MONOGRÁFICO

A common data representation model for customer behavior tracking

Un modelo de datos común para la representación del comportamiento observado del cliente

Nicolas Martin Casariego Sarasquete
Adjunct Professor
(IE Business School)

Co-Director of Customer Analytics Farm
(IE School of Human Sciences & Technology)

Reception date: 20 April 2017

Review date: 20 June 2017

Published: 1 July 2017

To cite this article: 1 de julio de 2017

Para citar este artículo: Casariego Sarasquete, N. M. (2017): A Common data representation model for customer behavior tracking, *Icono 14,* volumen 15 (2), pp. 55-91. doi: 10.7195/ri14.1078v15i2.1078

Abstract

Customer behavior has drastically changed in recent years. Being able to anticipate your customer's' behavior is the holy grail for every business leader.

Now we are able to track and store part of the observed customer behavior, thanks to digital transformation processes, social networks adoption, cloud infrastructure and big data technology availability.

This observed behavior is made up of repetitive transactions and recurring purchases, as well as navigation and interactions through digital properties, channels, devices, applications and social networks.

The present article establishes the foundation for a common data representation model of tracked customer behavior.

With this data model in place, organizations will be able to represent any given customer-centric business model allowing to analyze the most popular marketing problems: like segmentation, cross-selling, retention, etc.

Also most popular analytic and predictive software tools could be used for consumer behavior storage and analysis.

Key Words

Customer experience - Customer behavior - Customer intelligence - Behavioral data model - Customer predictive analytic - Customer behavior patterns

Resumen

El comportamiento del cliente ha cambiado drásticamente en los últimos años. Ser capaz de anticipar el comportamiento de nuestros clientes es el santo grial de todo líder de negocio.

Ahora somos capaces de trazar y almacenar parte de ese comportamiento observado del cliente, eso gracias a la transformación digital, la adopción de redes sociales, la infraestructura en la nube y la tecnología disponible de big data.

Este comportamiento observado se compone de transacciones repetitivas y compras recurrentes, así como de interacciones y navegación a través de las propiedades digitales, canales, dispositivos y aplicaciones.

El presente artículo establece las bases para un modelo de datos común para representar el comportamiento trazado del cliente.

Con este modelo, las organizaciones serán capaces de representar cualquier modelo de negocio centrado en el cliente, y analizar los problemas de marketing más comunes: segmentación, venta cruzada y retención.

También permitirá usar las herramientas analíticas y predictivas más populares para el análisis del comportamiento del cliente.

Palabras clave

Experiencia de cliente - Comportamiento de cliente - Inteligencia de cliente - Modelo de datos de comportamiento - Analítica predictiva de cliente - Patrones de comportamiento de cliente

1. Introducción

1.1. Business needs & opportunity

According to Forrester we live in the "Customer Age" (Forrester, 2011), which means that control in the organization-customer relationship has changed hands. Nothing will be the same as before.

The first step to successfully manage both customer's value and experience is to know who your customer is. Hence the need to prioritize the design and construction of a 360 degree holistic view of the customer and its behavior.

To identify, to know, to valuate, to develop, to customize, to retain, and to anticipate are the actions that will allow us to reach the promise land of a relevant, personalized and ubiquitous customer service.

According to McKinsey survey "Why Customer analytics matter" report (Fiedler, Großmaß, Roth, & Vetvik, 2016), companies that make extensive use of customer analytics are more likely to have a considerable impact on corporate performance, outperforming its competitors.

There for to gather all the customer transactions and interactions across all channels is a need; as well as to understand and to learn from that behavior, thus being able to anticipate their next move. That is our mission.

1.2. Analytic needs & opportunity

Artificial intelligence is enjoying a second life thanks to cloud technology, big data & machine learning investments on image and language recognition.

Unfortunately, a similar advance level into the customer's behavior understanding arena is not reached. Mainly because, although more sophisticated algorithms have been developed, improvement in customer behavior data representation is very small.

Almost every business intelligence, artificial intelligence and machine learning tool use a tabular factual data input format, which is an incomplete structure representation for time-series or repetitive behavioral events analysis. Few researches are focused on new data structure representation models and algorithms, like TDA topological data analysis.

The purpose of this paper is to define a common data representation for customer behavior tracking with the support of the following formalisms: relational algebra, functional dependencies, temporal algebra, spatial algebra and aggregated chronological sequences of events. It aim to be a better data structure representation for a common event-driven business problem, like customers' repetitive and recurrent behavioral actions and interactions.

2. Method

The objective of the present work is to define a common data representation model for customer behavior tracking, with the following properties:

1. Generalization
2. Time considerations for cause-effect discovery
3. Portability

A progressive combination of the mentioned formalisms and techniques will be used.

Generalization:

Generalization as a way of being able to represent any given customer-centric business model allowing to analyze the most popular marketing problems: like segmentation, cross-selling, retention, etc. Well-established and universally accepted methods for data modelling will be used: relational algebra and functional dependencies.

Both models will be defined: the conceptual data model for a generalized usage, and the logical data model for a particular given example. In both cases we will be using Entity relationship modelling -ER- (Chen, 1976), Dimensional modelling -DM- (Kimball & Ross, 2013), and Online analytical processing –OLAP- (Codd, 1993). Finally, we will use Functional dependencies –FD- (Armstrong, 1974) for data model domain's "augmentation", as well as Backus Naur Form –BNF- notation (Knuth, 1964) to complement data modelling diagrams.

Time considerations for cause-effect discovery:

We need to include temporal considerations for cause-effect discovery, and we need to be able to represent chronological sequences of events together with related entities' attributes. So we will use Temporal algebra -TA- (Allen, 1983), plus spatial algebra, combined with the hypothesis test for causality (Granger, 1980).

Portability:

Portability means that the resulting common data representation model could be implemented using the most popular data storage alternatives like structured relational SQL databases, as well as semi-structured non-SQL databases. Portability means also that consumer behavior analysis -based on the common data representation model- could be done using the most popular analytic and predictive tools such as Business intelligence and Machine learning software platforms.

To achieve this portability property we will define conceptual data representations instead of physical ones. So data modelling paradigms -like the dimensional modelling (Kimball & Ross, 2013) and OLAP properties (Codd, 1993) - will be mainly used.

3. Development

3.1. Customer knowledge objectives

Traditional customer analytics (also known as customer intelligence) starts from plain past and present customer data, and put its focus on the **who, what, when and why** dimensions of analysis. We will call this approach as "customer descriptive analytics", for which several well-known data representation models are applied, such as tabular (or vector arranged matrices), fact data input format or, even, multi-dimensional star schema data representation (Kimball & Ross, 2013).

Almost every business intelligence, visualization, statistical and data mining tool can manage this generalized data representation, which is quite good, since it enables an open and wide landscape of analytical choices.

Let's say that every single row, in the matrix or record in the fact table, represents an independent business fact. A priori, traditional business analytics is not expected to find any dependency between those business facts, and that's when behavioral analytics comes on stage.

When talking about "customer behavior analytics", focus should be on the **how and why** dimensions of analysis, thus meaning that chronologically ordered customer facts are potentially related among them. When one is able to connect the dots between those customer's facts, then customer behavior is manifested.

Web analytics is the most extended version of behavior analytics, but both data representation model, and analytical objectives are only focused on web pages navigation. Visitors, sessions and pages are the main entities; while visitors count, session length, and page views are the main metrics for analysis of this web centric paradigm.

Google Analytics and Adobe Analytics (former Omniture) are the two most popular tools, together with their own proprietary data representation and storage. Path analysis is the battle horse in this analytic space, although it is not enough to represent the whole customer behavior and knowledge.

Fortunately, with the later development of e-commerce, more sophisticated features were added to path analysis, evolving into what nowadays is known as behavior analytics.

Path analysis business goals considers conversion rate as the main success metric. Visitors' Path analysis visitors' segmentation rely mainly on customer's web traffic source and other session attributes.

A Sankey diagram is a visualization tool used to show a many-to-many mapping between origrn and destination domains. Google Analytics uses Sankey diagrams to show how traffic flows from page to page on a given web site.

Cohort analysis compares the behavior of different groups of users over a given time frame, enabling a very simple consumers' clustering and clues about consumers' life cycle. They are hard to perform using traditional web analytics tools because of the difficulty of identifying visitors, grouping them into cohorts and then tracking the behavior of cohorts over time.

E-Commerce analytics have recuperated the traditional affinity analysis (also known as basket analysis) in order to figure out which are the combinations of products that are purchased together, by adding this contribution to the existing palette of behavior analytic tools.

Despite of all the above listed improvements, we have not reach a consensus on an open and common behavioral analysis data representation. Even worse, we are not even able of reusing existing powerful data analytic tools on top of this data models.

The objective of my work is to create a common data representation model for tracking customer behavior, in order to enable the analysis and discover of customer behavior patterns. Thus evolution, alterations, similarities or absence of customer behavior could be analyzed and alerted. With such data model we can analyze customer behavior using well proven customer analytics algorithms, such as time-series analysis, clustering, classification, attribution, affinity, etc.

3.2. Customer knowledge considerations

Given the fact that information can only be distillated from available data, when we talk about customer behavior knowledge, we will refer only to observed customer behavior, which is just a part of it.

What you observe is what you know. To lead with this customer behavior incompleteness it is necessary to gather as much customer data as possible in order to extend our customer knowledge.

Having a customer identity strategy is the first step in the complex process to build a 360 degrees customer profile. We will therefore start by discussing this critical part of customer behavior analytics.

The second element to be taken into account is that for us, the observed customer behavior are all the customer events and actions that we can recognize, track and store about him using his/her customer identity. Even when we already know that this recorded behavior is necessarily incomplete.

The third element to be discussed is the "absence" of a recurrent expected behavior. Within a stable and controlled customer tracking environment, no data also implies data.

Fourth and last element under discussion will be the identified "business meaning" inside observed customer behavior. Positive, neutral or negative effect categorization of a specific customer behavior will help to better understand what is the customer intention.

3.3. Customer identity

Customer behavior analytics requires of a robust and consistent customer tracking environment based on two main pillars: customer identify and customer event tracking across all the channels.

Customers have several types of Identity:

- Personal identity
- Transactional Identity
- Social Identity
- Digital Identity
- Household Identity

Associated with customer identity there are plenty of different identifiers (instruments or supports of Identity) such as: social security number, passport, electronic signature, biometrics, Facebook ID, loyalty card, credit card, SWIFT account, email, mobile phone, postal address, etc.

The biggest challenge for organizations is to be able to identify and remember that image of the customer, in addition to recognizing in each channel, interaction, experience or transaction.

3.4. Customer 360 profile

Customer 360 degrees holistic vision is a gradual and ongoing process that builds, enriches and refines the customer's profile. Each data source, channel, transaction and interaction opportunity are the components that allow to build customer knowledge, profile, preferences and behavior.

Customer 360 profile is the union of several partial views, such as transactional profile, declared profile, social profile, interactive profile, third party profile, etc.

3.5. Customer behavior causality

With the aim of simplification, we state that customer behavior is the observed consequence of its habits, choices and decisions.

If it is considered that the customer is a "homo economicus", consequently its habits, choices and decisions are based on a rational selfishness (principle that means that an action is rational if and only if it maximizes one's self-interest).

Behavioral economics also studies the effects of psychological, social, cognitive, and emotional factors on the economic decisions of individuals, as well as the consequences they have on market prices, returns, and resource allocation.

Those psychological, social, cognitive, and emotional factors are hard to trace and measure, but we will take into consideration the fact that customer behavior could be potentially influenced by the environment, the market, the culture, the brands and other consumers.

Thus "observed customer behavior" is the systematic traceability of those habits, choices and decisions, plus other related events that could have an impact in customer behavior. At the end, gathered data about the environment and customers' actions are the best guess and representation of customer behavior that can be done in an information system.

Brands tend to trust, and they are probably right, that their marketing initiatives and their promotional stimulus influence customer behavior. That's why the art and science of measuring brand advertisement, communications, promotions, customer service and loyalty initiatives was developed; despite the fact that only some personalized direct actions can be measured, being the rest indirect measurement through channel or message attribution.

3.6. Traditional marketing & customer analytics

Traditional marketing & customer analytics usually consider time effect just as a time dimension reference, as any other dimensions like market, product, etc., are. This is partly because almost all marketing & customer phenomena are considered "stationary" or one shot action scenario.

A quick and not exhaustive review of current state of the art of marketing and customer analytics best practices will be made, with the main purpose of demonstrating that there are few examples of the use of time factors.

- **Customer descriptive analysis**

  Also known as profiling analysis, basically consists of analyzing all customer attributes, one by one, by using statistics algorithms. Time dimension is only occasionally used.

- **A/B testing**

  The most popular "ceteris paribus" experiment in marketing with two variants, A and B, which respectively represent both the control and the variation in the controlled experiment. This type of test uses both statistical hypothesis testing and multivariate analysis. Time dimension is hardly used.

- **Campaign lift analysis**

  Campaign lift analysis measures the performance of a targeting model at predicting or classifying response cases of a marketing campaign. It obtains an enhanced response (with respect to the whole population) measured against a random choice targeting model. Typically using classification algorithms. Time dimension is again only occasionally used.

- **Customer segmentation**

  Customer segmentation is the very basic technique used by marketers to group customers in segments of similar characteristics. Traditionally by using clustering algorithms, cohort algorithms, RFM (Recency, Frequency, Monetary) algorithms, attrition scoring algorithms, or purchase propensity scoring algorithms. Time dimension is hardly used outside of the attrition scores.

- **Market basket analysis**

  Market basket analysis is used by retailers to understand the purchase behavior of customers in terms of which products or articles are purchased together within a single transaction. This information can then be used for both cross-selling and up-selling purposes. Usually by using affinity and association rule learning algorithms. Time dimension is only occasionally used.

There are a few exceptions that take time considerations or time series factors into account within the analysis, such as:

- **Financial & Sales reports**

  These are the most basic business intelligence tools. Typically using multidimensional (OLAP) algorithms. The time dimension is used in this case for evolution analysis purposes, as well as for month length normalization. Time series analysis is very commonly used for short term business forecasting like future demand, etc.

- **Marketing attribution analysis**

  According to IAB (IAB, 2012), marketing attribution is the process of identifying a set of user actions (events) that contribute in some manner to a desired outcome, and then assigning value to each of those events. In digital advertisement, attribution is done at a user-specific level, where a consistent user identifier can be established across all analyzed events. Digital marketing attribution typically uses logistic regression algorithms. Time dimension and time effect are critical in this case, since marketing attribution provides a level of understanding of what combination of events in what specific order influences individuals to engage in a desired behavior (conversion). Another time consideration would be the halo effect of broadcasting stimulus, like TV ads.

Almost all marketing & customer phenomena are considered either "stationary" or one shot action scenario, not because a real conviction, but because both practical limitations and constraints to analyze facts and events that go much farer away.

That is the reason why a new common data representation model of customer behavior is needed, which can also cover chronological sequences of events for marketing & customer analysis.

3.7. Customer behavior data universe

Let's start building the customer behavior data universe. Just for clarity and simplicity aims the conceptual model of our universe will be described by using Chen's ER entity relationship diagram (Chen, 1976), which will enable to map the real world into an abstract model that we can handle and operate.

In our customer behavior data universe, CBDU for short here in after, some of the components will be enumerated by extension, despite the fact that we will define a generalized conceptual model that could potentially be extended to more data elements.

- **CBDU Entities**

  Several entities take part of our model. Our main entity is customer and it will therefore be the focus of our attention. Other relevant entities are: product or service, channel, campaign, content, offer, agent and session. The selection of a different set of entities or even of a different main entity would not affect the whole structure.

- **CBDU Context entities**

  There are two special named entities in the model: time and geo-location. They are context entities for time-spatial contextual reference of the whole customer behavior model.

- **CBDU Entity relationships**

  The two main entity relationships in the model are:

1. The events which represents a foot print of an entity's action like the customer. They can potentially affect to other entities.

2. The metrics which represent a time based entity's key performance indicator evolution.

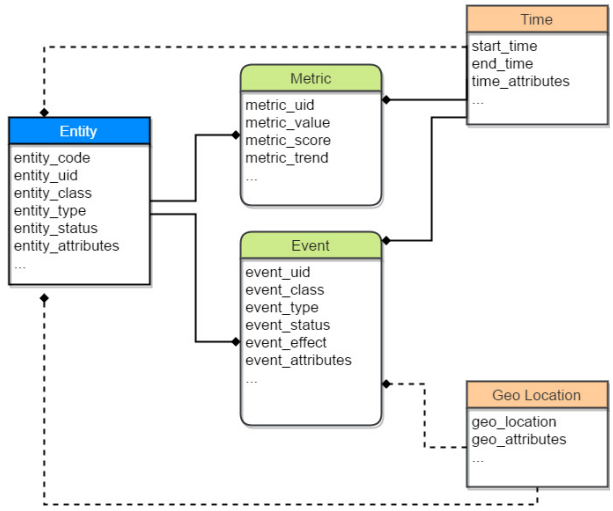

Figure 1: *CBDU Customer behavior data generalized metadata model ER entity relationship partial diagram.*

```
universe ::=    events, entities, context, metrics

events ::=                  event { event }
entities ::=    entity { entity }
metrics ::=     metric { metric }

event ::=                   event_instance_uid,
                event_start_time, event_end_time, [event_geolocation,]
                event_class, event_type, event_status, event_effect,
                event_count, event_magnitude, event_unit,
                event_nature, event_result, event_likelihood,
                entity_reference {,entity_reference} {,attribute}
entity_reference ::=        entity_code, entity_instance_uid
event_effect ::="positive" / "neutral" / "negative"
event_nature ::="fact" / "no_fact" / "predictive"
event_result ::="neutral" / "guess" / "fail"

entity ::=                  entity_code, entity_instance_uid,
                entity_class, entity_type, entity_status { , attribute }
entity_code ::= "customer" / "product" / "channel" / "campaign" /
                "content" / "offer" / "agent" / "session"
attribute ::=   "(" attribute_name ";" attribute_value ")"

context ::=     time_context / geo_context
time_context ::=start_time, end_time, entity
geo_context ::= geo_location, entity

metric ::=                  metric_instance_uid,
                entity_code, entity_instance_uid,
                metric_start_time, metric_end_time,
                metric_class, metric_type, metric_status,
                metric_value, metric_previous_value,
                metric_score, metric_previous_score,
                metric_trend, metric_previous_trend
```

The CBDU metadata model formalization using Backus Naur Form notation (Knuth, 1964) is the following:

As it might be observed this conceptual model is a generalized metadata definition that can be converted into a logical data model. Using both a transformation process and a metadata description for every particular business model you would like to build.

There is also no compromise with a specific data storage representation, since the described metadata model focuses only on the **functional dependencies** of the metadata model as we are going to see later on.

We explicitly keep open the entity list that could be involved on every event; as well as the list of attributes of each entity, by using **attribute-value pairs**, to be adapted and mapped to every single business problem. In that way everyone could map it with his favorite database representation, like relational databases, columnar databases, or No-SQL databases.

For example, we will instantiate our BCDU metadata model into a specific business model as follows:

- Each customer event is related to a customer unique identifier and a campaign unique identifier
- Each customer event is related to a concrete time window and a geo-location
- There are marketing offers linked only to specific time windows
- Only customer's metrics are being measured along time periods

The following Entity relationship –ER- diagram reflects the previous business model definition, although preserving all the properties of the generalized metadata model, in terms of both functional dependencies and analytical richness.

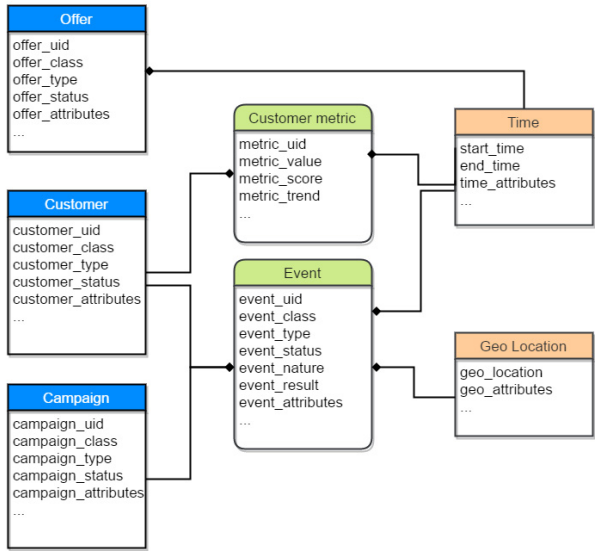

Figure 2: *CBDU Customer behavior data universe ER metadata diagram mapped to a specific business model.*

The final aim of a common data representation model of customer behavior is to enable a business analyst to use his complete analytic toolset against such data model, in order to extract the desired customer behavior knowledge. Let's start with a multidimensional approach.

3.8. Multidimensional data model

There is a partial direct translation between our previous data model representation and a multidimensional data representation model more suitable for OLAP analytic tools.

Edgar Codd worked up 12 rules for defining **OLAP** (Online Analytical Processing), a standard data processing which allows consolidation and analysis of data in a **multidimensional space**.

In our example, Codd's (Codd, 1993) relevant rules are the following ones:

- Rule #1: Multidimensional conceptual view
- Rule #6: Generic dimensionality
- Rule #7: Dynamic sparse matrix handling
- Rule #8: Unrestricted cross-dimensional operations
- Rule #10: Intuitive data manipulation
- Rule #12: Unlimited Dimensions and aggregation levels

Let's use Ralph Kimball's **star schema** representation of dimensions, hierarchies and measures. The star schema splits business process data into facts, which hold the measurable, quantitative data about a business, as well as dimensions that are descriptive attributes related to fact data.

In our example, we will do the following correspondence

ER Diagram	Star schema
Entity relationship	Fact table
Entity	Dimension lookup table
Entity hierarchies	Dimension hierarchies
Entity attribute	Dimension
Event non numeric attribute	Dimension
Event numeric attribute	Measure

Thus we obtain the following star schema mapping:

Data item	Category
customer_class	Dimension
customer_type	Dimension
customer_status	Dimension
customer_attributes	Dimension
campaign_class	Dimension
campaign_type	Dimension
campaign_status	Dimension
campaign_attributes	Dimension
time_attributes	Dimension
geo_attributes	Dimension
event_class	Dimension
event_type	Dimension
event_status	Dimension
event_nature	Dimension
event_result	Dimension, Classification goal
event_attributes non numeric	Dimension
event_attributes numeric	Measure
event_count()	Measure

And the following star schema diagram only for the red colored entities and relationships:

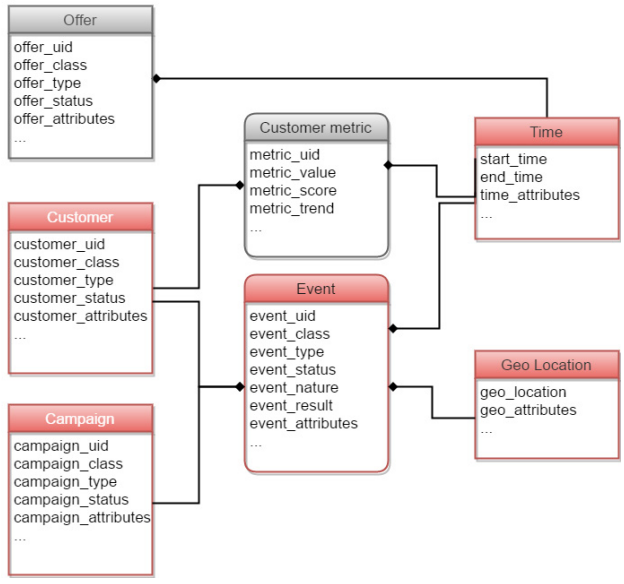

Figure 3: *CBDU Customer behavior data model's Star schema (in red color).*

3.9. Functional depedencies

Armstrong's axioms (Armstrong, 1974) are a set of axioms (or, more precisely, inference rules) used to infer all the functional dependencies on a relational database.

Given a relation **R**, a set of attributes **X** in **R** is said to functionally determine another set of attributes **Y**, also in **R**, (written **X** → **Y**) if, and only if, each **X** value in **R** is associated with precisely one **Y** value in **R**; **R** is then said to satisfy the functional dependency **X** → **Y**.

In other words, a functional dependency **X** → **Y** means that the values of **Y** are determined by the values of **X**. Two tuples sharing the same values of **X** will necessarily have the same values of **Y**.

In this example we found out the following functional dependencies:

- **event_uid → event_attributes**
- **event_uid → customer_uid, campaign_uid**
- **event_uid → start_time, end_time, geo_location_uid**
- **customer_uid → customer_attributes**
- **campaign_uid → campaign_attributes**
- **start_time, end_time → time_attributes**
- **geo_location_uid → geo_attributes**

Going through Armstrong's axioms:

- **Transitivity:**

  If **X** → **Y** and **Y** → **Z**, then **X** → **Z**

- **Augmentation:**

  If **X** → **Y**, then **XZ** → **YZ** for any **Z**

- **Union:**

  If **X** → **Y** and **X** → **Z**, then **X** → **YZ**

- **Pseudo transitivity:**

  If **X** → **Y** and **YZ** → **W**, then **XZ** → **W**

We can expand or augment our original event attributes scope to the following one:

- **event_uid** → **event_attributes,**
  customer_attributes,
  campaign_attributes,
  time_attributes,
  geo_attributes

It is like having an "augmented" event's fact table. Thus any kind of multidimensional analysis (slice, dice, drill down, roll up, etc.) can be performed based on the dimensions and measures of this partial model of customer behavior.

3.10. Customer metrics' evolution

In our approach to a customer behavior generalized data representation model, a simple and standard set of customer's metrics or key performance indicators will be defined, which will reflect any critical change in customer profile along time. These critical metric changes are not necessarily covered by the customer entity attributes, neither by the customer behavior events and actions.

Resuming the generic metric relationship specification in Backus Naur Form notation, in order to go into deeper details for the specific customer's metric relationship entity:

```
Customer_metric ::=        metric_instance_uid,
                           "customer", customer_uid,
                           metric_start_time, metric_end_time,
                           metric_class, metric_type, metric_status,
                           metric_value, metric_previous_value,
                           metric_score, metric_previous_score,
                           metric_trend, metric_previous_trend

metric_class ::=           "kpi" / "attribute"

metric_type ::=            "value" / "risk" / "lifecycle" / "mot" /
                                "privacy" / "loyalty" / "social" / "content"

metric_score ::=           "high" / "average" / "low"

metric_trend ::=           "up" / "neutral" / "down"
```

These metrics evolution standardization helps business analysts to understand customer profile evolution between arbitrary periods of time. And makes easier to present alerts in an executive dashboard or reporting tool, as well as to be integrated into business rules engines.

3.11. Temporal algebra

Allen's interval algebra (Allen, 1983) defines 13 possible relations between two intervals. These are the temporal operators defined by the W3C RDF Resource Description Framework for two intervals <t1, t2> and <t3, t4>:

- Before or Sequence (SEQ): joins <t1, t2> and <t3, t4> where t2 < t3.
- Meets: modified SEQ operator such that t2 = t3.
- Overlaps: modified SEQ operator such that t2 > t3.
- Starts: modified SEQ operator such that t1 = t3 & t2 < t4.
- During: modified SEQ operator such that t1 > t3 & t2 < t4.
- Finishes: modified SEQ operator such that t2 = t4 & t1 > t3.
- Equal: modified SEQ operator such that t1 = t3 & t2 > t4.

In contrast, an RDF stream model defined on time points, enables to distinguish only between two temporal relatedness (i.e., "before" and "equal"). Finally, a comparison between a time point and a time interval enables eleven different relations to be distinguished (i.e., all except "overlaps" and "equals").

In order to be able to perform temporal comparisons between a time interval <t1, t2> and a time point <t3>, the following temporal algebra (TA) operators has been created as follows:

- Before: joins <t1, t2> and <t3> where t2 < t3.
- During: joins <t1, t2> and <t3> where t1 <= t3 & t3<= t2.

3.12. Customer metrics inheritage

When a customer event occurs, any business analyst would like to associate easily all customer metrics' evolution within the same time frame window, in order to have a more complete understanding of both customer attributes and potential causes for an outcome effect produced by such event.

In our example, we found out the following functional dependencies:

- **event_uid → customer_uid**
- **customer_uid, event_time →TA→ { customer_metric_metric_uid }**
- **{ customer_metric_metric_uid } → { customer_metric_attributes }**

Going through Armstrong's axiom:

- **Pseudo transitivity:**

  If **X → Y** and **YZ → W**, then **XZ → W**

Going through pseudo-SQL with temporal algebra (TA):

- Select **{ customer_metric.metric_uid }**

  Where e**vent.customer_uid = customer_metric.customer_uid**

  And

  (Before(**customer_metric.start_time, customer_metric.end_time, event.time)**

  Or

  During**(customer_metric.start_time, customer_metric.end_time, event.time)**)

We can expand or augment our original event attributes scope to the following one:

- **event_uid → event_attributes,**
  customer_attributes,
  campaign_attributes,
  time_attributes,
  geo_attributes,
  {customer_metric_atributes}

3.13. Temporal cause-effect relation

What is causation?

The cause-effect relation affects all aspects of consumers. Two philosophers who contributed a great deal to our understanding of causation are David Hume and John Stuart Mill.

For David Hume (Khoo, Chan, & Niu, 2002), causation comprises the following three conditions:

1. Contiguity in time and place
2. Priority in time
3. Constant conjunction between the cause and the effect

When a person finds, from experience, that an event of kind **A** is always followed by an event of kind **B**, the person comes to conclude that event **A** causes event **B**.

But John Stuart Mill (Khoo et al., 2002) argued that constant conjunction is not sufficient for inferring causation, unless the conjunction is also unconditional. He described four methods by which one can determine that **A** causes **B**.

Perhaps the most influential of these ideas is the method of difference, which has been extended to distinguish between necessary and sufficient causes.

Ordinarily, regressions reflect "mere" correlations, but Clive Granger (Granger, 1980) argued that causality in economics could be tested by measuring the ability to forecast (predict) the future values of a time series using prior values of another time series.

The intuition behind Granger-causality (also known as predictive causality, which does not mean necessarily true causality) is the following one: "We say that a variable **X** that evolves over time Granger-causes another evolving variable Y if predictions of the value of **Y** based on its own past values and on the past values of **X** are better than predictions of **Y** based only on its own past values".

Granger defined the causality relationship based on two underlying principles:

1. The cause occurs prior to its effect

2. The cause has unique information about the future values of its effect

Given these two assumptions about causality, Granger proposed to test the following hypothesis for identification of a causal effect of **X** on **Y**:

$$\mathbf{P[\ Y(t+1) \in A\ /\ UI(t)\ ] \neq P[\ Y(t+1) \in A\ /\ UI\text{-}X(t)\ ]}$$

Where **P** refers to probability, **A** is an arbitrary non-empty set, and **UI(t)** and **UI-X(t)** respectively denote the information available as of time t in the entire universe, and that in the modified universe in which **X** is excluded. If the above hypothesis is accepted, we say that **X** Granger-causes **Y**.

All these attempts to define, identify and predict causality in different domains and with different purposes raised some controversial discussions along time.

In this work, I prefer to extract some common principles and concepts from previous theories, in order to be applied into a pragmatic approach to define,

identify, store, retrieve and predict customer observed behavior. Such behavior could be potentially affected by the marketing initiatives carried out by the brands, the market or the environment over a given period of time.

3.14. Time-spatial attributes inheritage

Customer behavior could be potentially influenced by the environment, the market, the culture, the brands and other consumers. But these causes are usually linked to entities -like campaign, offer, channel, agent, market- within a specific geo-location and time frame window, rather than linked to specific customers.

Using the following Hume's time-spatial conditions for causation:

1. Contiguity in **time and place**

2. Constant **conjunction** between the cause and the effect

In our example, we found out the following functional dependencies:

- **event_uid → event_time**

- **event_time →TA→ { offer_uid }**

- **{ offer_uid } → { offer_attributes }**

Going through Armstrong's axiom:

- **Transitivity:**

  If **X** → **Y** and **Y** → **Z**, then **X** → **Z**

Going through pseudo-SQL with temporal algebra:

- Select **{ offer.offer_uid }**

  Where **event.offer_uid = offer.offer_uid**

  And During (**offer.start_time, offer.end_time, event.time**)

Even when it is not the case here, analog procedure could be followed to perform spatial comparisons between a geo-location area <{latitude, longitude}> and a geo-location point <latitude, longitude>. The following spatial algebra (SA) operator will be defined:

- Contains: joins <geo_area> and <geo_point> where <geo_point> belongs to <geo_area>.

We can augment again our original event attributes domain to the following one:

- **event_uid → event_attributes,**

  customer_attributes,

  campaign_attributes,

  time_attributes,

  geo_attributes,

  {customer_metric_atributes},

  { offer_attributes }

Now a generalization of the customer behavior generalized data representation model can be made to obtain an augmented attributes domain using functional dependencies, temporal algebra and spatial algebra:

- **event_uid → event_attributes,**

  { entity_attributes },

  time_attributes,

geo_attributes,

→TA→ **{ entity_metric_attributes },**

→TA→ **{ entity_attributes },**

→SA→ **{ entity_attributes }**

3.15. Business semantics

A great difference between customer behavior analytics and general event based analytics is that customer actions or events have a well identified business semantics.

Some customer actions are considered to have a positive effect, while others are considered to have a negative effect or no effect (neutral). Being able to categorize the customer actions and events will help to answer the **how & why** business questions in a more accurate way.

In our customer behavior generalized data representation model there is place for a business semantic representation formalized as:

```
event_effect ::="positive" / "neutral" / "negative"
```

Please note that we consider **event_effect** as a **classification goal** for data mining purposes. This is a first step in order to understand the effects of certain actions and events in customer behavior.

Now predictive analytics against the customer actions and events can be performed, with a business classification goal in place, and an augmented set of extra attributes that are added by using functional dependencies FD, temporal algebra TA, and spatial algebra SA.

Moreover, we can learn from past customer actions and events to predict or classify future actions and events in advance. We support in our customer behavior generalized data representation model a closed loop learning procedure based on the following data structure:

```
event_nature ::="fact" / "no_fact" / "predictive"
event_likelihood ::= 0 / … / 100
event_result ::="neutral" / "guess" / "fail"
```

In our metadata model coexists customer events of different nature using **event_nature** attribute; real customer events categorized as "**fact**", with predicted customer events categorized as "**predictive**". Those predicted customer events have an associated likelihood (between zero and 100) of being true, which have been provided by the predictive algorithm and stored in the **event_likelihood** attribute.

The learning aspect of this closed loop procedure consists in performing a regular evaluation of the predicted events accuracy, placing the evaluation result in the **event_result** attribute. So, we will assign "**guess**" or "**fail**" values depending on the later appearance of the predicted customer action or event. The learning and evaluation procedure is not covered in this article.

As soon as we have a well trusted predictive algorithm in place, we will be able to add customer actions and events of a different nature. When a predicted event with a high likelihood did not happen, we will switch the event's nature from "**predictive**" to "**no_fact**" (absence of customer action), meaning that there is a missing event which have been expected. This creates an early alert about a change in the customer behavior.

3.16. Chronological sequences of events

While customer actions and events like "navigate", "login", "purchase" and "click-through" are punctual evidences of its behavior; a chronological sequence of customer actions and events like ["login", "navigate", "add-cart", "purchase", "pay"] is a clearer picture of customer behavior.

Our generalized customer behavior augmented data representation model contains all the observed customer actions and events including the direct and indi-

rect related entities attributes and metrics. That is every single evidence of customer behavior, which can be sorted in chronological order.

To build our chronological sequences of events, we need to define the look back time window. There are at least three hierarchical levels of time frame windows to be used, depending on our analytic zoom focus:

- **Device session:**

  Typically, this device or channel's session lasts from minutes to hours. Our attention will be focused on the UX user experience. An example of this UX session is an e-Commerce website session.

- **Omni-channel session:**

  Typically, this multi device or omni-channel's session lasts from days to weeks. Our attention will be focused on the CX customer experience. An example of this CX session is a summer promotional campaign across email, social media and offline channels, grouped in an omni-channel session.

- **Lifecycle session:**

  Typically, this lifecycle session lasts from weeks to months. Our attention will be focused on the customer lifecycle experience. An example of this CL session are the recurring purchases in a supermarket, including every customer action or event across all channels and touch points grouped in a lifecycle session.

All these three businesses focus relay on an additional data processing known as "sessionize", that groups and labels every action or event under a session dimension. Both the beginning and end of a specific session depends on the average time lagging between events, and are always based on a session time-out rule.

The look back time window and time unit depend on business focus and needs. Now we are ready to build dynamic chronological sequences of events using multidimensional operators on top of our customer behavior generalized augmented data representation model.

By using OLAP graphical query interface, all customer events are shown placing customer dimension on rows and time dimension on columns, counting the number of events and displaying ticks colored by event type dimension, filtering by year=2016 and slice by quarter.

Filter: none Page: none	[Event_time] on Rows slicer by [event_time].[quarter].[2016]
[event_customer_uid] on Columns	[event_count] on Cells color by [event_type]

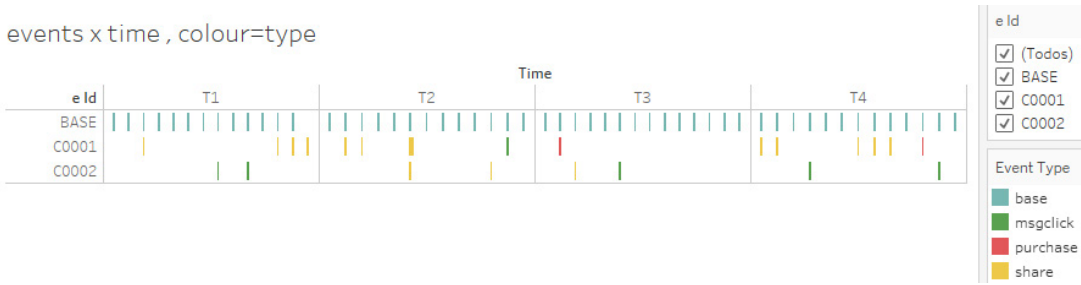

Figure 4: *Customer events visualization per each customer along time colored by event type.*

Figure 5: *Customer events visualization per each customer along time named by event type.*

Figure 6: *Customer events visualization per each customer along time named by event* channel.

OLAP aggregation functions may be used for string typed attributes concatenation to build an extended set of chronological sequences of events and event's details. Let's assume that our look back time window will be one year before the current customer event (expressed in Unix epoch timestamp with seconds precision in GMT time zone).

```
look_back_time_window = 86.400
```

Going through pseudo-SQL with string aggregation function:

- Select **event_customer_uid**, StringAggregation(**event_uid , ">**") as **event_sequence**

  Group by **event_customer_uid**

  Order by **event_time**

  Where **event.time** Between(**today() , today() - look_back_time_window**)

We can recreate the same aggregated sequences as in the visualization diagram.

- Select **event_customer_uid**, StringAggregation(**event_type , ">"**) as **event_type_sequence**

  Group by **event_customer_uid**

  Order by **event_time**

  Where **event.time** Between(**today() , today() - look_back_time_window**)

- Select **event_customer_uid**, StringAggregation(**event_channel , ">"**) as **event_channel_sequence**

  Group by **event_customer_uid**

  Order by **event_time**

  Where **event.time** Between(**today() , today() - look_back_time_window**)

Obtaining the following augmented attributes:

- "C001", "share>share>share>purchase>"
- "C002", "purchase>msgclick>msgclick>"

- "C001", "web>email>social>email>"
- "C002", "web>web>email>web>"

Going through Granger-causality relationship underlying principles:

1. The cause **happens prior** to its effect

2. The cause has unique information about the future values of its effect

Going through Armstrong's axiom:

- **Pseudo transitivity:**

  If **X** → **Y** and **YZ** → **W**, then **XZ** → **W**

We can augment again our original event attributes domain to the following one:

- **event_uid, look_back_time_window → event_attributes,**
  customer_attributes,
  campaign_attributes,
  time_attributes,
  geo_attributes,
  { customer_metric_atributes },
  { offer_attributes },
  event_sequence,
  event_type_sequence,
  event_channel_sequence

4. Conclusions

We finally arrive to the generalization of the common data representation model of customer behavior, to obtain an augmented set of attributes by using functional dependencies FD, temporal algebra TA, spatial algebra SA, and chronological sequences of events CSE:

- **event_uid, look_back_time_window → event_attributes,**
  { entity_attributes },
  time_attributes,
  geo_attributes,
  →TA→ { entity_metric_attributes },
  →TA→ { entity_attributes },
  →SA→ { entity_attributes }
  →CSE→ { event_sequences }

This de-normalized data model will allow to perform any OLAP analysis, or predictive analysis if data are exported into a tabular structure.

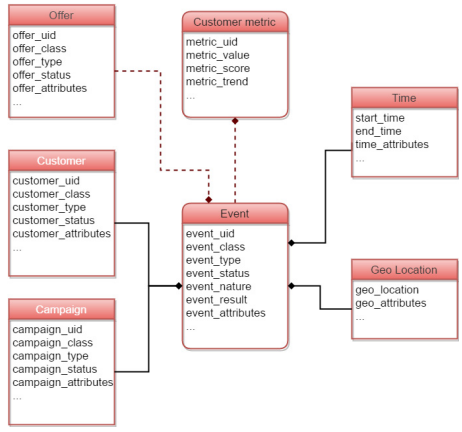

Figure 7: *CBDU Customer behavior generalized augmented metadata model's Star schema (in red color).*

This customer behavior generalized data representation model enables to represent general business models based on observed customer behavior, made of repetitive transactions, recurring purchases and payments, navigation and interactions across disparate digital properties, channels, devices, applications and social networks.

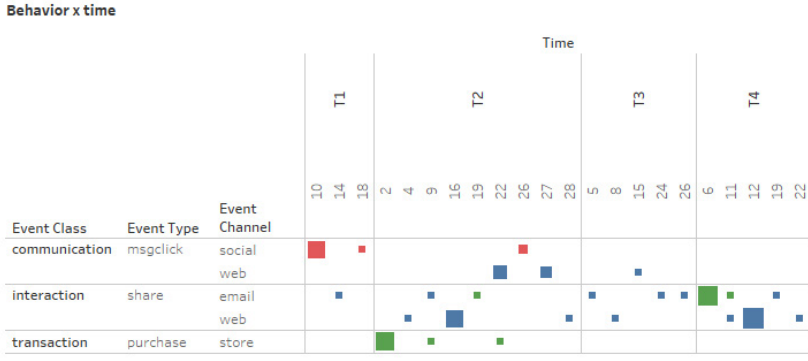

Figure 8: *Customer behavior generalized augmented data representation model shows behavioral patterns with different effects.*

This work will be the foundation for the next natural steps in customer predictive analytics:

- **Customer behavior patterns:**

  Being able to identify, represent, store, index, manage, access, query, filter and visualize customer behavior patterns located in customer behavior generalized data representation models.

- **Customer behavior patterns predictive analysis:**

  Being able to integrate and manage customer behavior patterns with popular analytical tools, plus manage customer behavior patterns fuzzy matching algorithms together with machine learning algorithms.

4.1. Data model limitations

The present data representation model presents some scope and granularity limitations:

- **Scope limitations:** The current data representation model is mainly oriented to identify recurrent observable customer behavior patterns. That's the case in some of the following business models: Business-to-Business models and Business-to-Consumer models (within the following categories: financial services, insurance, credit cards, e-commerce, telecommunications, digital media, healthcare services, retail, cosmetics, consumables, etc.). Outside of this scope, the knowledge representation capabilities of the data model are limited.

- **Granularity limitations:** The current data representation model is oriented to identify principal finite patterns in observable customer behavior. That's the case when you have tens of transactions' categories, tens of tracked customer actions, covered by hundreds of temporal sequences of events. Outside of this data granularity, the visual expression capabilities of the data model are limited.

References

Allen, J. F. (1983). Maintaining knowledge about temporal intervals, *Communications of the ACM*, 26(11), 832-843. https://doi.org/10.1145/182.358434

Armstrong, W.W. (1974). Dependency Structures of Database Relationships. *Proc. 1974 IFIP Congress, North Holland, 580-583*.

Chen, P. P. (2002). The Entity Relationship Model — Toward a Unified View of Data. *Software Pioneers*, 311-339. https://doi.org/10.1007/978-3-642-59412-0_18

Codd, E. F., Codd, S. B., & Salley, C. T. (1993). Providing OLAP (On-line analytical processing) to user-analysts: an IT mandate. Ann Arbor, MI: Codd & Associates.

Fiedler L., Großmaß T., Roth M., & Vetvik O. J. (2017). Why customer analytics matter. Retrieved from http://www.mckinsey.com/business-functions/marketing-and-sales/our-insights/why-customer-analytics-matter

Forrester. (2011). Age of the Customer. Retrieved from https://go.forrester.com/age-of-the-customer/

Granger, C. W: (1980). Testing for causality: A personal viewpoint. *Journal of Economic Dynamics and Control*, 2, 329-352. https://doi.org/10.1016/0165-1889(80)90069-X

IAB. (2012). IAB Attribution Primer. Retrieved from http://www.iab.net/media/file/AttributionPrimer.pdf

Khoo, C., Chan, S., & Niu, Y. (2002). The Many Facets of the Cause-Effect Relation. *The Semantics of Relationships Information Science and Knowledge Management*, 51-70. https://doi.org/10.1007/978-94-017-0073-3_4

Kimball, R., & Ross, M. (2013). *The data warehouse toolkit the complete guide to dimensional modeling*. New York, NY: Wiley.

Knuth, D. E. (1964). Backus normal form vs. Backus Naur form. *Communications of the ACM*, 7(12), 735-736. https://doi.org/10.1145/355588.365140