



Rules in SalesForce

There are countless different product variations, with many potential buyers that have infinite characteristics. Every salesman thinks: the way I have success in sales is unique to my experience and my product. Expensive failures of CRM software implementations are used to support that claim.

SalesForce, now number one in CRM software market shares, changed the general opinion.

Disruptive? Yes, because it's cloud based with an out-of-the-box sales process based on best practices that can be customized by end-users with a low priced base model.

I did my own implementation, was up-and-running in a couple of days, had sales, marketing and support work together and could daily ask for an up-to-date report.

SalesForce is a standard product to support a sales product. There is a product and someone interested. It's a lead. The lead turns out to be working for a larger organization. It's a contact. Contacts from the same organization show interest? It's an account. The interest is real? It's an opportunity... and at some point this results in an order that must be supported. That is SalesForce in a nutshell with all fat printed words being SalesForce objects with standard behavior.

This is the next [column](#) to assess the use of business rules in Commercial Of The Shelf products (standard software) to enhance the adaptivity of the product for personal usage.

The standard behavior can be changed by adding your own data-fields and rules. You need the shoe-size of all your contacts? Possible: add your own field. Assign Dutch leads to Dutch sales? Possible: add an assignment rule. Female contacts with shoe-size greater than 42 must get a special offer? Can be done, with an auto-response rule.

The rules in Salesforce have a slightly different form depending on the rule type.

- Validation rule: raises an error message for a field specified as error location WHEN the error condition formula is true.
- Assignment rule: assigns a lead or case to a specified user WHEN criteria are met OR a formula evaluates to true.
- Auto response rule sends a message based on a specified template WHEN criteria are met OR a formula evaluates to true.
- Alert rule sends a message to one or more Salesforce users WHEN an opportunity is for an amount within specified thresholds.

Workflow rules, escalation rules and default rules use similar strategies. So Salesforce uses a template based way to edit different kinds of rules and has generalized the way conditions are specified in two ways: as a formula and as a list of criteria. Let's have a closer look at the conditions.

Formulas use operators, fields and functions and evaluate to TRUE or FALSE. The user is guided with an editor or just types in code that looks like this:

```
AND(
```

```
/*competitor field is required, check to see if field is empty */
```

```
LEN(Competitor__c) = 0,
```

```
/* rule only enforced for ABCD record types */
```

```
RecordType.Name = "ABCD Value",
```

```
/* checking for any closed status, allows for additional closed picklist values*/
```

```
CONTAINS(TEXT(StageName), "Closed")
```

```
)
```

By including this example I may have lost half of my audience (according to a famous quote by Stephan Hawkins about equations in presentations).

This is not business friendly.

I associate the brackets in the formula with functional programming languages but it is probably inspired by Excel formulas. Compared to Excel it is nice that comments and formatting are allowed.

Criteria are easier to use and offer a subset of the semantics in formulas. The editor looks like a table where you choose a field, an operator and a value for each criterium (row).

	Field	Operator	Value	
1.	Lead: Rating	equals	Hot	
2.	Lead: Lead Source	not equal to	Partner	
3.	Lead: Lead Source	not equal to	Conference	
4.	--None--	--None--		
5.	--None--	--None--		

[Add Row](#) [Remove Row](#)

[Clear Filter Logic](#)

Filter Logic:

1 AND (2 OR 3) [Tips](#)

All criteria must be met for the condition to evaluate to TRUE unless you specify Filter Logic. The filter logic references the row numbers and combines it with logical operators (AND / OR) and brackets to indicate priority.

A creative solution that does not compromise the easy-to-read table.

Rules are evaluated once in a user-defined order. This is a robust strategy to deal with conflicting rules and avoid circular reasoning. Rules are evaluated when a record is created, saved or displayed, depending on the rule type. The scope of a rule is an object. A rule for the object contact may only refer to fields of a contact. Some rule types allow to check the field of a related object, like the account name of a contact, expressed in a way that is familiar for programmers: contact.account.name.

Rules in Salesforce should be popular. Each rule is a potential time saver. I have seen a case that automated a robust lead scoring strategy in Salesforce. The result is an automated way to separate out qualified leads without having to do manual work. Great. But some organizations may need a Salesforce consultant to update and manage the rules. That is time consuming and costly.

What could be improved from a 'business rules' perspective?

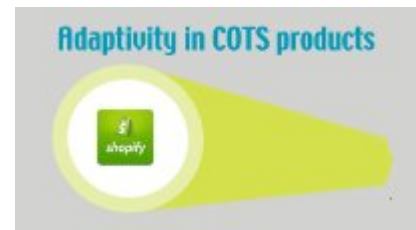
More management features

Rules in Salesforce have a name, description and may be active or inactive. The rules are listed by rule-type and object but are not a Salesforce object themselves. Why not? Why not have workflow to approve rules? Extra fields to store the source of a rule? Get an overview of all rules that are valid during summer?

Business friendly representation

The formula's are not business friendly to read while the rules are made-up by the business. Improvements should come from controlled natural languages like SBVR's RuleSpeak and decision tables as alternative representation and have translations of "LEN(Competitor__c) = 0" to it's actual meaning: *Competitor field is empty*. If the rule is expressed in a business friendly way it may be used directly as instruction when a rule is violated. This saves work and improves consistency.

Introduce a feedback loop



The sales process is sensitive for trends. A strategy that worked one day may not work the other day. An automated strategy with rules is agile by allowing exceptions. Stay up-to-date by catching the exception in the rule management process.

Next in this series is about [Rules in Shopify](#).

Let me know if you have been the one reading more than 900 words by liking this post.