"Pay-As-You-Go Pricing" Challenge, May-2020
A solution based on a relational model of business rules
by Leonid Nisenboym

1. The Challenge Description (from Web Site)

SaaS (Software as a Service) is taking over the cloud computing market. It offers pay-as-you-go pricing models that enable customers to pay only for what they use. Here is an example of pricing rules when your monthly bill is determined by the number of units (e.g. executed transactions) you actually use by tiers:

Tiers	Units	Cost
Free Tier	The first 10,000 units are free	$0 / unit
Tier 1	Up to 20,000 units per month	$0.050 / unit
Tier 2	From 20,000 to 50,000 units per month	$0.030 / unit
Tier 3	Above 50,000 units per month	$0.010 / unit

Now assume that you are a vendor of a SaaS product that needs to report the actual use of your product to the cloud provider in accordance with the above rules. For example, AWS requires each SaaS vendor to accumulate the actual use per customer and report the charged amount to the AWS metering system every hour by tiers. So, you need to create a pricing decision service that implements these rules. Your service will be called at the beginning of each hour and will receive 3 parameters:

- Number of all used units (from the very first use until the last hour including)
- Number of units used this month (the last hour including)
- Number of units used the last hour.

Your pricing service should produce the numbers of charged units for Tier 1, Tier 2, and Tier 3 **for the last hour**. Here are examples:

All units	This month units	Last hour units	Tier 1	Tier 2	Tier 3
2,000	2,000	160	0	0	0
10,200	10,200	350	200	0	0
25,600	8,678	1,234	1,234	0	0
1,500	1,500	1,500	0	0	0
22,000	22,000	20,500	10,000	2,000	0
32,500	32,500	10,500	0	10,500	0
55,600	20,500	700	200	500	0
85,600	25,700	1,500	0	1,500	0
120,258	60,390	2,350	0	0	2,350
120,000	120,000	120,000	10,000	30,000	70,000

2. How I Addressed this Challenge?

To address this challenge a relational model of business rules and related technology was applied. This technology includes a set of business rule structures, their relational models and methods of representation and reasoning. Its distinctive feature is a relational view at business rules. I am developing this technology with the goal to create means for rapid development of agile, domain oriented, specialized decisioning solutions addressing particular business needs. This Challenge provides an opportunity to add a new business rule structure and related methods to the relational business rules technology toolset.

Below the process of developing solution to the challenge is shown including business rules design and its application for building the correspondent relational rule base, reasoning solution, development and testing results.

3. Solution Development

3.1. Business Rules Design

3.1.1. Glossary

To create a business rules glossary the source data elements should be formalized.

The master data elements are denoted as follows: FreeUnits (maximum quantity of free units), Tier1_Max (Tier 1 threshold), and Tier2_Max (Tier 2 threshold).

The customer records data elements are denoted as follows: Arg1 (all units), Arg2 (this month units), Arg3 (last hour units). To bring more clarity in understanding the decisioning process the customer data is replenished by adding additional attribute denoted as CurrMonth_Billable. This attribute defines this month quantity of billable units by excluding free units (when applicable) from this month units (Arg2). As it will be seen below it is a key attribute defining the decisioning process. After applying the proper replenishment procedure the customer data looks as follows:

RecNo	Arg1	Arg2	Arg3	CurrMonth_Billable
1	2,000	2,000	160	0
2	10,200	10,200	350	200
3	25,600	8,678	1,234	8,678
4	1,500	1,500	1,500	0
5	22,000	22,000	20,500	12,000
6	32,500	32,500	10,500	22,500
7	55,600	20,500	700	20,500
8	85,600	25,700	1,500	25,700
9	120,258	60,390	2,350	60,390
10	120,000	120,000	120,000	110,000

3.1.2. Structure and Grouping

Structurally business rules represent inferences in the form of relation containing two types of attributes: conditions (premises) and decisions (conclusions). There are five condition-related attributes. They provide answers to the following four questions: what is the status of free units' usage (denoted as Condition1), how much units to distribute (denoted as Condition2), which tier the distribution should start from (denoted as Condition 3) and which tier the distribution should end at (defined by Condition4 and Condition5 when necessary). There are three decision attributes that define the quantity of units to be assigned to each of three tiers. They denoted as Tier1_Usage_Decision, Tier2_Usage_Decision, and Tier3_Usage_Decision.

All business rules organized in four groups depending on the tier the unit distribution ends. The names and notations of the groups are as follows: zero-fee group (denoted as 0), tier 1 group (denoted as 1), tier 2 group (denoted as 2),, and tier3 group (denoted as 3), Each business rule notation consists of two parts: business rule group number separated by dot with the sequence number of the rule in the group, for example, 0.1, 1.1, 1.2, 2.1, etc.

3.1.3. How decisions are made?

The smallest quantity between CurrMonth_Billable and Arg3 defines the distribution quantity. For example, if Arg3 smaller than CurrMonth_Billable it means the latter includes units used during current month before last hour. Therefore, Arg3 represents the distributable quantity. On the other hand, if CurrMonth_Billable is smaller than Arg3 it means that the Arg3 quantity includes free units not used and therefore the quantity to distribute is represented by CurrMonth_Billable quantity.

The variance between CurrMonth_Billable and Arg3 defines the tier where distribution should be started while the value of CurrMonth_Billable defines the tier where the distribution should end.

Below is an example of a customer record which includes units' distribution decisions as well as the name of the business rule applied and the corresponding business rule:

Customer record

RecNo	Arg1	Arg2	Arg3	CurrMonth_Billable	Tier1_Usage	Tier2_Usage	Tier3_Usage	RuleName
7	55,600	20,500	700	20,500	200	500	0	Rule 2.1

Business rule

Rule Name	Condition1	Condition2	Condition3	Condition4	Condition5	Tier1_Usage_Decision	Tier2_Usage_Decision	Tier3_Usage_Decision
Rule 2.1	(Arg1-Arg2) >= FreeUnits	Arg3 <= CurrMonth_Billable	(CurrMonth_Billable-Arg3) <=Tier1_Max	CurrMonth_Billable > Tier1_Max	CurrMonth_Billable <= Tier2_Max	Tier1_Max - (CurrMonth_Billable - Arg3)	CurrMonth_Billable-Tier1_Max	0

3.2. Relational Rule Base

Below is a list of business rules the relational rule base contains.

Rule Name	Condition1	Condition2	Condition3	Condition4	Condition5	Tier1_Usage_ Decision	Tier2_Usage_ Decision	Tier3_Usage_Decision
Rule 0.1	Arg1 < FreeUnits					0	0	0
Rule 1.1	(Arg1-Arg2) >= FreeUnits	Arg3 <= CurrMonth_ Billable		CurrMonth_Billable <=Tier1_Max		Arg3	0	0
Rule 1.2	(Arg1-Arg2) < FreeUnits	CurrMonth_ Billable < Arg3		CurrMonth_Billable <= (Tier1_Max-FreeUnits)		CurrMonth_Billable	0	0
Rule 1.3	(Arg1-Arg2) < FreeUnits	Arg3 <= CurrMonth_ Billable		CurrMonth_Billable <= (Tier1_Max - FreeUnits)		Arg3	0	0
Rule 2.1	(Arg1-Arg2) >= FreeUnits	Arg3 <= CurrMonth_ Billable	(CurrMonth_Billable-Arg3) <=Tier1_Max	CurrMonth_Billable >Tier1_Max	CurrMonth_Billable <= Tier2_Max	Tier1_Max - (CurrMonth_Billable - Arg3)	CurrMonth_Billable-Tier1_Max	0
Rule 2.2		Arg3 <= CurrMonth_ Billable	(CurrMonth_Billable-Arg3) > Tier1_Max	(CurrMonth_Billable-Arg3) <= Tier2_Max		0	Arg3	0
Rule 2.3	(Arg1-Arg2) < FreeUnits	Arg3 <= CurrMonth_ Billable	(CurrMonth_Billable-Arg3) > (Tier1_Max-FreeUnits)	CurrMonth_Billable<=(Tier2_Max-FreeUnits)		0	Arg3	0
Rule 2.4	(Arg1-Arg2) < FreeUnits	CurrMonth_ Billable < Arg3	(Arg3-CurrMonth_Billable) <=(Tier1_Max-FreeUnits)	CurrMonth_Billable <= (Tier2_Max - FreeUnits)		Tier1_Max-FreeUnits	CurrMonth_Billable - Tier1_Max + FreeUnits	0
Rule 2.5	(Arg1-Arg2) < FreeUnits	Arg3 <= CurrMonth_ Billable	(CurrMonth_Billable - Arg3) < (Tier1_Max-FreeUnits)	CurrMonth_Billable <= (Tier2_Max-FreeUnits)		Tier1_Max-FreeUnits	Arg3-Tier1_Max+FreeUnits	0
Rule 3.1			(CurrMonth_Billable-Arg3) > Tier2_Max			0	0	Arg3
Rule 3.2	(Arg1-Arg2) < FreeUnits	CurrMonth_ Billable <= Arg3	(Arg3 - CurrMonth_Billable) <= (Tier1_Max - FreeUnits)	CurrMonth_Billable > Tier2_Max		Tier1_Max-FreeUnits	Tier2_Max - Tier1_Max	CurrMonth_Billable + FreeUnits-Tier2_Max
Rule 3.3	(Arg1-Arg2) >= FreeUnits	Arg3 <= CurrMonth_ Billable	(CurrMonth_Billable - Arg3) >= Tier1_Max	CurrMonth_Billable > Tier2_Max		0	Tier2_Max-Tier1_Max	Arg3 - (Tier2_Max-Tier1_Max)
Rule 3.4	(Arg1-Arg2) < FreeUnits	Arg3 <= CurrMonth_ Billable	(CurrMonth_Billable-Arg3) <= (Tier1_Max-FreeUnits)	CurrMonth_Billable > (Tier2_Max--FreeUnits)		Tier1_Max-FreeUnits	Tier2_Max - Tier1_Max	Arg3-Tier2_Max + FreeUnits
Rule 3.5	(Arg1-Arg2) >=FreeUnits	Arg3 <= CurrMonth_ Billable	(CurrMonth_Billable-Arg3) <= Tier1_Max	CurrMonth_Billable > Tier2_Max		Tier1_Max-(CurrMonth_Billable-Arg3)	Tier2_Max - Tier1_Max	Arg3 - (Tier1_Max-(CurrMonth_Billable-Arg3))-(Tier2_Max - Tier1_Max)

3.3. Reasoning Engine

The reasoning engine scrolls through customer records and looks in the rule base for a match between customer record and business rule conditions. After the match has been found the corresponding tier decision formulas are applied.

3.4. Development

To implement the solution an application was developed on the basis of MS Access.

There are two components in the application that are accessible through the front end. They are: master data and rule base. Free units quantity and tier limits can be changed and effect decisioning results without making any changes on the programming level. It is also true for the rule base. Relational approach to its design makes it well structured, visible, easy to edit and update.

3.5. Testing

Below is a report showing results of testing. Each customer record includes the source data as well as unit distribution decisions and explanation - the name of business rule applied.

Rec No	Arg1	Arg2	Arg3	CurrMonth_ Billable	Tier1_Usage	Tier2_Usage	Tier3_Usage	Rule Applied
1	2,000	2,000	160	0	0	0	0	Rule 0.1
2	10,200	10,200	350	200	200	0	0	Rule 1.2
3	25,600	8,678	1,234	8,678	1,234	0	0	Rule 1.1
4	1,500	1,500	1,500	0	0	0	0	Rule 0.1
5	22,000	22,000	20,500	12,000	10,000	2,000	0	Rule 2.4
6	32,500	32,500	10,500	22,500	0	10,500	0	Rule 2.3
7	55,600	20,500	700	20,500	200	500	0	Rule 2.1
8	85,600	25,700	1,500	25,700	0	1,500	0	Rule 2.2
9	120,258	60,390	2,350	60,390	0	0	2,350	Rule 3.1
10	120,000	120,000	120,000	110,000	10,000	30,000	70,000	Rule 3.2