



kiwibot

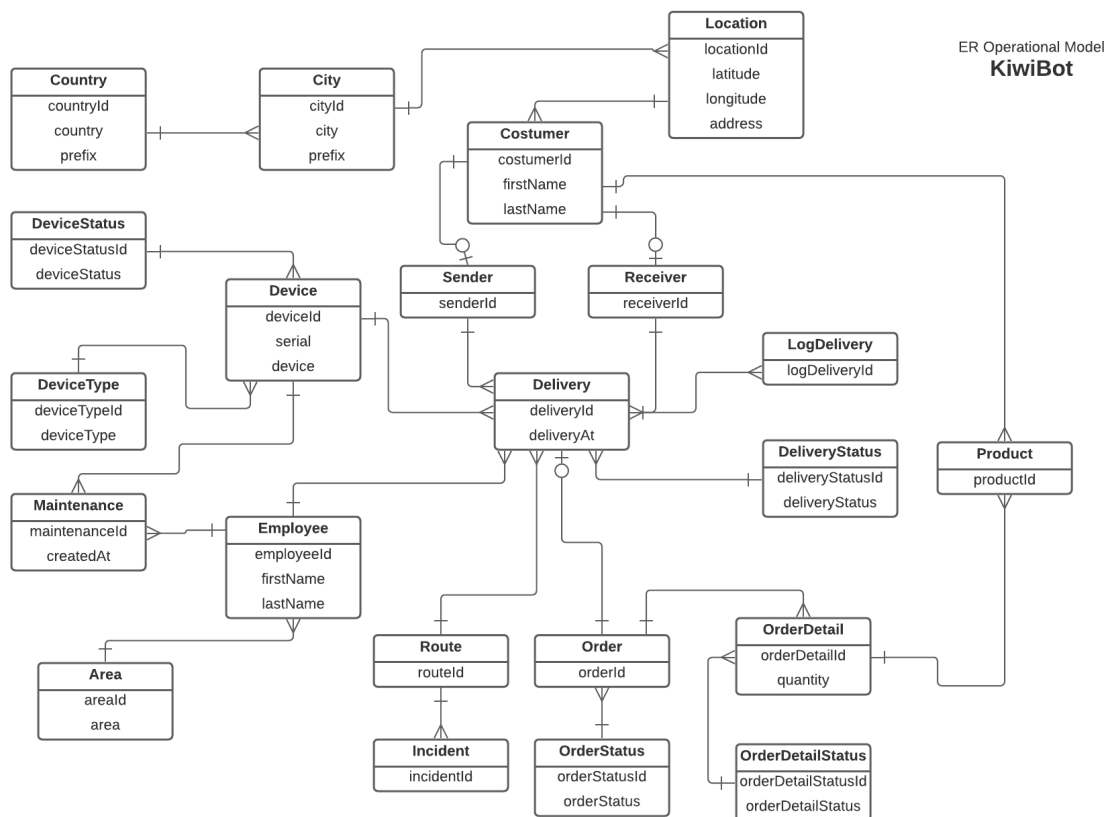
DATA ARCHITECT CASE STUDY

Assignment 1: — Data Architecture

Considering that Kiwibot's operating model is based specifically on deliveries, the main reason why the devices were designed, the data model focuses on the entity called: **Delivery**. Each of the entities of said model is described below:

- **Delivery.** Entity in charge of storing each of the deliveries. They must be framed in a temporality and geographical location.
- **Customer.** Stores the basic information of each of the clients. A customer is considered as a person who can send or receive a package. This customer can have both roles, but in different deliverables.
- **Sender.** Customer who sends the package.
- **Receiver.** Customer who receives the package.
- **LogDelivery.** Stores each of the incidents that occur with the delivery. This entity provides information for monitoring and auditing of each package.
- **DeliveryStatus.** Base information of the delivery statuses.
- **Location.** It stores information about each of the geographical points related to a Customer that receives or sends a package.
- **City.** It corresponds to the cities of a country where the organization operates.
- **Country.** It is up to the countries.
- **Device.** Basic information about the devices used by Kiwibot and who deliver the packages.
- **DeviceStatus.** Device status information.
- **DeviceType.** Stores information about device types.

- **Maintenance.** Entity in charge of storage for the control of the inputs and outputs for maintenance of the devices.
- **Employee.** Entity in which information is stored on the employees who schedule deliveries and those who actively participate in the maintenance of the devices.
- **Area.** Basic information on the areas of the organization where employees work.
- **Order.** Entity in which all package delivery orders are registered.
- **OrderDetail.** Stores the details of each of the delivery orders.
- **OrderStatus.** Stores order statuses.
- **OrderDetailStatus.** Stores the statuses of the items that correspond to the detail of an order. Note: The combination of the detail statuses, meet the order status.
- **Product.** Stores the information of the product owned by the client (Customer) and that is part of a delivery order.
- **Route.** Entity in charge of storing the routes established between the *Sender* and *Receiver* of the package in each of the deliveries.
- **Incident.** It stores the incidents as a log that have occurred in each of the established routes.



This diagram was built using Lucidchart

Assignment 2: — Data Implementation

Relational databases offer the opportunity to store information consistently and congruent with the operation of a specific company. This means that an engine must be used that allows to maintain these aspects without too much construction using special algorithms and additional code.

I would start by building the physical data model, starting from the proposed entity-relationship model and adding the attributes required for the operation. I would use an engine like MySQL, which is very popular for consistent storage using the appropriate keys and indexes (Primary, Unique, and Foreign). This database allows, among other things, to access from applications designed for mobile devices, and to keep each of the actors aware of situations related to deliveries, devices, packages, orders, etc.

The activities to be followed in favor of the database that will support Kiwibot's main operation are described below:

1. Assurance with the entire Operational team that the model fully meets all the needs and requirements of Kiwibot. This work should be done in specific meetings with stakeholders and information managers in the business areas using practices such as:
 - Interviews.
 - Operational Testing using questions.
 - Hypothetical scenarios of situations without the use of technology.
2. Survey and specification of attributes in each of the entities of the entity-relationship model.
3. Construction of the physical data model, using a CASE type tool, which allows the generation of the Scripts automatically for the creation of the Database and each of its components. One of the best is called *PowerDesigner* from the manufacturer *Sybase*. The use of this type of technology will allow us to optimize resources and to be able to build what is required in a fast and timely manner, with a reduction mainly in time and cost.

However, it is worth clarifying that the source of information for reports and decision making using the strategic KPIs that are defined at the organizational level, must be another using repository in BIG QUERY, with data extraction and transformation technology from MySQL, such as for example Dataflow and CDC capabilities. This practice avoids congesting the OLTP database to the maximum, and allows decentralizing reports for operational management in Kiwibot.

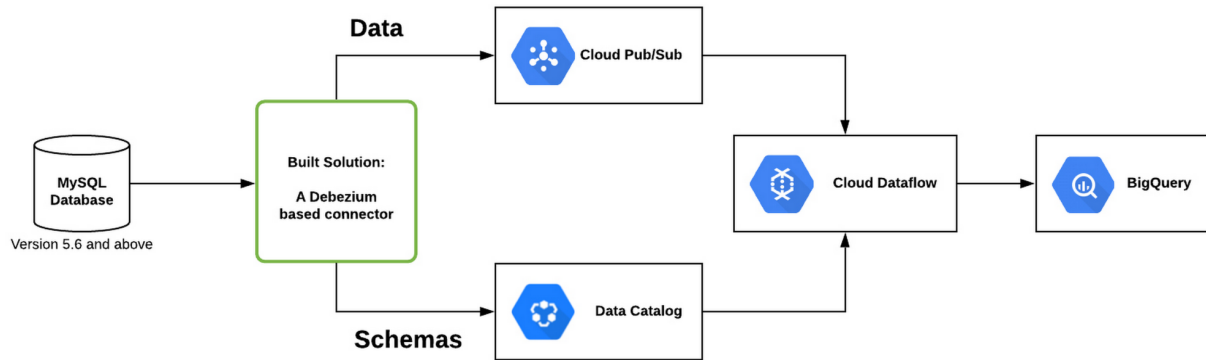


Diagram taken from Google Developers

<https://developers-latam.googleblog.com/2020/04/como-puedo-mover-datos-de-mysql-bigquery.html>

Although there are many tools and technology associated with the process of Extraction and Loading of data to isolated repositories that do not affect the operation, I recommend some that could provide us with better benefits in cost and scalability:

- * Fivetran (www.fivetran.com)
- * Stitchdata (www.stitchdata.com)
- * Airbyte (airbyte.io)

Assignment 3: — Business Understanding

Measurement indicators (KPI) allow companies to determine the degree of execution of their activities, and in that same measure to be able to make the pertinent decisions in favor of improving them: *"What is not measured, is not managed."*

Considering Kiwibot's main activity, the indicators with the greatest impact are those related to package deliveries. Some of the questions would be:

1. Are deliveries being made at the right times?

Metric that allows the operation to fine-tune the time it takes to deliver the package to the customer, and that same measure ensures the continuous use of our service.

2. Do customers receive the packages in full?

Completeness is an aspect that builds trust among customers, as it will show that Kiwibot is a serious organization that respects and exists only for the customer.

From the above metrics, we can obtain a precision indicator called OTIF (On Time In Full), which is given by the number of packages on time and complete that have been delivered to customers. Another no less important is the one given by On Time Delivery, calculated with the number of deliveries made on time, over the total number that occurred in a specific time range.

3. Is Kiwibot's level of service significant with respect to the area where it operates? This metric defines the orders that Kiwibot is able to fulfill in a given time.

Another extremely important and highly relevant aspect is related to Kiwibot devices. Determining its functionality and the possibility of detecting faults in the established routes in a timely manner will allow us to react promptly to adversity.

4. Can we react in real time to adverse situations presented en route by our devices?

It is crucial to have immediate measurements that alert us about this situation, and more at the time of the operation. A delivery of a package could be affected by external situations, and the possibility of taking action on the spot is vital for the organization. On the other hand, it is necessary to have traceability of these situations that can be measured and in some way mitigated so that they do not occur again as much as possible.

5. Do we know the level of satisfaction and the perception of our clients about the service provided?

Improving delivery services and times in order to increase the level of satisfaction of our customers, must be a consequence of an optimal operation. However,

knowing first-hand what they perceive about our service will allow us to improve secondary aspects that could generate other aspects that make a difference about our delivery compared to the traditional way.

Assignment 4: — Business Evaluation

One of the aspects with the greatest impact on the operation of any company is to allow it to understand and feel the situations that are being generated, regardless of whether they are good or bad. Obviously in the search for an optimal operation, they always try to analyze the bad ones in order to improve of course and the good ones, try to keep them with an excellent rating.

Now, it is not possible for a work team to notice bad practices in a certain process, if first, we do not obtain them in a controlled way, and secondly, we do not communicate them. Converting what is affecting us into measurable information will always be a challenge. The capture of events in each of the devices to automate the fault survey process, would allow us, for example, to take actions with the required immediacy. The reading of these events, their storage, processing and analysis gives us an unbeatable framework of possibilities in order to improve each of the technical details. that affect our devices.

With the above, I mean that each of the events generated in the entire Kiwibot ecosystem must be stored in repositories, to have the opportunity to debug and process, turning everything into information and then, with the indicated tools, into knowledge.

On the subject of storage we have mentioned some technologies such as MySQL and BigQuery, directly related to information. The first for OLTP (Transactional) storage and the second for Reporting and visualization of dashboards and KPI's in business intelligence. You can even implement algorithms that provide more details related to predicted and data science topics using this second repository.

However, it is worth clarifying that the data must be taken and uploaded to the respective repositories, to start a process of debugging and cleaning them in order to prepare them for the following: Visualization.

The visualization must be divided into two large segments: Reports and Business Intelligence.

Reports: Although normally many of them must be part of transactional applications, others, on the contrary, can be designed autonomously because they do not require immediacy in the operation. End-of-period reports, data that must be processed to obtain a new result, are some examples of those that we could implement outside of the operation. Some tools that can serve as a focus for this alternative are:

1. *Google DataStudio*. In it, we can design reports and reports, which are connected to the alternate repository and information refined in BigQuery.

2. *Looker*. Tool that, although it requires knowledge in its own language called LookML, is a good alternative to provide the organization with the relevant aspects of the business, taking advantage of the fact that it is an extension of Google Cloud.

Business Intelligence: Although by definition we say that it is the set of strategies and tools, with which we support our organization to make the best decisions, sometimes I think it falls short. There are countless processes and controls that must be implemented before releasing a piece of information. Now, the tools to be used are closely related to the level of data maturity that the organization has.

In most of the times, a tool does not solve problems that are had within an organization. Having a culture of information and how valuable it is in companies is much more relevant than the same tool that is implemented.

1. *Google DataStudio*. On the subject of BI (Business Intelligence), it is not one of the best, but it does allow organizations to provide that required level of maturity in the capture, processing and understanding of their data.

2. *Looker*. It offers organizations a different look at the use of dashboards and KPI's to make the best decisions. Although initially I mentioned it for the implementation of reports, it allows to extend a little more the details of the same in the construction of elements that contribute to the actors to make the best decisions.

3. *Qlik (QlikSense / Qlik View)*. In my opinion, one of the most robust tools in terms of Business Intelligence and data visualization. Although it requires full knowledge of its structured language in a fusion of Language C, Basic and Pascal, both to extract and transform the data, it offers visualization alternatives through its own environment using native objects for it, or implementing extensions created by others manufacturers or even using the technology of Mashups so that through own developments and HTML designs, embed your own analysis. A rich graphics tool full of technology.

