TEST DATA SOLUTIONS FOR ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING APPLICATIONS

Artificial Intelligence and Machine Learning are two of the hottest buzzwords in the field of Information Technology. Artificial Intelligence is the broader term and is defined by Investopedia this way:

Artificial intelligence is a term for simulated intelligence in machines. These machines are programmed to "think" like a human and mimic the way a person acts. The ideal characteristic of artificial intelligence is its ability to rationalize and take actions that have the best chance of achieving a specific goal, although the term can be applied to any machine that exhibits traits associated with a human mind, such as learning and solving problems.

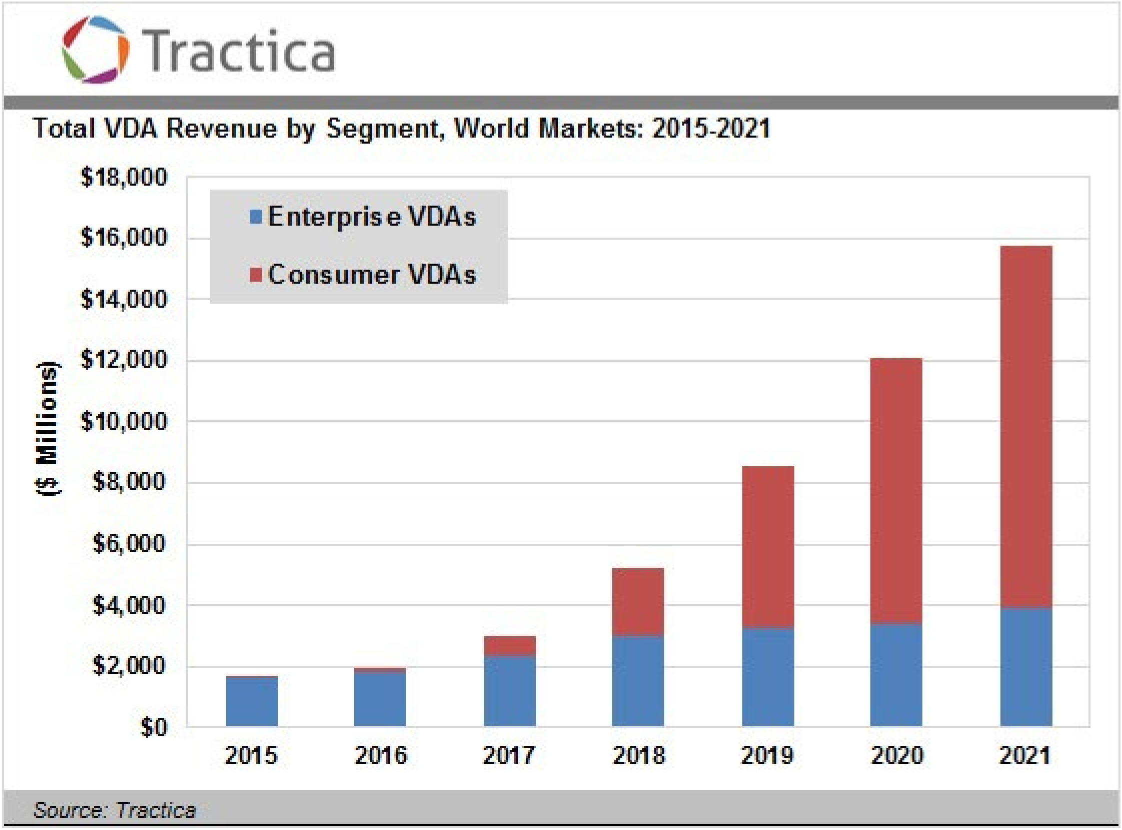
Artificial Intelligence (AI) is behind the growing popularity of the *Virtual Digital Assistant* (VDA) as popularized by *Google Home, Siri, Cortana* and *Alexa* and used by consumers to answer questions and automate everyday tasks. Business are increasingly using VDAs for sales, marketing and customer service applications as well.

For example, Bank of America's *Erica* serviced 3.5 million users and 11 million transactions within three months of its launch, helping banking customers to check their account balances, monitor transactions, access account numbers and check credit scores. Allstate's VDA, referred to as *Abie*, helps agents to quote business insurance products quickly and accurately, while reducing call center traffic and providing instant answers to policy questions during the quotation process.

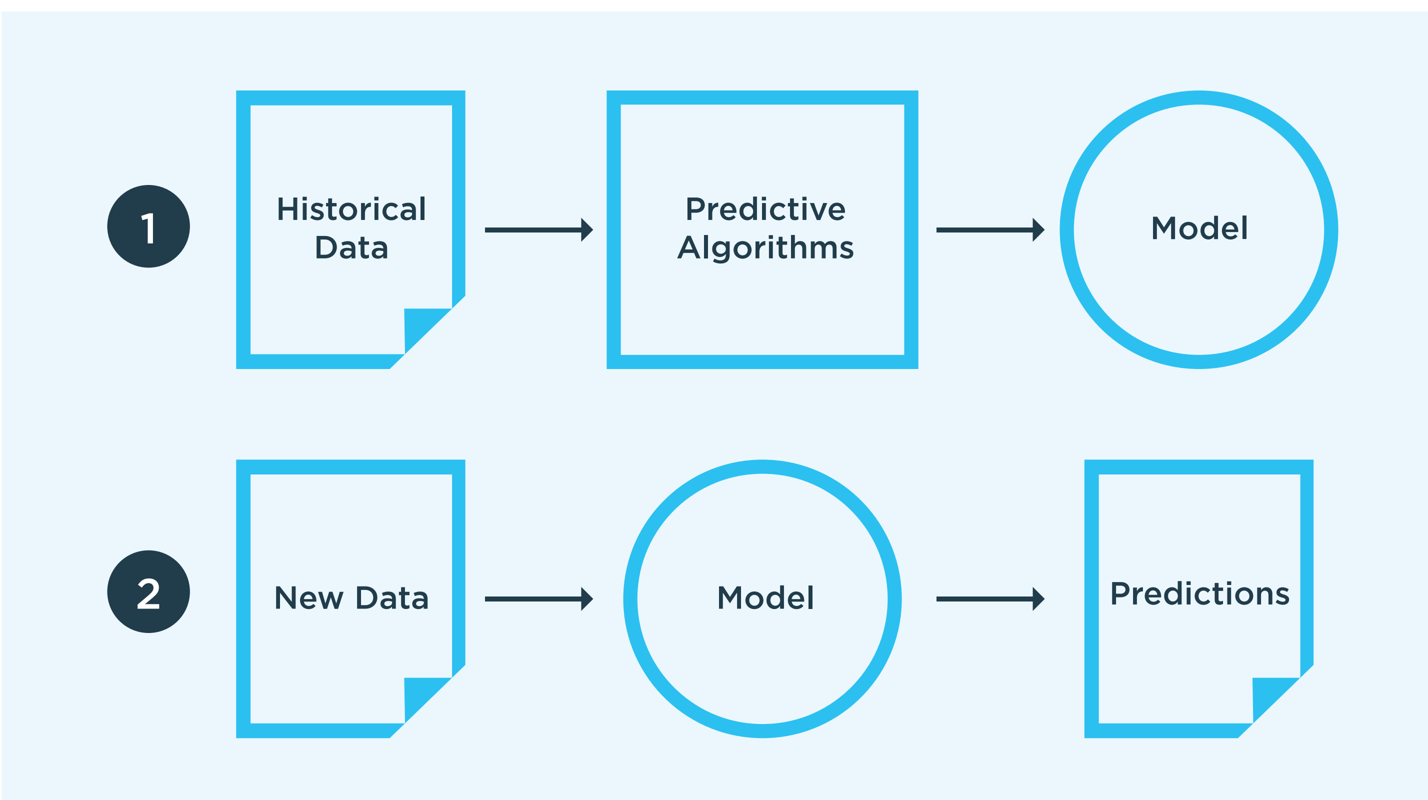
According to market research firm Tractica, active consumer VDA users will grow from 390 million in 2015 to 1.8 billion worldwide by the end of 2021.

During the same period, active enterprise VDA users will rise from 155 million in 2015 to 843 million by 2021.

The market intelligence firm forecasts total VDA revenue will grow from $1.6 billion in 2015 to $15.8 billion in 2021.



Machine Learning (ML) is a subset of artificial intelligence and is the enabling technology behind the rapidly growing field of *predictive analytics.* Machine learning uses sophisticated algorithms that allow computers to recognize patterns from current and historical data, learn from those patterns and then make predictions about future outcomes.



For example, those outcomes might be behaviors a customer is likely to exhibit during a shopping experience or anticipate possible changes in the stock market.

Predictive analytics help business leaders to understand and predict possible future occurrences by analyzing the past.

Internet-based applications of machine learning are becoming commonplace –events that appear in your Facebook feed, product recommendations made by Amazon, and movie suggestions presented in Netflix – they all make predictions based on data patterns analyzed by machine learning algorithms.

Predictive analytics has become a high-growth market opportunity. According to Stratistics MRC, the Global Predictive Analytics market is expected to grow from $3.89 billion in 2016 to reach $14.95 billion by 2023 with a CAGR of 21.2%. Fueling this rapid growth of machine learning and predictive analytics are a multitude of business applications that cross virtually every industry sector.

Here are a few examples of machine learning in action:

Recommendations Engine: Suggestions to buyers for related products during the shopping and purchasing process to cross-sell and upsell the value of transactions.

Fraud Detection: Looking for patterns and behaviors that serve as markers for criminal or fraudulent behavior during the operation of online transactions.

Personalized Marketing: Segmenting and targeting marketing campaigns to match high-value buyers with high-probability purchases based on historical and demographic data analysis.

Operational Efficiency: The use of predictive models to forecast inventory levels and manage enterprise resources based on historical operating data.

Dynamic Pricing: Setting the optimum price levels for products and services by analyzing changing market conditions and consumer demand (e.g., airline ticket and hotel occupancy).

Risk Reduction: Using credit scores to assess the likelihood of defaulting on a purchase, evaluating the risk of insurance claims, or predicting the outcome of a collections process.

Health Care Applications: Using machine learning to predict the likelihood that patients will develop a chronic disease or how they will respond to a potential treatment plan.

Insurance Applications: Improve the process of underwriting, pricing, preventing fraudulent claims and optimizing marketing programs targeting business or consumer segments.

Predictive Maintenance: The use of machine learning and vehicle data to help predict which components might fail and recommend preventative actions.

MEETING THE TEST DATA CHALLENGE FOR AI AND ML

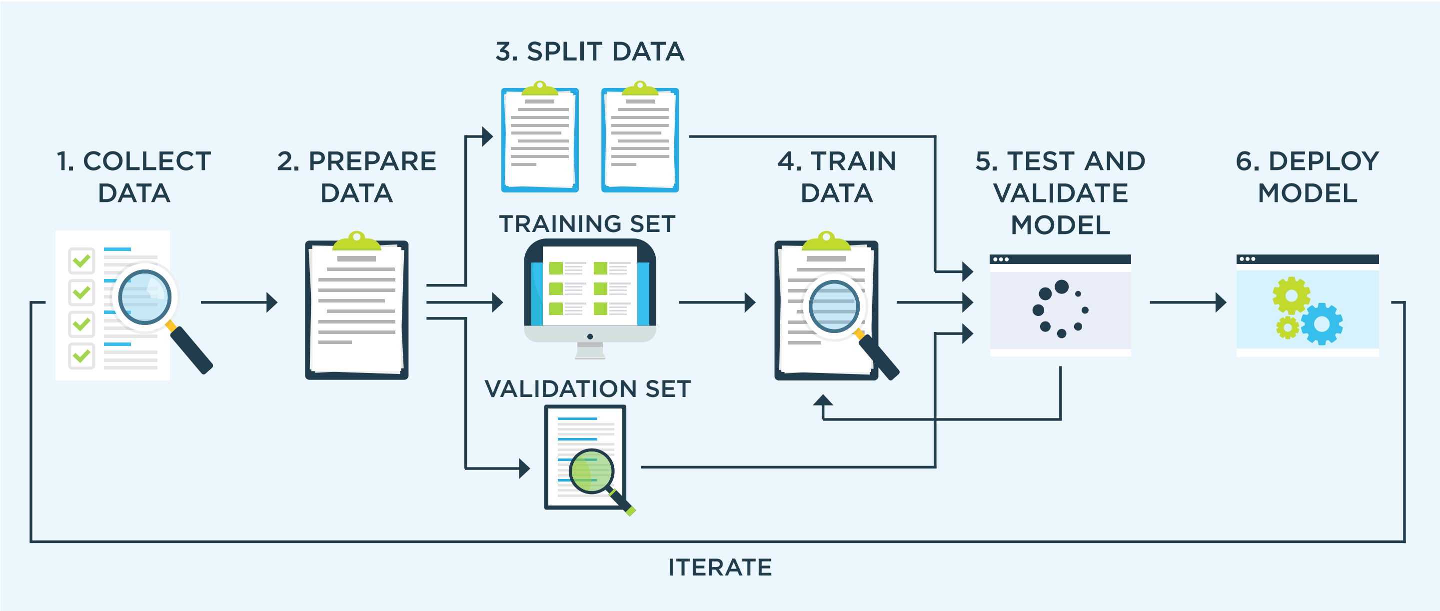
When developers and data science practitioners think about new applications for AI, ML and predictive analytics, they often think the bulk of the work will be in the development of the algorithms and how to code them. However, the biggest challenge is often on provisioning the data used to train, validate and test the model for accuracy and robustness. When perfecting a new algorithm for AI and ML applications, it helps to remember this simple rule of thumb:

The Accuracy of Algorithms used for AI and ML = High Quality Training and Test Data at Scale

How much training data is enough? To answer that question, simply ask yourself how accurate the results must be. The more data used to train and test the model, the better the learning process and the higher the accuracy of the results.

The greater the volume and variety of training data used, the more accurate and robust the model for predicting future outcomes will be. The challenge is this: How to provision a high volume of high-quality training data without spending an enormous amount of time collecting, labeling, classifying, cleaning, pruning, normalizing, and formatting the data with the help of domain experts who understand the data requirements.

Also, important to remember is the need for 3 different kinds of data during the development process: One dataset is needed to train the model, one dataset to validate the model and one dataset to test the model.



DATA REQUIREMENTS FOR ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Training Dataset: The sample of data used to train and evolve the accuracy of the model.

Validation Dataset: The sample of data used to provide an unbiased evaluation of the model.

Test Dataset: The sample of data used to provide a final test of the model prior to release.

These 3 datasets must be different to ensure the integrity of the model and how it will perform in real-world applications scenarios. That's where GenRocket's ability to generate high-volumes of data based on a predefined data model, data attributes and patterns of data variation is a perfect match for AI and ML application development. Once the domain expert specifies the data requirements, GenRocket's real-time synthetic test data engine generates controlled and conditioned data at the rate of 10,000 rows per second. This allows developers and testers to create very large datasets on-demand for the separate purposes of training, validating and testing a machine learning application.

THE TECHNICAL CHALLENGE

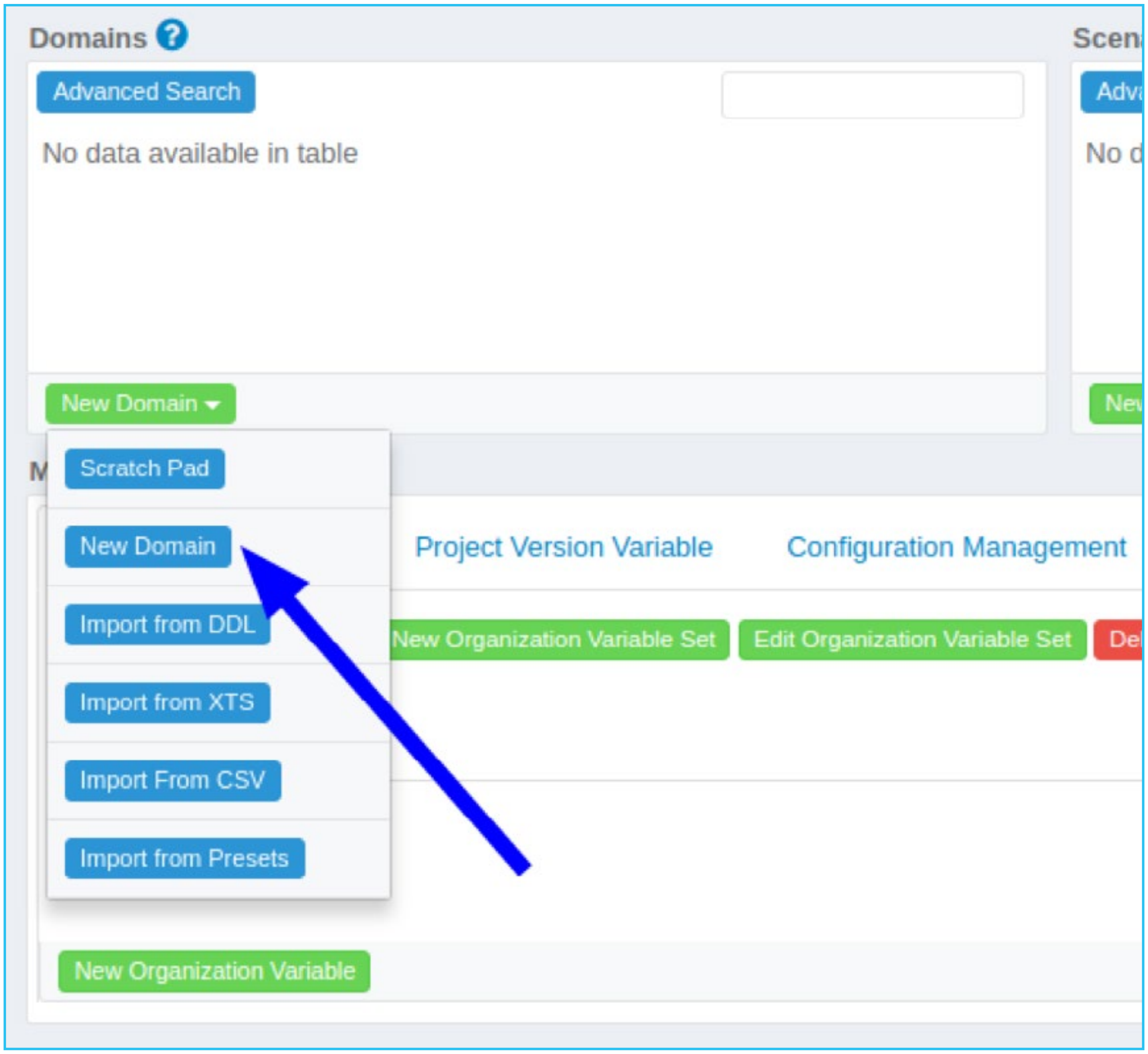
- A global services organization developing a new Artificial Intelligence application needed a large dataset of customer data with given pattern of data to train the system.
- While there were many, many use cases, one use case example was where GenRocket generated 10 million rows of customer data, with a given pattern of:
    - 20% of the Customer Names should have either Jr / Sr. (60%) or Mr / Mrs (40%) with same customer details like address, but different Phone number and Unique Identification Number.
    - The unique identification number for each customer should have a unique alphanumeric character consisting of: <2 letters> <2 digits> <2 digits> <2 digits> <1 letter> Eg. AB 12 34 56 C

THE GENROCKET SOLUTION

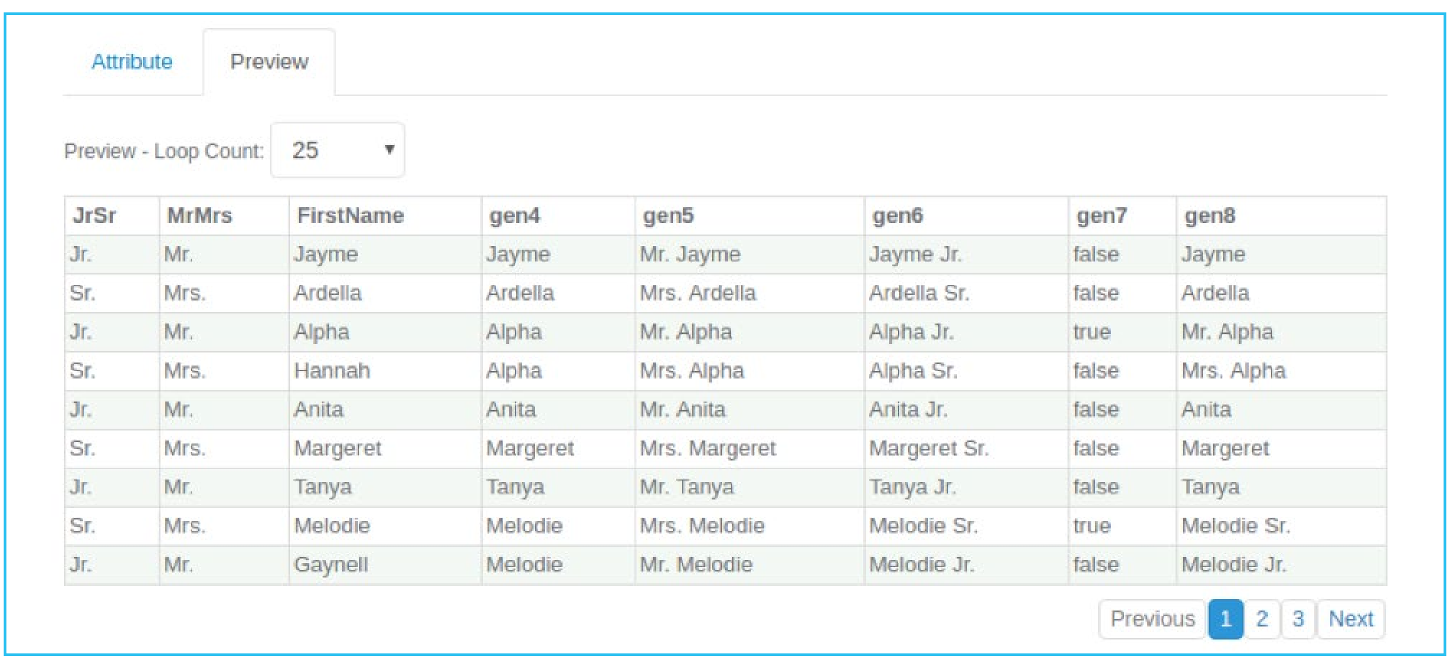
- Generating data with complex business logic can be done easily with GenRocket by breaking it down into simple concerns.
- Linked Generators were used to design complex patterns of data where the percentage of data generated was controlled from multiple lists by using Generators called the WaitAmountGen and WaitAmountReferenceGen; these Generators referenced another Domain Attribute's generated value.
- Linked Generators is a powerful GenRocket feature because linked Generators are able to directly reference each other within an Attribute as opposed to indirectly accessing another Domain's Attribute to get its Generated value. Thus, linking Generators to each other within an Attribute provides the ability to generate complex conditioned data without the necessity of having to access another Attribute's generated value. This also means that Attributes may reference other Attributes that generate complex data via linked Generators, thus yielding even more complex, conditioned data.

SETUP

- A New Domain was created using the New Domain option in the GenRocket web app and the appropriate Generators were added to each Attribute.

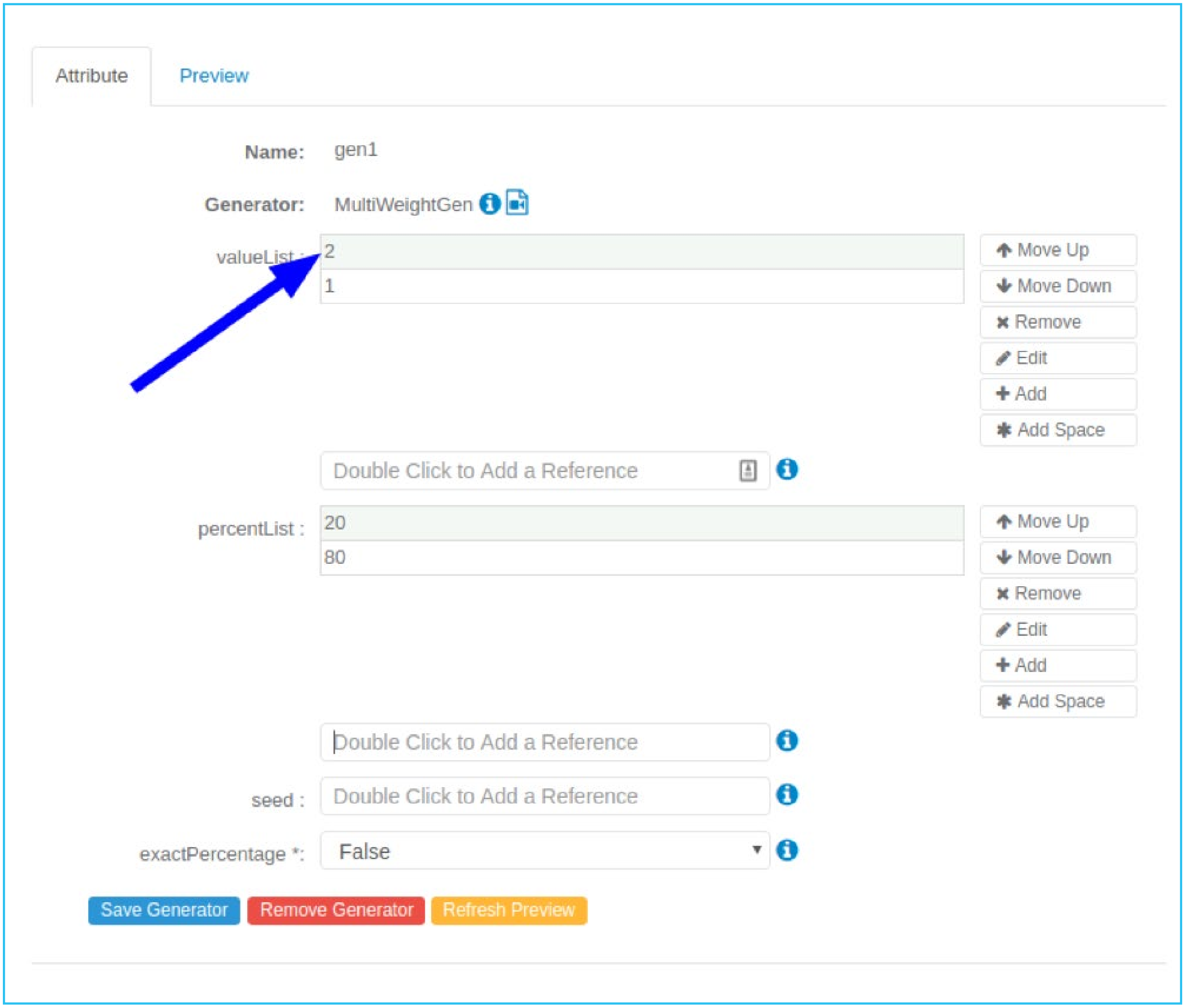


- The following image shows how the Customer name was Generated using Linked Generators.

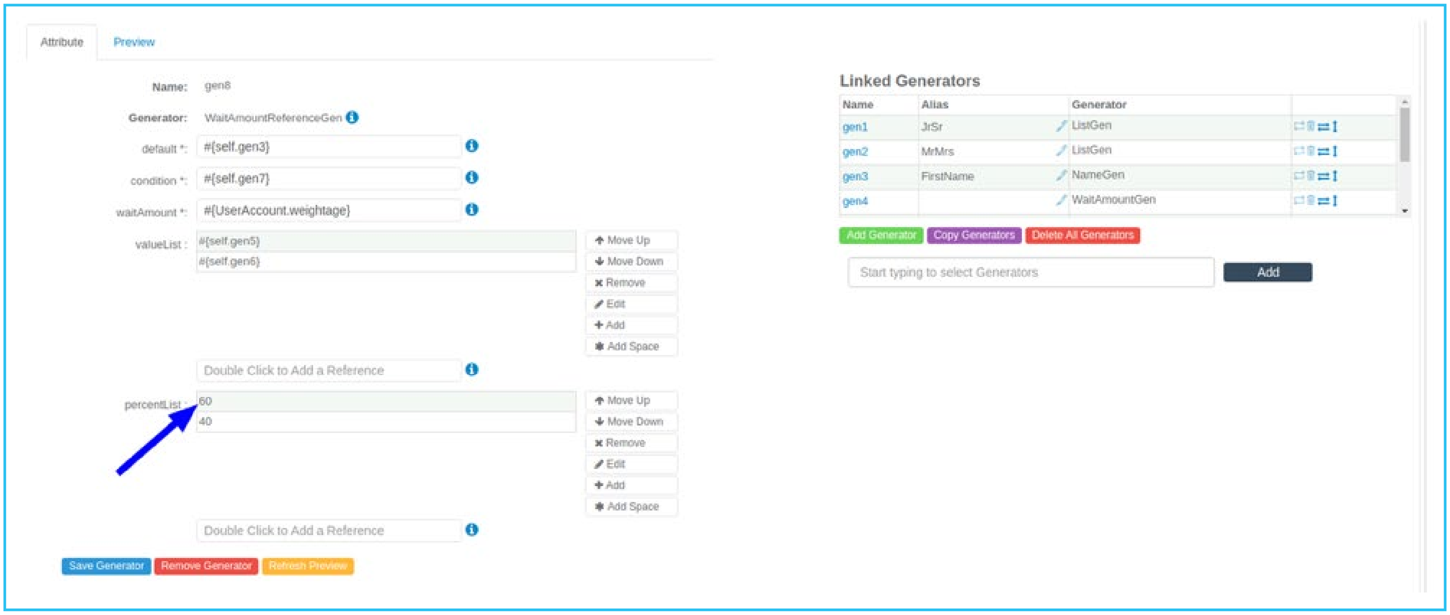


JrSr	MrMrs	FirstName	gen4	gen5	gen6	gen7	gen8
Jr.	Mr.	Jayme	Jayme	Mr. Jayme	Jayme Jr.	false	Jayme
Sr.	Mrs.	Ardella	Ardella	Mrs. Ardella	Ardella Sr.	false	Ardella
Jr.	Mr.	Alpha	Alpha	Mr. Alpha	Alpha Jr.	true	Mr. Alpha
Sr.	Mrs.	Hannah	Alpha	Mrs. Alpha	Alpha Sr.	false	Mrs. Alpha
Jr.	Mr.	Anita	Anita	Mr. Anita	Anita Jr.	false	Anita
Sr.	Mrs.	Margeret	Margeret	Mrs. Margeret	Margeret Sr.	false	Margeret
Jr.	Mr.	Tanya	Tanya	Mr. Tanya	Tanya Jr.	false	Tanya
Sr.	Mrs.	Melodie	Melodie	Mrs. Melodie	Melodie Sr.	true	Melodie Sr.
Jr.	Mr.	Gaynell	Melodie	Mr. Melodie	Melodie Jr.	false	Melodie Jr.

- For Creating the Names with the given condition, an additional Attribute was created with the MultiWeightGen Generator, which assigned the value ‘1’ 80% of the time and the value ‘2’ 20% of the time. This was to ensure that 80% of customer data had the condition of the different first name, last name, address, city, zip code phone number and unique id and 20% of customer data had the condition of the same second name, address, city and zip code.

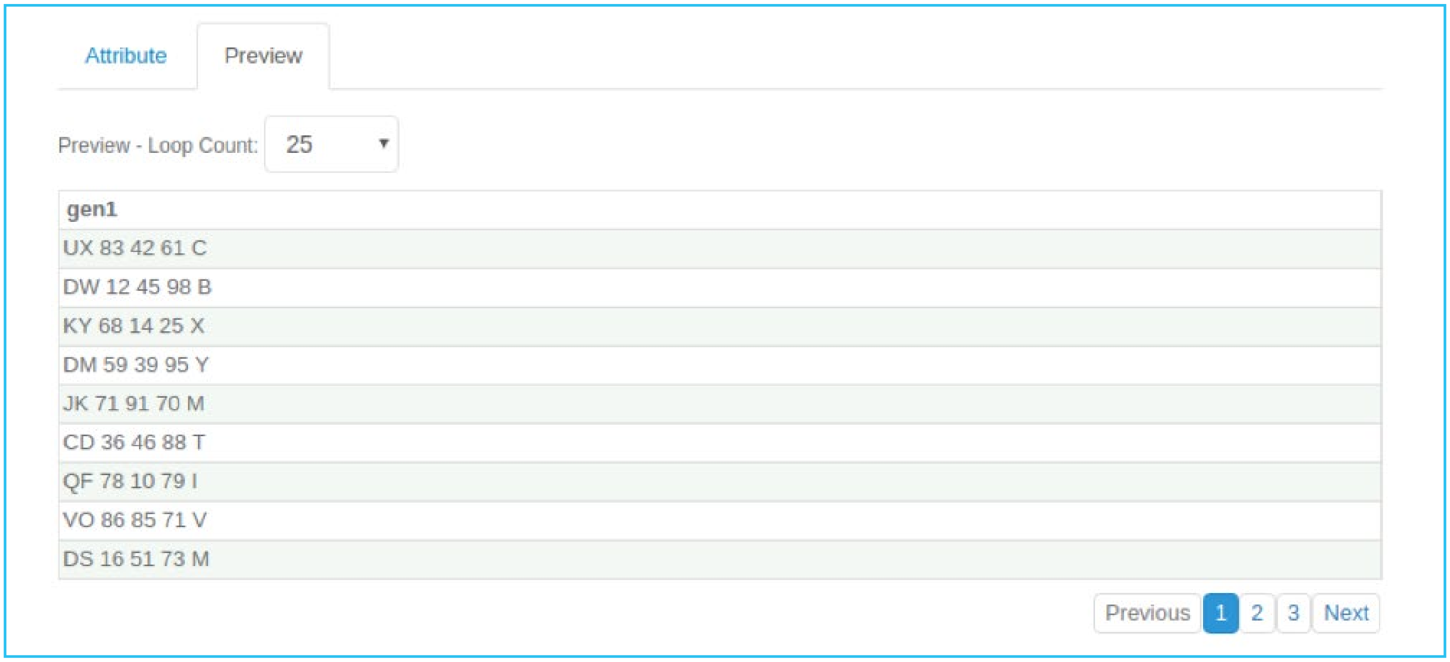


- By using Linked Generators the patterns and percentages of data generated from multiple lists were controlled and conditioned by referencing the MultiWeightGen Generator's generated value. The following image shows the parameters set in the WaitAmountReferenceGen Generator for generating First name with Jr / Sr. (60%) and Mr / Mrs (40%)

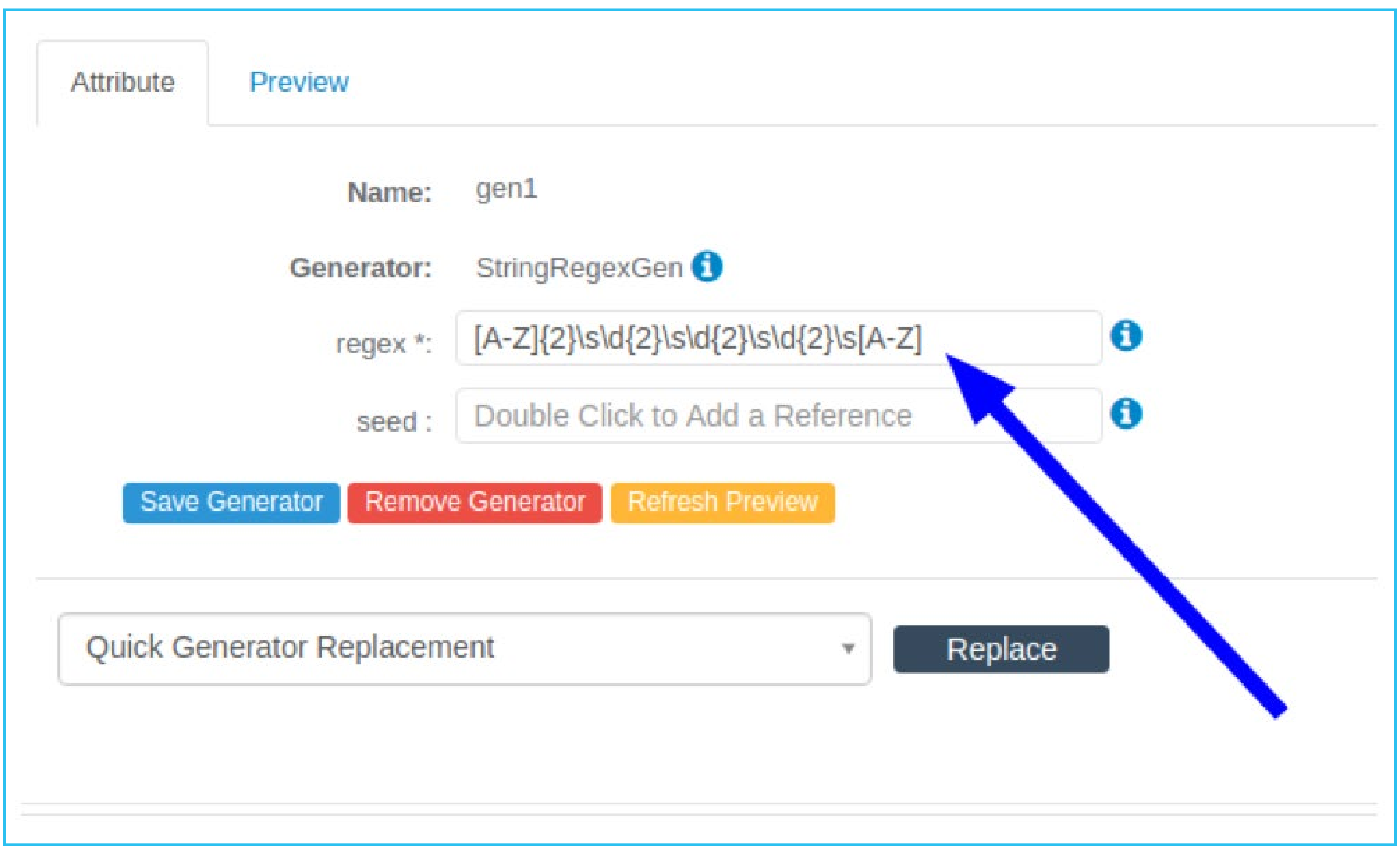


- To ensure the same value was repeated for the data being generated for all the Attributes, a linked WaitAmountGen Generator was added and referenced to the value generated by the MultiWeightGen Generator. The following image shows the AddressGen being referenced to the MultiWeightGen for the waitAmount. This process was repeated for second name, address, city and zip code.

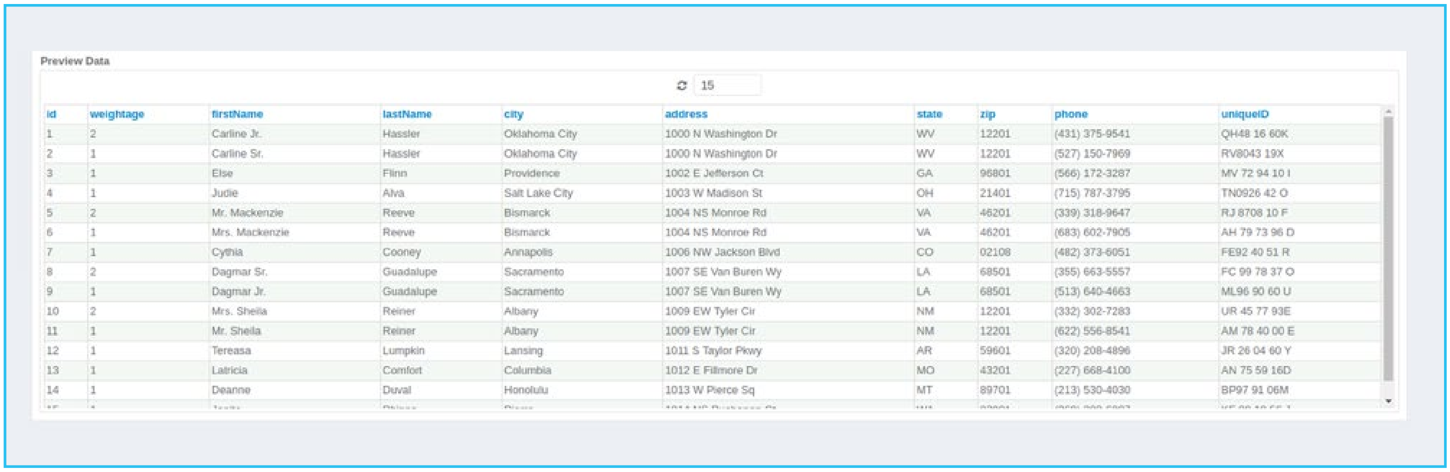
- Unique Identification Numbers like different countries' identification numbers were generated by using the StringRegexGen Generator. The following image shows a sample of data generated by using StringRegexGen by converting the given condition <2 letters> <2 digits> <2 digits> <2 digits> <1 letter> to a regex [A-Z]{2} \d{2} \d{2} \d{2} [A-Z]



- The Unique Identification Number was created by simply using StringRegexGen and adding the regex value in the regex parameter. The following image shows that the regex value was added to the regex parameter in the StringRegexGen.



- The image below shows preview data generated in the GenRocket web app matching the required data criteria.



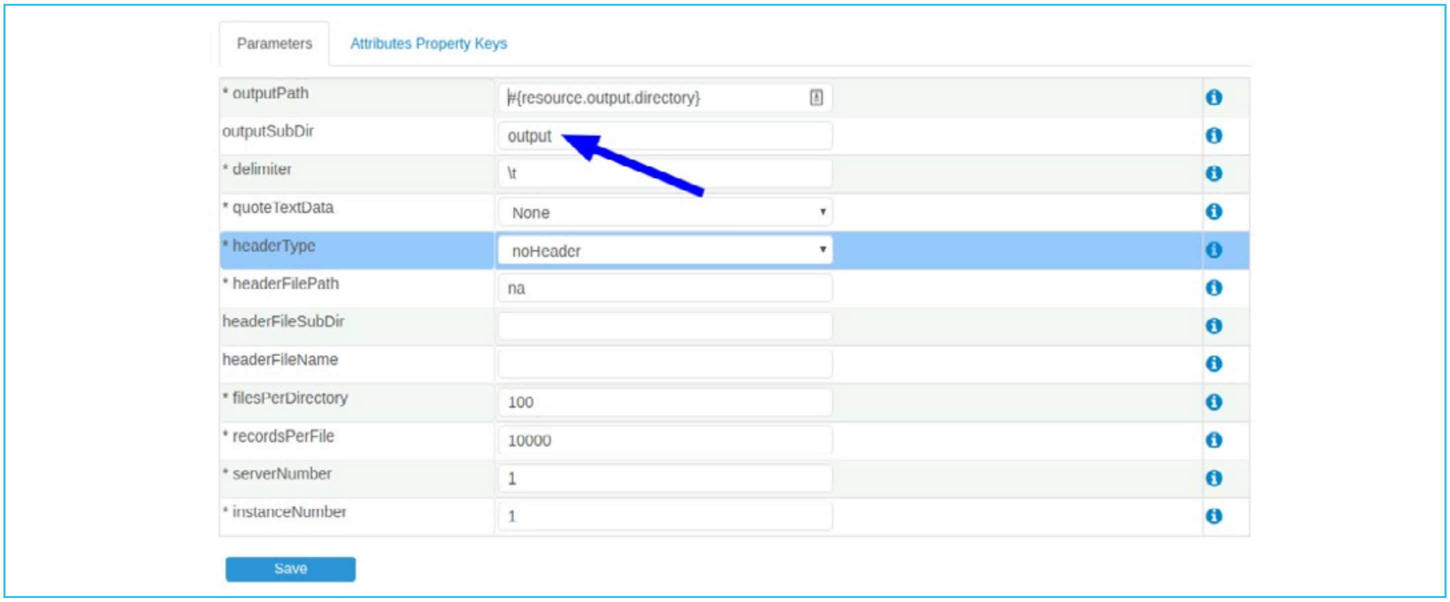
Preview Data

15

id	weightage	firstName	lastName	city	address	state	zip	phone	uniqueID
1	2	Carline Jr.	Hassler	Oklahoma City	1000 N Washington Dr	WV	12201	(431) 375-9541	QH48 16 60K
2	1	Carline Sr.	Hassler	Oklahoma City	1000 N Washington Dr	WV	12201	(527) 150-7969	RV8043 19X
3	1	Else	Flinn	Providence	1002 E Jefferson Ct	GA	96801	(566) 172-3287	MV 72 94 10 I
4	1	Judie	Alva	Salt Lake City	1003 W Madison St	OH	21401	(715) 787-3795	TN0926 42 O
5	2	Mr. Mackenzie	Reeve	Bismarck	1004 NS Monroe Rd	VA	46201	(339) 318-9647	RJ 8708 10 F
6	1	Mrs. Mackenzie	Reeve	Bismarck	1004 NS Monroe Rd	VA	46201	(683) 602-7905	AH 79 73 96 D
7	1	Cythia	Cooney	Annapolis	1006 NW Jackson Blvd	CO	02108	(482) 373-6051	FE92 40 51 R
8	2	Dagmar Sr.	Guadalupe	Sacramento	1007 SE Van Buren Wy	LA	68501	(355) 663-5557	FC 99 78 37 O
9	1	Dagmar Jr.	Guadalupe	Sacramento	1007 SE Van Buren Wy	LA	68501	(513) 640-4663	ML96 90 60 U
10	2	Mrs. Sheila	Reiner	Albany	1009 EW Tyler Cir	NM	12201	(332) 302-7283	UR 45 77 93E
11	1	Mr. Sheila	Reiner	Albany	1009 EW Tyler Cir	NM	12201	(622) 556-8541	AM 78 40 00 E
12	1	Tereasa	Lumpkin	Lansing	1011 S Taylor Pkwy	AR	59601	(320) 208-4896	JR 26 04 60 Y
13	1	Latricia	Comfort	Columbia	1012 E Fillmore Dr	MO	43201	(227) 668-4100	AN 75 59 16D
14	1	Deanne	Duval	Honolulu	1013 W Pierce Sq	MT	89701	(213) 530-4030	BP97 91 06M

LARGE VOLUME DATA GENERATION

- Standard GenRocket data generation is between 10,000 to 15,000 rows of test data per second.
- For the use case 10 millions rows of data was taking over 16 minutes so the decision was made to use the GenRocket Partition Engine to speed up data generation.
- To generate the large volumes of data in delimeted file format using the Partition Engine, the DelimitedPartitionReceiver was used. This Receiver outputs data in a delimited file format to one or more files parsed over multiple instances via the GenRocket Partition Engine. This allowed for huge amounts of data to be generated, in parallel, quickly. The generated files were then merged together into a single file using the PartitionFileMergeReceiver.
- In the DelimitedPartitionReceiver Parameters tab, the client defined the output directory in which the generated data was to be stored, the number of records generated per file and the number of files in each directory. The following image shows how the DelimitedPartitionReceiver parameters was set up for this use case.



- After creating a new Scenario and downloading the scenario to a local machine, the GenRocket Advanced REST Client was run to pass the parameters to the Partition Engine and 10 GenRocket instances were launched.
- The GenRocket Partition Engine, working with DelimitedPartitionReceiver automatically created a directory structure to store the generated data and each of the 10 partitions stored 1 million rows of data.
- Below you can see a sample of the data generated that met the use case specifications. And 16 minutes of data generation were reduced to 3 minutes and 13 seconds for 10 million rows of data.
- Note: The Partition Engine can be used to generate hundreds of millions to billions of rows of data in minutes.

300001	1	Leatrice	Mccants	Lincoln	301001 N Washington Dr	NC	06101	(680) 714-2089 QX 29 79 dd O
300002	1	Veta	Yung	Jefferson City	301002 S Adams Sq	RI	06101	(580) 511-3609 IH 79 87 dd G
300003	1	Noemi	Franke	Pierre	301003 E Jefferson Ct	DC	82001	(353) 388-6015 HT 00 29 dd F
300004	1	Stevie	Padgett	Providence	301004 W Madison St	SD	96801	(796) 728-4959 DN 51 72 dd L
300005	2	Barbra Jr.	Ries	Columbia	301005 NS Monroe Rd	RI	36101	(616) 790-7123 KP 49 37 dd H
300006	1	Barbra Sr.	Ries	Columbia	301005 NS Monroe Rd	RI	36101	(693) 240-9189 VT 74 46 dd Z
300007	1	Myrtis	Shih	Providence	301007 NW Jackson Blvd	NV	83701	(482) 559-5383 MA 30 98 dd F
300008	1	Carolyne	Coco	Topeka	301008 SE Van Buren Wy	OK	05601	(605) 748-1958 YS 36 07 dd H
300009	2	Anthony Jr.	Sayers	Sacramento	301009 SW Harrison Ln	IA	59601	(716) 426-1327 RX 61 81 dd J
300010	1	Anthony Sr.	Sayers	Sacramento	301009 SW Harrison Ln	IA	59601	(331) 791-1806 ZQ 13 71 dd V
300011	1	Mable	Murry	Sacramento	301011 N Polk Pk	LA	48901	(474) 664-5312 UN 64 40 dd 7
300012	1	Rosaline	Voyles	Topeka	301012 S Taylor Pkwy	AK	02108	(401) 543-9387 FY 28 08 dd S
300013	1	Trena	Doll	Raleigh	301013 E Fillmore Dr	AK	62701	(225) 725-4079 LB 23 98 dd M
300014	1	Audrea	Moll	Nashville	301014 W Pierce Sq	TX	20001	(392) 331-7190 UD 06 24 dd W
300015	1	Jayne	Weathersby	Nashville	301015 NS Buchanan Ct	WY	89701	(232) 601-3418 LA 67 58 dd L
300016	1	Marleen	Flaherty	Topeka	301016 NE Lincoln St	TX	70801	(277) 277-7265 TM 72 54 dd D
300017	1	Rikki	Peet	Concord	301017 NW Johnson Rd	MS	48901	(492) 521-3497 GQ 16 63 dd P
300018	1	Kristy	Tompkins	Charleston	301018 SE Grant Ave	NY	40601	(497) 476-0360 ZD 04 15 dd Y
300019	2	Mr. Leticia	Lamb	Jackson	301019 SW Hayes Blvd	NJ	72201	(782) 396-6639 PI 20 31 dd Y
300020	1	Mrs. Leticia	Lamb	Jackson	301019 SW Hayes Blvd	NJ	72201	(208) 507-7459 FK 32 90 dd Z
300021	1	Mathilda	Cadet	Boston	301021 N Arthur Ln	MT	17101	(536) 714-2498 FW 25 80 dd F
300022	1	Erma	Wesolowski	Atlanta	301022 S Cleveland Cir	MN	19901	(543) 323 1229 JJ 86 30 dd V
300023	1	Yvette	Camara	Salt Lake City	301023 E Harrison Pk	WA	53562	(682) 571-5843 QK 14 49 dd O
300024	1	Henry	Rinker	Sacramento	301024 W Cleveland Pkwy	MA	97301	(448) 571-4548 RJ 95 03 dd B
300025	2	Mr. Adela	Wacker	Santa Fe	301025 NS McKinley Dr	NE	73301	(787) 331-7032 QY 17 61 dd P
300026	1	Mrs. Adela	Wacker	Santa Fe	301025 NS McKinley Dr	NE	73301	(228) 344-5483 HN 38 47 dd B
300027	1	Kym	Mcmullen	Tallahassee	301027 NW Taft Ct	DC	20001	(439) 200-7883 KE 57 41 dd L
300028	1	Carol	Hine	Baton Rouge	301028 SE Wilson St	SD	48901	(731) 287-5557 ND 66 71 dd K
300029	1	Kim	Jeske	Boston	301029 SW Harding Rd	DE	80012	(451) 425-4902 IQ 22 60 dd K
300030	1	Carmel	Abbey	Des Moines	301030 EW Coolidge Ave	MI	87501	(567) 676-4099 JA 67 76 dd R
300031	1	Susanna	Randazzo	Boston	301031 N Hoover Blvd	CO	02108	(553) 319-8466 MN 27 15 dd I
300032	2	Mrs. Lizabeth	Wingard	Santa Fe	301032 S F. Roosevelt Wy	HI	55101	(310) 113-7636 OZ 59 84 dd N
300033	1	Mr. Lizabeth	Wingard	Santa Fe	301032 S F. Roosevelt Wy	HI	55101	(557) 516-3536 HM 88 04 dd I
300034	1	Ida	Bivins	Denver	301034 W Eisenhower Cir	NY	99801	(795) 773-1453 ZP 98 95 dd A
300035	1	Halina	Gulick	Phoenix	301035 NS Kennedy Pk	CA	57501	(648) 198-9085 TV 76 01 dd L
300036	1	Kristin	Depasquale	Des Moines	301036 NE Johnson Pkwy	CT	68501	(684) 293-8237 BC 56 12 dd J
300037	2	Mr. Alesia	Lavelle	Cheyenne	301037 NW Nixon Dr	AZ	25301	(334) 521-9918 PD 13 92 dd Y
300038	1	Mrs. Alesia	Lavelle	Cheyenne	301037 NW Nixon Dr	AZ	25301	(522) 428-6682 EX 76 79 dd T
300039	2	Sanora Jr.	Tanaka	Carson City	301039 SW Carter Ct	DE	32301	(572) 308-6496 WU 46 63 dd F
300040	1	Sanora Sr.	Tanaka	Carson City	301039 SW Carter Ct	DE	32301	(471) 462-6040 ZC 58 11 dd P
300041	1	Sheron	Fowler	Helena	301041 N H.W. Bush Rd	AK	40601	(522) 617-7350 WW 52 05 dd L
300042	2	Mrs. Kari	Keys	Annapolis	301042 S Clinton Ave	IL	43201	(721) 369-5926 JN 16 99 dd N
300043	1	Mr. Kari	Keys	Annapolis	301042 S Clinton Ave	IL	43201	(559) 641-8970 AW 07 48 dd Z
300044	1	Velva	Oropeza	Jackson	301044 W Obama Wy	SC	84101	(652) 711-4638 ON 17 57 dd V
300045	1	Treva	Luker	Saint Paul	301045 NS Washington Ln	OK	97301	(378) 581-4394 GI 47 02 dd W
300046	1	Myrtis	Mccool	Santa Fe	301046 NE Adams Cir	NY	17101	(673) 574-9392 AG 37 17 dd K
300047	1	Kristina	Person	Lansing	301047 NW Jefferson Pk	HI	39201	(680) 102-3740 ON 54 16 dd K
300048	1	Noma	Landis	Albany	301048 SE Madison Pkwy	OR	05601	(694) 546-7001 CE 51 56 dd R
300049	1	Alex	Foor	Juneau	301049 SW Monroe Dr	PA	30301	(659) 415-7560 BX 57 55 dd P
300050	1	Sheree	Brick	Washington	301050 EW Adams Sq	ND	55101	(594) 630-7551 OA 61 46 dd E
300051	2	Jolynn Jr.	Birch	Richmond	301051 N Jackson Ct	NM	73301	(251) 516-5343 KD 56 98 dd L
300052	1	Jolynn Sr.	Birch	Richmond	301051 N Jackson Ct	NM	73301	(426) 129-3720 UV 86 81 dd F

IMPACT

- By using various combinations of Linked Generators, the customer was able to design and model data sets for training and testing of an Artificial Intelligence / Machine Learning application.
- GenRocket generated 10 million rows of conditioned training data in a little over 3 minutes.
- This solution greatly reduced the time and effort of creating unique data sets that were useful for the customer use case at a much lower cost.