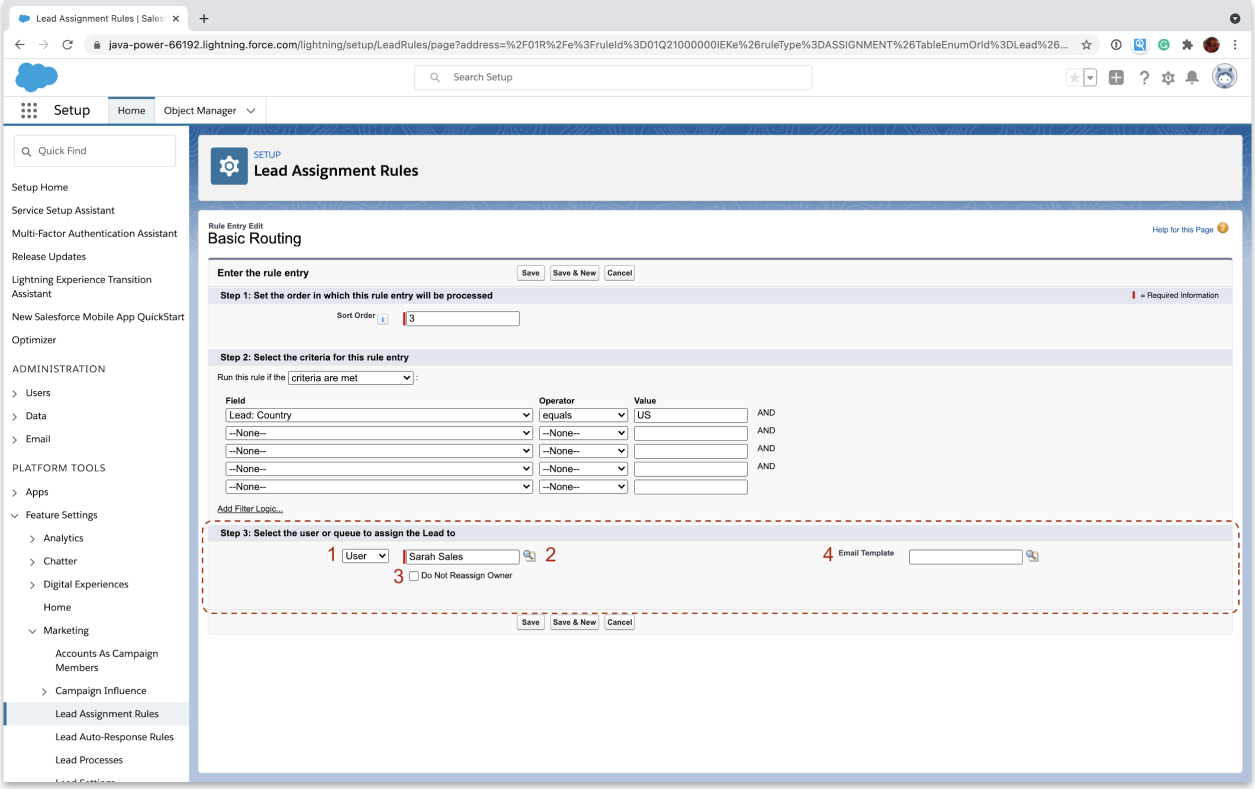
□ Do Not Reassign Owner

Email Template:

Instead of using filter criteria, this rule uses a formula [1]. Formulas can get pretty complex. The one here will evaluate to true if the "AnnualRevenue" field of the Lead is at least $1MM but less than $2MM. Just remember that your formula must evaluate to true/false. Formulas allow for a lot of flexibility but can be hard to debug, especially if you have a lot of rule entries.

Assignment

Let's go back to the rule entry screen and focus on the assignment section:



Rule Entry Edit

Basic Routing

Enter the rule entry

Step 1: Set the order in which this rule entry will be processed

= Required Information

Sort Order: 3

Step 2: Select the criteria for this rule entry

Run this rule if the criteria are met :

Field	Operator	Value	
Lead: Country	equals	US	AND
--None--	--None--		AND
--None--	--None--		AND
--None--	--None--		AND
--None--	--None--		

Add Filter Logic...

Step 3: Select the user or queue to assign the Lead to

1 User Sarah Sales 2

3 □ Do Not Reassign Owner

4 Email Template

Before we go any further let's clarify what we mean by assign. In this case, it means we're going to set the Owner field on the particular Lead record to either a User or a Queue. Depending on which type of object you choose [1], you will need to specify the specific user or queue you want to use [2].

So, what's the difference between a user and a queue?

- **User** - In business terms, this is a particular sales rep who will be responsible for managing the lead. In technical terms, this is an active Salesforce user configured in your Salesforce org.
- **Queue** - A Queue is a list of records (usually Leads, but not always) that are accessible to a specific group of users. Each of those users can take ownership of a record from the queue, removing it for the others. When you assign ownership of a Lead to a Queue, you're effectively making it first-come, first-serve for a group of users. This is **not** the same as a round robin assignment where you assign the Lead to one person from a list of users.

There's one last thing to know about assigning ownership. Note the checkbox labeled "Do not Reassign Owner"[3]. This controls what happens when a lead is updated and matches the criteria. If this is checked, the assignment will essentially be ignored. Use this to prevent toggling users back and forth.

The last field, "Email Template" [4] allows you to specify a particular email template to use for notifying the new assignee when the assignment is complete.

Lead assignment issues and challenges

Lead Assignment Rules are a reasonable choice for a small team with simple rules. They work and the only cost to you is the administrative overhead of creating and maintaining them. Unfortunately, they're extremely limited and don't cover many common use cases. They also become very brittle as your team size and lead velocity increase. Here are a few common issues and challenges:

- **Leads only** - They only apply to the Lead object (there are Case Assignment Rules for Cases, but that's it). This may make them a poor fit for your sales process if you need to do lead-to-account matching or you're pursuing an account-based strategy. If you need to assign other things besides Leads, you'll need to try a more flexible automation solution like Flow.
- **Hard to test** - Unlike Process Builder or Flow, your Lead Assignment Rules don't provide any form of versioning or debugging so it's hard to test them without just doing it live.
- **Hard to audit** - There's no record of *why* a lead was assigned in a particular way. You'll have to go look through your rule entries to figure it out. This can quickly get painful as the number of rules increases.
- **One rule at a time** - If your company has very different rules for different leads (e.g. from different campaign sources or applicable to different product lines), it can be very challenging to craft your rule entries in a way that handles more than one lead routing flow.
- **No round robin** - It's technically possible to do a very poor version of round robin with Lead Assignment Rules, we don't recommend it. It's extremely brittle and will break without lots of maintenance.

If you've got a small team with simple processes, give Salesforce lead assignment rules a try; they may give you just what you need. If you find yourself needing to assign other objects besides Leads, perform round robin rep assignment or maintain a full audit trail, you might want to consider Gradient Works Assignment.

Round Robin Distribution

Round robin assignment is familiar to most sales teams. It's a deceptively simple concept that often becomes more complex in practice. We'll take you through the key concepts you need to supercharge your round robin approach.

Round robin distribution or assignment is a fundamental way to distribute ownership of leads, accounts, contacts, cases or anything else to a set of sales (or support) reps.

While round robin assignment is most common for inbound lead routing, you can use it anywhere you have a group of users that need to receive work from a central list. This paper is about leads, so we'll talk specifically about leads in the rest of this section, but remember it applies to anything you want to assign, at any stage of your customer lifecycle.

While round robin assignment is fair in that it distributes an equal number of leads to each rep in a group, it doesn't always give you the outcomes you're looking for. This means sales organizations almost always use a modified form of round robin in practice.

Let's start with the basics and then move on to the more advanced techniques.

Round robin basics

Forget about leads and sales reps for a minute and let's use something a bit more tangible: a card game.

Most of us have played a card game with a dealer who hands out cards. The dealer starts with one player, gives that player a card and then goes around the table, handing each player a card until some number of cards have been given out. This is a good way to evenly divide a large number of things (cards) among a small number of people (players). That's basic round robin.

Let's apply this to sales. The cards are leads and the players are sales reps. Pretty simple right? There's just one change we need to make. Instead of having a limited set of cards that we hand out all at once, imagine our dealer has a machine running 24/7 that periodically hands them a card to distribute. This is more like how inbound leads work. From there the process is the same. Keep track of the last rep to get a lead, then hand the new lead to the next player around the table. Rinse and repeat, forever.

Card games don't stop when you deal the cards (obviously). Playing the game usually involves players using up the cards in their hand and getting assigned new ones, until there are no cards left. Round robin assignment in sales is similar. We assign leads to reps and those leads get disqualified, they close, they go quiet, and so on. The difference is that, unlike a card game, we a) never stop playing the game since there are always new leads, and b) don't control when a new lead becomes available. Over time, players (reps) can end up with very different hands (books).

This causes several challenges:

- How do you handle a ramping rep who is just getting started and doesn't currently have anything to work? Essentially you have a game where new players with no cards can join at any time.
- What happens if some reps have more leads than they can reasonably work? Imagine a card game where one player has 20 cards in their hand and another player has 5. Do you keep handing out new cards equally to both?
- What happens when a rep isn't available? Maybe they're out sick or taking a vacation. This is like a player getting up and leaving the table for a little while; you probably don't want to just keep handing them cards.

Weighted assignment

Let's go back to the card game analogy. We assumed that every time we went around the table, we gave one card to each player. It doesn't have to be that way. Imagine if one player was special and got dealt two cards every time around. This approach is known as *weighted round robin* (in the same sense as a weighted average).

To fully generalize this idea, imagine that at the start of every round of distribution, each player has a counter of how many cards they're supposed to receive. Each time the player gets a card, their counter goes down by one. If their counter is 0, they don't get a card. The dealer continues dealing cards until each player's counter is at 0. Once that happens, everyone's counter resets and the process starts over.

Why all the complexity with rounds and counters and such? The reason comes back to our card machine that periodically produces cards. If you're dealing with something like leads that come in randomly one at a time, you generally want to avoid handing a whole bunch of consecutive leads to the same rep. Doing so wouldn't be particularly fair because you're starving the other reps for some period. It's also not optimal because you might overload the lucky rep who's getting several leads at once.

Weighting has many real-world uses. It can help a rep who's behind catch up to other reps without totally unbalancing the distribution. Teams often use this approach to help certain reps build pipeline when ramping or if they've been out for a while. Weighting can be a good way to reward certain reps or ensure that senior reps get more opportunities than more junior reps.

Take capacity into account

As we assign leads to reps, some leads enter sales cycles and some get disqualified. Over time, different rep behaviors (and some luck) will lead to some reps having too much and some reps not having enough.

One way to handle this is to take capacity into account. This kind of capacity-aware assignment can really take two forms: load-balancing (which really isn't round robin at all) and capping.

Load-balancing is about ensuring every rep has the same amount of stuff to work (their load). If we go back to our card game analogy, imagine if the dealer always gave the next card to the player with the fewest cards in their hand. That would be a fully load balanced approach. It may not be ideal, however, because it has the same issue as weighted round robin in which you might give a lot of consecutive cards to a single player.

With a simple load balancing approach, think about what would happen when a ramping rep suddenly gets added into the rotation. You'd immediately hand every assignment to them until they had as many assignments in their book as the next lowest rep. If you do that, you'll end up with some unhappy reps.

A variation on this approach is capping. You use the normal round robin distribution approach (weighted or otherwise) but you also introduce a maximum capacity. If any rep's load is at capacity, you skip that rep in the distribution until they have available capacity. (Clever readers will realize this is like setting their weight in the round robin to 0.)

One challenge with capping is that you need some way to deal with the scenario in which every rep is at capacity. In this case, you could run into a scenario in which no one gets a lead which is not ideal. Often, it makes sense to institute some kind of fail-safe assignment to a manager.

It's important to realize that in both of the above scenarios it's not necessary for the load or maximum capacity to just be a count of leads or even a fixed number that's the same for every rep. It could be defined as a potential book value or even a multiple of each rep's quota.

Capping is an under-utilized tool in the round robin assignment toolkit. It can lead to more balanced outcomes while avoiding the issue that assigns a bunch of leads to one rep.

Handle rep availability

When one of the players gets up and leaves the card table, it doesn't make sense to keep handing them cards. The same is true if your reps are out sick or on vacation. Any good round robin implementation needs to take this into account and allow certain reps to be temporarily removed from the rotation.

Another availability consideration is working hours. This is especially important in global organizations that use a follow-the-sun model to ensure they're responsive at all hours. While a full round-the-clock model is more common in customer support or IT, the basic concept often shows up in organizations that try to cover a lot of time zones. For example, a sales org might have East Coast reps handle inbounds from 8am to 5pm ET and West Coast reps handle inbounds from 11am to 8pm ET (8am to 5pm PT). In short: East Coast handles the early morning, both handle the midday period and the West Coast handles the evening. A full round robin implementation should be able to handle this kind of approach.

Manual skipping

We have yet to see a round robin assignment implementation where the need to manually skip a rep didn't come up in some way. No matter how pure and optimized your assignment method is there will always be a case when a manager wants to skip a rep.

The reason for this is simple: unexpected things happen. Managers might skip a rep if they've recently assigned them a discretionary lead (e.g. as a result of a rules of engagement judgment) or, more rarely, as a disciplinary action.

A flexible round robin implementation will allow for skips in certain circumstances. It's always a good idea, however, to document skips and the rationale for them since one of the main purposes of a round robin approach is to fairly and automatically allocate leads. It may be necessary to explain the decision later.

Additional round robin thoughts

As you move to more advanced forms of round robin, make sure you can always explain to a rep the rationale for why they received (or didn't receive) a particular lead. We highly recommend that you maintain a log of assignments logic, an audit trail of assignment decisions, and some kind of process documentation. These will come in very handy if you need to explain the system's behavior.

Most organizations start with some form of basic round robin and quickly find they need to handle the situations described above. We also find that once a round robin capability is in place, most organizations see opportunities to use it beyond leads and start automating distribution throughout their customer lifecycle.

Implementing Round Robin in Salesforce

Fairly assigning leads to sales reps is core to how sales teams manage ownership. In many cases that means round robin assignment in Salesforce.

Before you implement round robin assignment in Salesforce, make sure to think about what you're trying to achieve. For example, some of the options below (e.g. Lead Assignment Rules) are only applicable to Leads and won't work for Accounts, Opportunities, Tasks, etc. If you're trying to assign qualified opportunities to account executives (AEs) you'll need to make sure you pick the option.

You should consider whether or not you want to do the basics or if your round robin needs weighting, capping, load balancing or availability management. Just be aware that the methods below will focus on the most basic form of round robin.

The rest of this section will cover the following methods for doing round robin assignment in Salesforce:

- Lead Assignment Rules
- Process Builder
- Apex
- Flow
- Using sales engagement software
- Bonus method using Gradient Works Assignment

But first, some math and an example involving a deli...

Most of the examples below use the MOD Salesforce formula operator in some way. Mod is short for the "modulo operation" which sounds complicated but is actually pretty simple: it just gives you the remainder left over after you divide two numbers. For example:

- MOD(5,3) is 2 because 5 divided by 3 has a remainder of 2
- MOD(9,3) is 0 because 9 divided by 3 has no remainder (3 goes evenly into 9)
- MOD(2731,3) is 1 because 2731 divided by 3 has a remainder of 1 (3 goes into 2731 910 times if you're curious)

The neat part about the mod operation is that it's always a number between 0 and 1 less than the number you're dividing by. So if you take any number mod 3, the result will always be either 0, 1 or 2. Ok, maybe that could be useful... but how? Glad you asked! Let's see how we can use MOD to evenly assign something.

Imagine you run a deli and you've got 3 sandwich artists. The artists get paid $0.50 per sandwich so you want to make sure each gets an equal number of customers throughout the day or else somebody will make less money and be upset.

The good news is that every customer that walks in takes a number from the little ticket dispenser by the door. Each new customer's number is 1 more than the previous customer's number. So how do you turn this system into a way of allocating customers? Here's a little algorithm:

1. Write the names of each sandwich artist on a board and give them each a number between 1 and 3.
2. Take each new customer's ticket number and MOD it by 3 (the number of sandwich artists). This gives you a remainder between 0 and 2.
3. Add 1 to the remainder so you have a tidy result between 1 and 3.
4. Use the resulting number between 1 and 3 to pick that customer's sandwich artist from the board.

Putting it all together, let's say your new customer takes a ticket with the number 8. You do 8 mod 3 and get 2. Add 1 and you get 3. Pick sandwich artist number 3 and send them this new customer. The next customer pulls a 9. 9 mod 3 is 0. Add 1 and you've got, well, 1. Send them on to sandwich artist one. Repeat.

I don't know about you but those customers sound suspiciously like Leads (or really anything else you'd want to assign) to me. It turns out that most of the solutions below use a variation on this where the ticket dispenser machine is replaced by an auto number field.

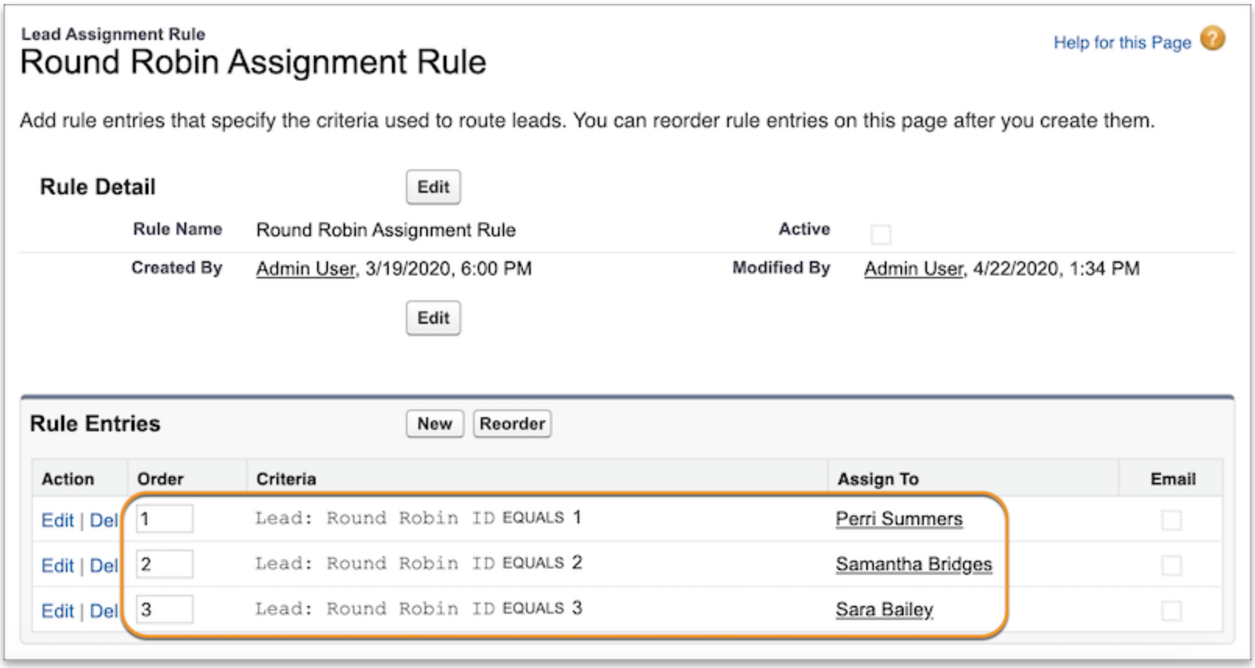
1. Round robin with Lead Assignment Rules (LARs)

We've done a deep dive on Salesforce Lead Assignment Rules in another post so we won't repeat everything here. However, if you've done any googling at all for "round robin lead assignment in Salesforce" you've probably encountered this Salesforce Help doc about creating round robin lead assignment rules.

The basic idea in the Salesforce Help doc is precisely the deli counter example above:

1. Add an auto-number field to Lead that counts up every time there's a new lead
2. Add a formula field on Lead to calculate the "Round Robin Id" by MODing the Lead number by the number of sales reps and adding 1
3. Hard-code Lead Assignment Rule Entries matching "Round Robin Id" to a rep

Step 3 looks like this:



Lead Assignment Rule

Round Robin Assignment Rule

Add rule entries that specify the criteria used to route leads. You can reorder rule entries on this page after you create them.

Rule Detail

Rule Name	Round Robin Assignment Rule	Active	□
Created By	Admin User, 3/19/2020, 6:00 PM	Modified By	Admin User, 4/22/2020, 1:34 PM

Rule Entries

Action	Order	Criteria	Assign To	Email
Edit | Del	1	Lead: Round Robin ID EQUALS 1	Perri Summers	□
Edit | Del	2	Lead: Round Robin ID EQUALS 2	Samantha Bridges	□
Edit | Del	3	Lead: Round Robin ID EQUALS 3	Sara Bailey	□

Let's break down the good, the bad and the ugly of this approach.

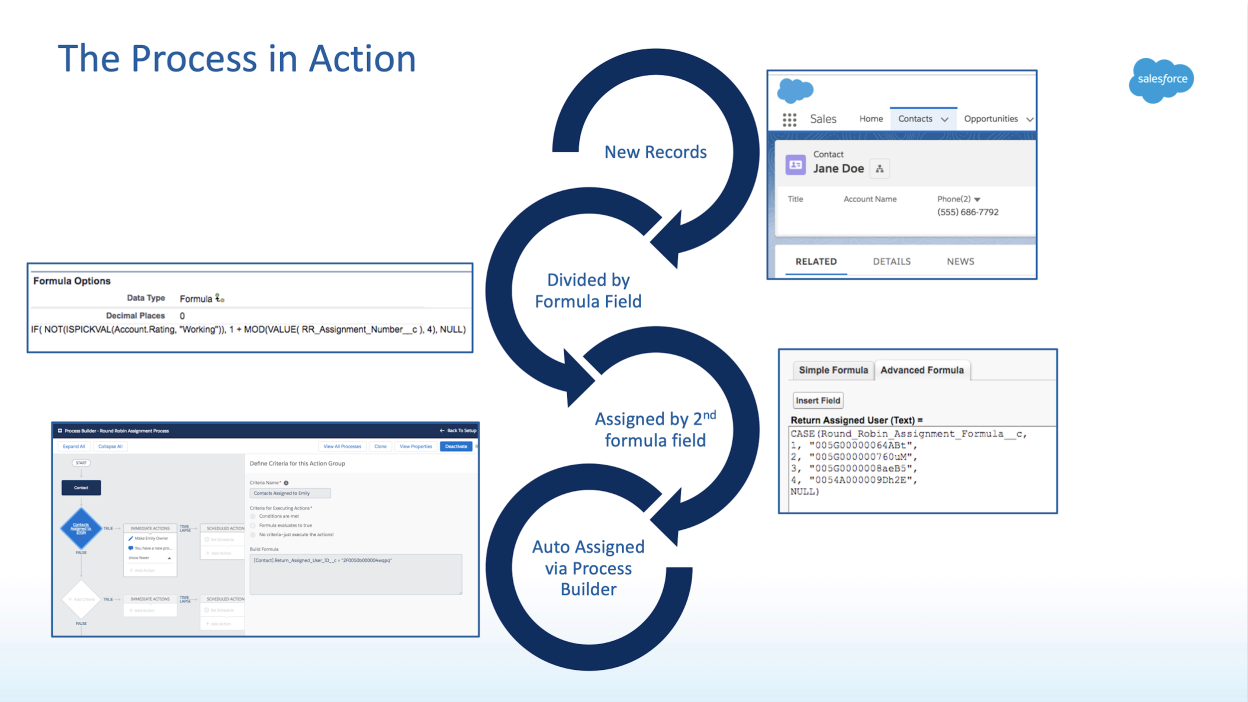
- **The good**
  - It's pretty simple with no complicated automation
  - It's built-in so it comes with no strings attached (besides Salesforce license $$)
- **The bad**
  - Leads only - Lead Assignment Rules only work for - you guessed it - leads
  - Hard to manage - Whenever reps get added, leave the company or just go on PTO, you have to update both the *Round Robin Id* formula to change the number of reps **and** update out the rule entries to assign *Round Robin Ids* to different reps
  - No audit trail - However, you can help by tracking field history for the owner field.
- **The ugly**
  - The minute you have any additional routing rules where some leads are created but aren't assigned to this group, your assignment gets skewed. Here's an example. Let's say 4 leads are created and given numbers 1, 2, 3 and 4. Let's say 2 and 3 aren't assigned because they used a gmail address. What happens when Leads 1 and 4 get assigned? They'll both get assigned to the rep in slot 2

(MOD(1,3)+1 = 2, MOD(4,3)+1 = 2). Reps 1 and 3 aren't going to be happy about that.

2. Round robin with Process Builder

Note: Salesforce is deprecating Process Builder in favor of Flow soon, so this won't work much longer.

Emily Douglas from Formstack discussed how to use Process Builder for round robin assignment at Dreamforce 2018. This solution is nice and flexible because it can be applied to any object. In fact, her example uses Contacts. Here's her overview slide:



The Process in Action

Formula Options

Data Type: Formula

Decimal Places: 0

```
IF( NOT(ISPICKVAL(Account.Rating, "Working")), 1 + MOD(VALUE( RR_Assignment_Number__c ), 4), NULL)
```

Return Assigned User (Text) =

```
CASE(Round_Robin_Assignment_Formula__c,
1, "005G00000064ABt",
2, "005G000000760uM",
3, "005G0000008aeB5",
4, "0054A000009Dh2E",
NULL)
```

If you watch the video and look closely, you'll see it's basically the same deli counter process used by the Lead Assignment Rule approach. The main difference here is that the last step happens in Process Builder which is much more flexible than Lead Assignment Rules and isn't tied specifically to the Lead object.

This means that the good, bad and ugly of this approach is pretty similar to the earlier approach.

- **The good**
    - Built-in - You don't need anything besides Salesforce

  - Any object - The pattern can work for any object, not just Leads. All you have to do is set up the custom fields on the object and set up a process builder to operate on objects of that type
  - Versioning - Process builder maintains versions as you make changes so at least you can refer back to the way things used to be if you make a tweak. Just note that this doesn't apply to the formula fields, only the logic in process builder
- **The bad**
  - Duplication - This approach describes a pattern you can use for any given object with some custom fields and a process builder. You'll need to try to keep that pattern consistent across every object you build this for.
  - Hard to manage - As with the LAR approach, you've got to manually edit a formula field to change the list of reps
  - More workflow effort - Unlike the LAR approach, you have to explicitly build in logic in process builder to do the assignments
  - Process Builder is on its way out - Salesforce has publicly stated that Flow is the future of workflow automation on the platform, not Process Builder. It's not going anywhere soon but it's not being improved.
- **The ugly**
  - Skewed assignments - Just like the LAR approach, this relies on a single deli-counter style auto number field. This will skew in the exact same way if you start implementing more complex routing rules. It is *possible* to eliminate some of these issues in process builder, but at the expense of a lot more complexity.

3. Round robin with Apex

As you probably know, developers can build entirely custom logic on the Salesforce platform using the Apex programming language. A typical Apex solution for round robin assignment usually takes the form of one of these two approaches:

- A variation on the deli-counter approach we've discussed so far, using a combination of formula fields to compute an auto-number value, the Apex version of the modulo operation (Math.mod) and triggers to kick off the code. This approach has all the drawbacks of the LAR and Process Builder approach.

- A more flexible (and powerful) approach that uses a set of custom objects, each of which define slots in multiple round robin queues. At the expense of a lot more complexity, this approach allows you to distribute the same type of object fairly to different groups based on any routing logic you can imagine.

A full Apex implementation is too complex to discuss here so let's just look at the good, the bad and the ugly of using Apex to do round robin:

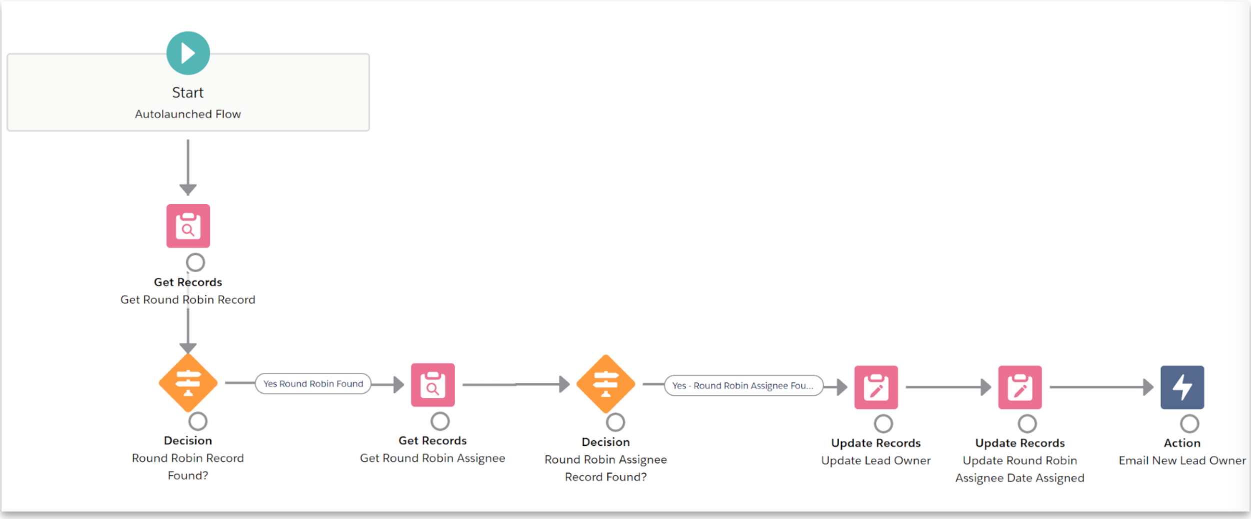
- **The good**
  - Nearly infinite flexibility - If you can describe it, you can most likely do it with Apex. Assigning any object, with any logic is entirely possible.
  - Round robin can coexist with other routing rules - It's entirely possible to build complex routing rules to different groups of users without skewing your round robin
  - Version control - Modern development approaches mean you can keep track of and independently test different versions of the code.
- **The bad**
  - Developers! It's code. You'll need a developer to write it in the first place and you'll need a developer to maintain it, debug it and change it.
- **The ugly**
  - Maintenance and change will be difficult. In my experience most code solutions in Salesforce become very hard to maintain over time, especially when the original author leaves. Even if you do have developers to support the custom solution, they're often very busy with other priorities and can't make changes on a time schedule that the business leads.

4. Round robin with Flow

We've talked a lot about Salesforce Flow around these parts because it's the future of Salesforce automation. The important thing to know about Flow is that it provides nearly all the flexibility of Apex with the low-code approachability of Process Builder. That makes it an ideal way to build solutions like round robin assignment.

There are several good examples of using Flow for this. You can find one in the book Lightning Sales Ops by Matt Bertuzzi but I'm going to specifically focus on this one by the sales operations consultants Kicksaw.

Here's a screenshot of Kicksaw's Flow in action:



The Kicksaw solution has a lot of moving parts but it gets rid of the "deli-ticket" skew problem that we discussed above. It introduces separate groups of reps with different territories so it can be used with more complex routing logic. It also always assigns to the rep who's gone the longest without receiving an assignment. While this won't allow for round robin assignment with weighting and capping, it will do the job for basic round robin quite well.

Our main issue with the Kicksaw solution is that it introduces an unnecessary process builder component to trigger execution of the Flow. That's easily done with record-triggered Flows.

- **The good**
  - Flexibility - Flow gives you the flexibility to do most things you can do in Apex
  - Maintainability - Admins can update the Flow logic and change the various user lists without needing any developer help
  - Routing rules without skew - Support multiple territories and routing logic without worrying that it will skew your assignment process
  - Future proof(ish) - Flow is the (current) future of automation on Salesforce
- **The bad**
  - More logic to deal with - Flow can require sorting out some tricky logic similar to programming; not all admins are comfortable with that

    - A lot to configure - The full Kicksaw solution requires some custom objects, a queue and even some Lead Assignment Rules in addition to Flow. That can be a lot.
- **The ugly**
    - It's all on you - In the end this solution is custom to your Salesforce and relies on you and your team to maintain it.

4.5. Round robin with sales engagement software

We're throwing in a 4.5 here because these recommendations aren't strictly Salesforce but instead might give you the ability to do basic round robin assignment with tools you already have. Sales engagement tools like Outreach and Salesloft have some basic round robin capabilities baked in so you may be able to get away with using them.

- **The good**
    - You're already paying for it
- **The bad**
    - Functionality is super limited and generally only applies to a few use cases
- **The ugly**
    - If you get rid of your sales engagement product, you'll also lose any round robin functionality you're using

Conclusions

Thank you for reading our very (very) long guide to lead distribution! While we talked a lot about leads specifically, many of these concepts can be applied to opportunities and accounts, as well. If you have any questions at all, please reach out to us at www.gradient.works.

And if you want to talk about automating your lead distribution (or opp or account distribution), consider Gradient Works. We've built flexible lead routing software that will help you replace your existing tools, manual processes, and complex custom code. Request a demo here. Let's set up a free trial to show you how it works.

Gradient Works