Plot	sub-plot	Tmnt	Sp	Ind	wt
1	A	X	Aus	1	10
1	A	C	Bus	1	20
1	B	X	Aus	3	10
1	B	C	Bus	4	10
2	A	X	Aus	1	20
2	A	C	Bus	4	10
2	B	X	Aus	5	20
2	B	C	Bus	4	10

Table 1: A dataset with more complex information

Mark and Huiping had a discussion about use cases to use "key yes", "distinct yes" and "identifying yes".
Give the data in Table 1. Assume we have the following observation types (Correct me if it's not very appropriate.)

- Plot (with measurement type: PlotLabel)
- SubPlot (with measurement type: SubPlotLabel)
- Tmnt (represent *Treatment* (With measurement type TmntType))
- Sp (represent *Species*, with measurement type: SpName)
- Ind (represent *Individual*, with measurement type: IndLabel and Weight)

Given the dataset in Table 1, users have different situations to catch.

- Case 1: *Plot* with label "1" should refer to the same one physical plot (i.e., the Plot in the first 4 row means the same thing); similarly, *plot* with label "2" should refer to the second physical plot (i.e., the Plot in the last 4 row means the same thing).
  - This can be captured in annotation by putting "Distinct yes" for observation type *Plot* and "key yes" for its measurement type *PlotLabel*.
- Case 2: *sub-plot*s with the same lable should refer to the same physical sub-plot if they are within the same plot; but the sub-plot with the same label with different *Plot* label are different sub-plots. E.g., Row 1 {Plot=1, sub-plot=A} refers the same sub-plot as that in Row 2, but is different from the one in Row 5 {Plot=2, sub-plot=A}.
  - This can be captured by putting "Distinct yes" for observation type *SubPlot* , "key yes" for its measurement type *SubPlotLabel*. We need to denote *Plot* is its context and with *identifying yes* specified on this context.
- Case 3: *Tmnt* with the same lable should refer to the same treatment process (So that we can aggregate on different treatment process, e.g., on "X" or on "C".) But the treatment at different sub-plot should refer to different treatment.
  - The first requirement can be captured by treating all the Treatment with value "X" as the same entity. The second requirement can be captured by treating the treatment in different sub-plots as different observations. I.e., treatments in row 1 and row 3 are of the same entity, but are different observations.

  - At the first glance, to represent this, *TmntType* for *Tmnt* should be specified with "key yes". It should have context *sub-plot* which is specified with "identifying yes".
  - After further analysis, **one question arises: The key measurements for the treatment observation is different from the key measurement of the entity treatment**. After considering the context, the key measurements for the treatment observation are {Plotlabel, SubPlotLabel, TmntType}. When two rows have the same value on these measurements, they represent the same observation instance. However, the key measurement for the treatment entity is just {TmntType}. When two rows have the same value on it, they represent the same entity instance. The *identifying* constraint can only capture the observation context. **This problem is more obvious when we analyze Case 5**.
  - Another different annotation may be applied to catch this semantic. E.g., treat the *treatments* in different rows as different entity instances. This way, the observation type and the entity type have the same key measurement types {Plotlabel, SubPlotLabel, TmntType}. However, this problem still exists for Case 4 and Case 5.

- Case 4: *Sp* with the same name should refer to the same species (e.g., a bird named *Aus* flies from sub-plot (1,A) to (1,B).) But the *Sp* with the same name at different sub-plot should refer to different observations of a specie.
  - At the first glance, to represent this, *SpName* for *Sp* should be specified with "key yes". It should have context *sub-plot* which is specified with "identifying yes".
  - **The same problem as Case 3: The key measurements for the *Sp* observation is different from the key measurement of the entity *Sp*.** the key measurements for the species observation are {Plotlabel, SubPlotLabel, SpName}. However, the key measurements for the species entity is just {SpName}.

- Case 5: *Ind* with the same label and and the same species name should refer to the same species. But the individual (with the same lable and the same species name) at different sub-plot should refer to different species observations.
  - **The same problem as Case 4 and Case 5: The key measurements for the *Ind* observation is different from the key measurement of the entity *Ind*.** the key measurements for the species observation are {Plotlabel, SubPlotLabel, SpName, Ind}. However, the key measurement for the species entity is just {SpName, Ind}. When two rows have the same value on these two columns, they represent the same entity instance. In this case, the observation context of Ind is *Sp* and *Sub-plot*. But the entity context of Ind is just *Sp*.

In summary, we can get a better idea about the problem described in the above use cases can when we answer the following two simple questions:
Q1: Will an entity type and an observation type (which is of the given entity type) always have the same key measurement type(s)? The above use cases give situations that the answer is no.
Q2: is *identifying* itself enough to distinguish the key measurement for observation types and for entity types? My temporary answer to this question is no.

A general thinking: the counterpart in RDB (Relational DataBase) is a relational scheme with key attributes. Here, we have two levels of objects: entity level and

instance level. Then, for different levels of objects, we need to have different ways to specify their key measurements.

Use cases to show that it's needed to have *key yes*, *identifying yes* and distinct yes

Q1: Why we need to distinguish the same entities using *key yes* and *identifying yes*)?

Assume the following table is the measurement for some plant tree at different spots.

Consider this question that a user may ask. Give me the average dbh for every *piru* tree (i.e., tree entity). First, we have three observations here. But how many tree entities here is a question.

There are several cases to consider:

- Case 1: The naive extreme way to interpret the data is that each observation is from different tree entity. Then, we have **four** tree entities. This may be too strict. People may say, well, I have some observations for the same entity.
- Case 2: The second naive extreme way is to interpret that different *spp* represent the different tree entity. That's obvious that *piru* is different from *abba*. With this constraint, we get **two** tree entities.
- Case 3: the assumption of case two has some obvious problem. People want to further limit that the same *spp* in the same *plt* should represent the same tree entity set. To achieve this, we use *identifying yes*. Now, it should return

$$(A, piru, 36), (B, piru, 33.2), (B, abba, 34)$$

plt	spp	dbh
A	piru	35.8
A	piru	36.2
B	piru	33.2
B	abba	34

(a)

plt	area	spp	dbh
A	1.0	piru	35.8
A	1.1	piru	36.2
B		piru	33.2
B		abba	34

(b)

Table 2: Dataset

Q2: Why we need to distinguish the same entities using *key yes* and *distinct yes*) to identify the same observation? For the example, we have one observation for splot A. What's the semantic purpose of this? What kind of query may need this? E.g., how many spots in this dataset?

Note 1: if one observation type is marked with *distinct yes*, all its measurements should be marked with *key yes*. Otherwise, we may have the same observation with different measurement values. E.g., what will happen for the following:

observation "o1" distinct yes
  entity "Plot"
  measurement "m1" key yes
    characteristic "EntityName"
    standard "Nominal"
  measurement "m2"
    characteristic "area"
    standard "sqft"

Will $(A, 1.0)$ and $(A, 1, 1)$ be treated as the same observation? According to the semantic meaning, they are the same observation because they have the same value on the key measurement and this observation type is marked witht *distinct yes*. However, there is something wrong here.

Based on this note, it seems like it is not useful to denote *distinct yes*. Basically, once all the measurements are marked with *key yes*, it automatically infers that it is distinct yes.