ARTICLE

Ensure Data Security

Solution: **XDM**

UBS HAINER

Data Masking with XDM

XDM's Masking Tool is designed to tackle the diverse challenges that come with masking data. Various scenarios are conceivable, and each has unique requirements. One example is masking Personally Identifiable Information (PII) to adhere to regulations such as GDPR or HIPAA, also in situations involving sensitive health data (PHI). Regardless of the scenario, maintaining data validity is crucial to enable realistic testing.

This idea applies to both, existing edge cases and data sets that have grown over time. By using real case studies derived from production data, albeit in a modified form, we can spot errors early on and still comply with data protection regulations. Tools like XDM for data masking can efficiently alter specific data, ensuring compliance without extra effort or manual adjustments.

Another crucial point in any data masking project within a company is the existing structure. The objective is to integrate seamlessly with the existing structures, which translates to the ability to execute a masking project irrespective of the database infrastructure used. A tool for data masking must be able to handle existing structures, minimizing additional manual effort.

XDM's Masking Tool seamlessly copies and modifies data across all database boundaries and storage systems. For example, it can copy and mask data from distributed database systems alongside data from legacy VSAM systems. XDM can also operate in a cloud-based environment, enabling reading from or writing to the cloud.

XDM facilitates hybrid solutions by supporting different types of database systems, allowing for consistent definitions of modification across different platforms. This eliminates the need to create separate modifications for each system type. Modifications aligned with compliance department specifications can be universally applied, reducing the need for manual effort.

Implementation: Locating pertinent data and applying masking

A data masking project often begins with identifying the data that must be masked. This involves determining which data needs to be anonymized, which can be anonymized, and which data is required for testing or development purposes.

XDM has a tool called the PII Finder, which is used to identify personally identifiable information (PII) in a table column. The tool uses pre-configured dictionaries that contain lists of sensitive information such as names, banking numbers, or street names to check if a

Ensure Data Security

Solution: **XDM**

UBS HAINER

Implementation (Continued)

column contains any such information. Technically speaking, XDM uses so-called “matchers” to define the columns and data undergoing analysis. Once the table is scanned, XDM generates a report that displays the likelihood of a specific piece of information appearing in a column. These reports provide the basis for identifying suitable masking techniques for the relevant columns.

Once the report is available, the next step in the project involves defining the data to be masked and the masking method. XDM identifies distinct components crucial for these modifications.

Modification *methods* are used to modify data. For example, a simple method may dictate that the output value is constant, regardless of the input value. Alternatively, more complex methods can be developed using programming languages like JavaScript or Groovy to adjust internal numbers according to specific requirements.

Modification *rules* define which modification method to use for which fields. This is done by assigning the previously created method to a column within a table.

These modification rules are grouped in a modification *set*. A modification set comprises at least one modification rule and is then connected to a data copy as the project progresses.

By encapsulating the modification methods, XDM users can apply them repeatedly on different objects without needing to redefine the same masking each time. Defining the masking is only required once, and it can be reused as often as needed.

XDM's capability to parameterize modification methods adds flexibility. Users can easily apply the same method across various scenarios by adjusting one or more parameters differently for each use.

In data masking, enforcement, often used alongside authorization concepts, ensures consistent masking of specific data whenever it's copied. XDM provides flexibility by allowing these enforcement rules to be defined at a higher level than individual copy objects, known as tasks within XDM.

Ensure Data Security

Solution: **XDM**

UBS HAINER

Areas of Application

XDM offers different options for copying and modifying data:

One option is to copy data in-place and modify it simultaneously. This process involves reading the data, modifying it, and then writing it back without the need for another data storage system. This is useful, when a data copy is already in place or prepared using fast cloning mechanisms and then the masking should be applied using XDM.

Alternatively, XDM can also copy the data to a separate target system. In this case, the source data is read, modified, and then copied to a different system. XDM can also automatically create missing structures in the target system.

If you want to limit the required data without copying and modifying entire tables or schemas, you can selectively copy and modify a subset of the data. XDM can use this selective copying procedure to copy individual data records for specific test cases.

In addition to the copying methods, GDPR requirements also mandate the ability to track masked data and delete it if needed. For example, a customer requests to remove data provided for testing or development purposes after its completion. To comply, it must be possible to retrieve the data for complete deletion.

Another requirement involves blacklisting certain data, which excludes it from the masking process. This ensures that the data is not included in the masked data set provided in the target system.

Defining and Customizing Modification Methods

XDM relies on modification methods to specify the type and method of data masking. By default, XDM includes a set of predefined methods that can be used right away in any masking project. These preconfigured methods are well-suited for common scenarios such as masking names, addresses, bank information, and geolocation data.

You can also create customized modification methods, which can be tailored to fit the requirements of the existing data structure. This customization can involve working with hash algorithms, integrating custom libraries (JavaScript, Groovy, or Java), or generating custom lookup tables. You can use the data that’s already in the application database to create lookup tables, which helps keep track of special cases in the original data.

Modification Methods (Continued)

By using specially calculated hash values, XDM ensures that data masking can be repeated as needed, and the values are consistently anonymized. It is also possible to generate hash values based on other fields, either in the same table or a different one. By randomizing the order of the lookup tables, it is possible to change the results of the masking procedure. So, it is possible to have deterministic masking results on a defined period, but at a given point in time, the results will be different.

XDM provides a variety of reports that summarize and document the data masking procedures used in data copies.

Masking Personal Information

Data masking involves changing existing information in a way that makes it impossible to draw meaningful conclusions about the original data. It's crucial to follow guidelines like GDPR for effective data masking. Projects that focus on data masking need a tool capable of masking existing data in accordance with these guidelines. Yet, various challenges must be addressed in this process. We'll explore these challenges using examples and showcase how XDM can effectively handle data masking.

Masking Names

One of the most common examples of data masking begins with a person's name, typically comprising a first and last name. Modifying a first name, for instance, often requires that the altered name remains identifiable as a person's name. This ensures that the data and reports generated from the modified data remain usable for technical and exploratory tests.

Within XDM, this can be accomplished using lookup tables containing a set of first and last names. A mathematical calculation is then used to determine which name replaces another. An identical name needs to be consistently changed throughout the database if it appears in multiple places. This can be achieved by using either a fixed number (e.g., a personal identifier) or the input name itself. Using the input name eliminates the need for an additional reference number, but it may result in completely different names if there are slight spelling variations in different database locations.

Masking Names (Continued)

Additionally, it's often necessary to ensure that the gender of the individual remains unchanged. For example, if the original name is "John," which is typically male, the masked name should also be male, such as "Mark". This requires information about the person's gender, which can be extracted directly from the table or determined using a lookup table. If neither is feasible, XDM can generate a random name as the masked value.

There are cases where last names contain double names or spaces or where first and last names are stored together as a single "name" in a table structure. It is crucial to accurately identify and mask these cases while following the predefined rules for modification.

Another example of special masking requirements involves birth data. In certain use cases, a person must belong to a specific age group even after the name has been masked. The following example is from the insurance industry: John, who is 18 years old, has insured his car. As a novice driver, he has been assigned to a specific tariff group for car insurance. To ensure that John remains in this rate group even after his real data is modified for testing, his date of birth cannot be arbitrarily changed but must be masked according to specific criteria defined by an XDM user. For instance, only the day and month may be masked, while the year remains unchanged.

Masking Addresses

Another example from the insurance industry involves altering policyholder addresses. When changing data, it's crucial to consider the type of insured property to ensure it remains unchanged. For instance, a single-family home is insured differently than a multi-family home. If data records of individuals who have insured a single-family home are needed for a test case, the masked data must accurately reflect this information.

This situation revolves around the specific risk of flooding (or other potential hazards such as avalanches) and the corresponding classification of buildings into different risk groups. These factors are considered when purchasing insurance and must be masked appropriately for testing scenarios. This implies that the address cannot be altered randomly. For instance, a person residing in risk area A should still be classified as living in risk area A even after the original data has been modified.

XDM offers various options to mask addresses. For example, you can specify that the masked address must be within a certain distance of the original address or in the same city or zip code area as the original address.

Masking Addresses (Continued)

All of the above examples need to be properly discussed with knowledge of the handling of the data in the respective application. For example, if a policy is completely booked for a certain classification, the calculation of fees may only rely on the information in the policy, even if addresses do not match a certain criteria that was changed during the masking. Maybe, a renewal of such a policy will then give different results than it would have in production. These scenarios should be discussed with the testing team and the application team to clarify the desired results of those operations when working with the masked data.

Masking Banking Information

Apart from names and addresses, another common application in masking projects involves obscuring banking details like IBANs, Swift codes, or routing numbers. Banking numbers can be masked using predefined algorithms that are shipped with XDM, while lookup tables are used to mask bank names.

Summary

XDM's features outlined in this article highlight the numerous advantages of XDM as a data masking tool in any project. With easy-to-use built-in masking methods, the ability to customize these methods, and the ability to identify suitable data beforehand, XDM stands out as the ideal tool for executing projects efficiently. Its user-friendly interface, minimal need for manual intervention, and seamless integration into existing environments make XDM a compelling choice for all data masking needs.

Visit us at https://ubs-hainer.com/product/xdm/masking/ and contact us today to start your successful data masking journey.

For more information, please contact: **info@ubs-hainer.com**