



Count your rules!

The most-asked question at business rules conferences is *“How many rules do you have?”* This question is typically asked of someone who is showing the results of his business rules project. The person asking the question often wants to get some kind of indication of the complexity, size, and maturity of the project at hand and believes the answer to this question will help. However, I doubt that the answer will be of much help because both the question and the answer can be interpreted in many ways....

A related question very often asked of vendors in the business rules space is *“What is the largest set of rules that is implemented with your product?”* The answer to this question is usually a very large number ... but again, what does the answer really tell you? The person asking the question often wants to get more information on matters like performance, scalability, and maturity of the product. However, I doubt that the answer will help because the question is not a good starting point to get real answers on such matters.

Another related question comes when your product manager asks you how many rules you have gathered this week ... and *“How many rules still need to be done?”* The analyst or domain expert able to answer this question is exceptional!

The answer to these questions usually varies between 100 and 30,000 rules. What can you conclude if the answer is at one end of the spectrum, somewhere in the middle, or at the other end of the spectrum? Suppose you asked one of these questions and you get an answer and the answer is ... 2500. What does that tell you?

The purpose of this column is to analyze the question and the answer in more detail.

Let's first look at the answer. If you want to be able to make comparisons between the answers, you need to be sure everyone counted the rules in the same way. Is there a standard way to count rules? Let's look at some examples and count the number of rules in the examples.

Example 1

A client is a gold client if one of the following conditions is met:

- the client traveled more than 100,000 miles by airline
- the client purchased goods for more than 10,000 dollars
- the client has been a gold client for the past two years and traveled 50,000 miles by airline

A client is a silver client if all of the following conditions are met:

- the client traveled less than 100,000 and more than 50,000 miles
- the client purchased goods for less than 10,000 dollars
- the client has not been a gold client for the past two years

How many rules do you count? 2 rules or 4 rules? From a business perspective this is often counted as two rules. From a technical perspective, it depends. There are tools (like Haley) that will only allow you to define this logic using 4 separate rules because disjunctions in rules are not allowed. There are also tools (ILOG, Blaze) that allow you to create one rule for this logic using the ELSE construct. What is the true answer?

Example 2

Now look at the following table and picture a manager who tells you they have 10,000 rules:

Year	Minimum number of miles	Level
1980	3,000	bronze
1980	5,000	silver
1980	6,000	gold
1980	9,000	platinum
...
2011	10,000	junior
2011	20,000	bronze
2011	40,000	silver
2011	70,000	gold
2011	100,000	platinum

A well-accepted view in the business rules community is that the table is a decision table and a decision table consists of rules — in fact, it represents a collection of rules. Now, suppose they have this loyalty program for 35 years, each year adding new entries and sometimes changing the level names and defining a transition between levels. This would result in $35 * 6 = 270$ rules. Likely, the manager used the same calculation method for similar kinds of logic, resulting in his belief that he has 10,000 rules and a maintenance issue.

What do you think?

- A. Whoa, that is complex to maintain!
- B. We really need a rule engine to execute this logic....
- C. These are some records in a database.

My thought is described by C.

Conclusion:

There is no standard or accepted method for counting rules.

Now, let me revisit the questions with you.

You wanted to understand the complexity and size of the domain? The first example has only 4 rules and the second example counts 270 rules. However, the first example is more complex to maintain because there is more terminology involved: we have 4 rules but 8 different concepts (client, silver client, gold client, miles, airline, years, dollars, goods, ... and you can easily think of more). In the second example, we have 270 rules but only 3 concepts.

Conclusion:

The number of concepts in a domain gives a better measure of complexity and size than the number of rules.

You wanted to understand the performance and scalability of a rule engine? This really depends on how the rule engine processes these rules and not on the number of rules. An interesting question to ask about Example 1 is whether the second business rule is processed when the first business rule is evaluated and concludes that the customer is a gold customer. In the second example, a good question is: Which rows are processed when it is known that a client has traveled more than 20,000 miles? Listen carefully to the vendor and try to understand how you need to represent your logic in their product to get optimal performance for your situation. Nothing is ever easy....

Conclusion:

The performance and scalability of a rule engine depends on the algorithm the engine uses to process your rules: represent your rules in such a way that the engine can do its job in an optimal way.

To the project manager and business consultant I would say: “Don’t count progress in number of rules but instead create a high-level overview of the decisions in your domain.” Ask the domain expert to indicate how ‘difficult’ a decision is and to provide that list to your project manager.

More on decisions in my next column ... as we all go with the flow.

Learned something new? Let me know by sharing this post.

This article was originally published by BRCommunity ([link](#)).