

The logo for MIOsoft, featuring the word "MIO" in a bold, white, sans-serif font, followed by "soft" in a smaller, lowercase, white, sans-serif font. A registered trademark symbol (®) is located to the upper right of the word "soft". The logo is centered within a solid blue square.


**MIO**soft®

# 3 CS OF DATA QUALITY

When you're figuring out how to measure your data quality, there's a lot of guidance out there.

A lot of it is framed in terms of dimensions of data quality. Dimensions are definitely a useful framing device for conceptualizing and aggregating data quality in important ways.

There's no set of data quality dimensions that is recognized as a universal standard. While this is OK, it can make it hard to get started.



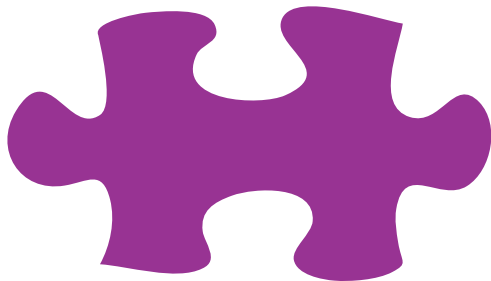
Here's the core data quality dimensions we suggest starting with. We've divided them into three related categories: completeness, correctness, and clarity.

To envision how all these fit together, imagine that your data is pieces of a puzzle.  
To get value out of your data, you need to assemble the puzzle (do data quality).



**COMPLETENESS** = having all the pieces to complete the puzzle shape.

**CORRECTNESS** = having all the pieces be from the same puzzle.



**CLARITY** = having the image on each puzzle piece be intact.

## COMPLETENESS

**KEY IDEA:** Your data is describing something—people or places or things or some combination of those.

Customer

Employee

Product

Material

Completeness is about how your data describes those objects:

### **EXISTENCE**

Does it contain the data you need?

Does it have the level of detail you need?

### **ACCESSIBILITY**

Can you get to it?

## CORRECTNESS

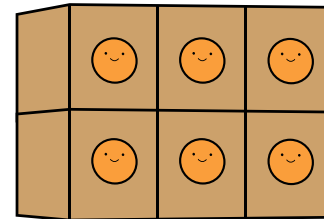
**KEY IDEA:** The something your data is describing is a real-life something.



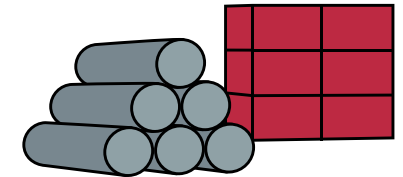
Customers



Employees



Products



Materials

Correctness is about your data's fidelity to the real-life objects it is describing:

### ACCURACY

Does the data about the entity reflect its real-world characteristics in a way that is suitable for your usage of the data?

### RECENCY

Is the data recent enough that you are confident in it?

# CLARITY

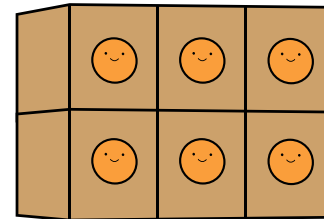
**KEY IDEA:** The something your data is describing has more than one aspect, and it has connections to other objects, too—it's not just floating in a void.



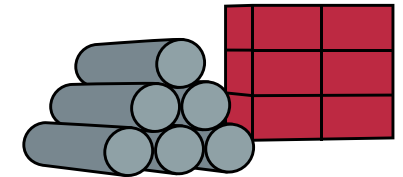
Customers



Employees



Products



Materials

Clarity is about understanding the different aspects of the entity and how they relate to each other, as well as how the entity relates to other entities:

## READABILITY

Is the data comprehensible to the people and systems that need it?

## STANDARDIZATION

Is all data that is similar comparable, no matter where it occurs?

## CONNECTIVITY

Do you have the data to recognize when there is data about a single entity in multiple places in a way that is relevant to you?

Do you have the data to recognize when there are relationships between entities that are relevant to you?

If you feel like you've gotten everything you were looking for by this point, great! We're glad we could provide you with some direction.

But if you (or stakeholders you have to answer to) want a more specific breakdown, read the next chapter for more information about how we define the different dimensions.



## DEEP DIVE

If you like detail, this is the chapter for you.

The broad strokes of data quality dimensions are good for surface engagement with data quality. But once it's time to actually start implementing it, everyone on your team needs to be on the same page about what each dimension means, exactly.

This is also where you come up with the concrete things you're going to measure, which is helpful when project stakeholders start asking things like: how do you plan to demonstrate results?

In this chapter, we'll explain how we define these different dimensions of data quality.

Each company is different, so you might find that you need to tweak some of our definitions to work for your purposes. Don't be afraid to do that if it means getting better results!



# COMPLETENESS

Completeness is about the presence of the data.



EXISTENCE

When data exists, it's somewhere in your systems. Pretty obvious.

And the data that doesn't exist? Maybe it was never collected, or maybe it was deleted. For data quality purposes, it doesn't matter.

Part of data quality will involve identifying existence gaps in your data, and whether they need to be filled in order to achieve the desired results for your project.

Whether you collect that data yourself, get it from a third party, or both, don't forget to conduct data quality on this gap-filling data. Recency of acquisition is, by itself, not a reliable indicator for high quality.

## ACCESSIBILITY

When data isn't accessible, you know that it exists, but you can't get to it.

There can be all kinds of reasons for accessibility barriers: regulations, policies, technology, resources, or anything else.

In some cases, the data quality solution you choose might itself be able to help: for instance, if you're having resource or technology barriers, or if you need to achieve a technical task like data masking in order to gain access.

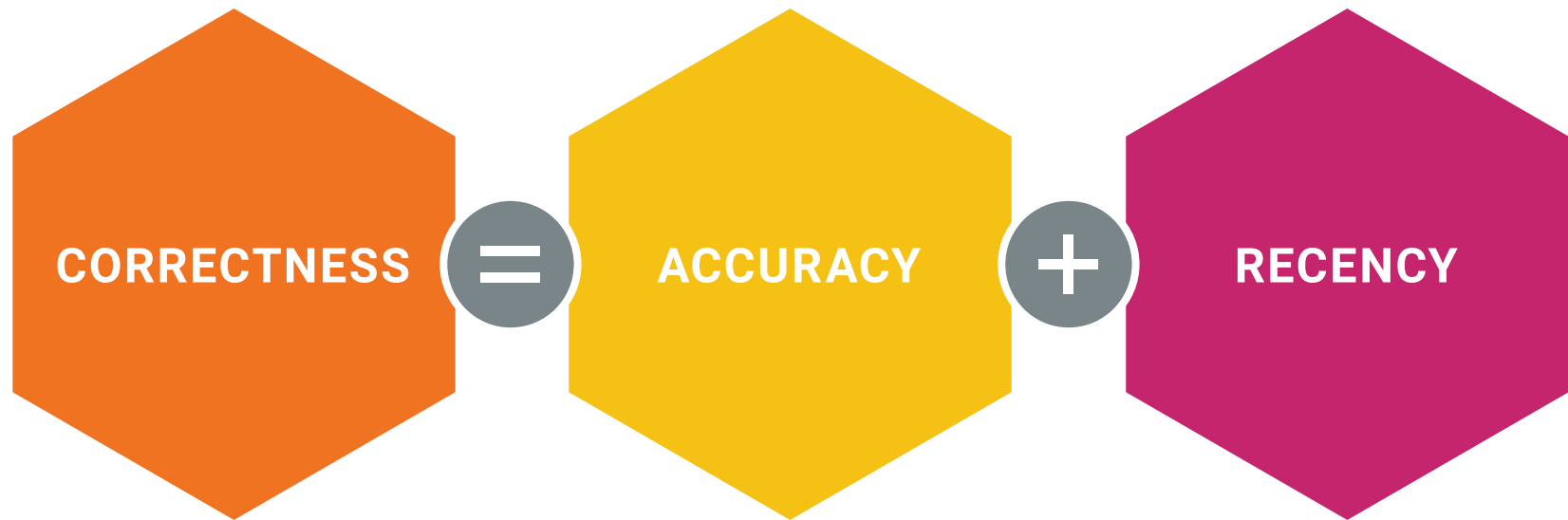
But if you're facing a non-technological barrier, like laws around where you can access from, or policies about who can access the data, you might simply have to count that inaccessible data as a negative in your overall data quality assessment.

When data is inaccessible, don't just forget about its existence. The output of your data quality project might make its way to people who do have access to that data.

And hopefully, those people can both compare your results against their knowledge of the data and provide you with useful feedback, and run some data quality themselves.

# CORRECTNESS

Correctness is about the factual appropriateness of the data.



RECENCY

Data can only be correct if it's recent enough that you have confidence in its accuracy.

Recency and accuracy are two sides of the same coin, but it's worth making the distinction between them.

For lots of data, accumulation is a type of change. So while a person only has one accurate birthdate, they can easily have multiple accurate shipping addresses: home, work, and school, for instance. For a shipping address, recency matters when you're trying to figure out whether a given address should be considered "good" or not.

And the degree of recency that a particular project requires in order to be accurate can vary significantly, even if they're using the same set of data.

We can divide the concept of "recency" into three rough categories, for the purposes of discussion: stable data, steady data, and volatile data. These categories are about how long data, once recorded accurately, is likely to remain accurate.

When we talk about these categories, we assume that once data is made accurate, it stays that way—inaccuracies aren't (re-) introduced. This isn't always the case, but its effect is pretty similar across recency categories, so we're not going to discuss it further here.

**Stable** data maintains its accuracy over time, full stop. If the data was accurate then, it's still accurate now. The GPS coordinates of a building or a person's date of birth are examples of data that is stable regarding recency.

**Steady** data maintains its accuracy over an extended period of time. Steady data that was accurate then is likely—but not guaranteed—to be accurate now, as long as it's been gathered in the last few months or years. Phone numbers, addresses, and spousal relationships are all examples of data that's steady regarding recency.

**Volatile** data maintains its accuracy for only a short period of time. You can't count on volatile data to stay accurate for very long—months at the most, and possibly only minutes or seconds. The time of last login, a credit card balance, or a cell phone location ping are all examples of data that's volatile regarding recency.

The degree of recency that is needed can vary between projects,

even when they use the same data.

This is because recency tends to become more important to data quality as the time sensitivity of a project increases and as its scope narrows. That's scope not just in terms of the number of entities involved in the project, but in terms of how the project's data is used.

For example, consider data that's always used in aggregation, for analytics, versus data that goes into front-line systems and guides interactions with high-value customers.

For the latter, up-to-the-second recency, even for volatile data, is important. For the former, depending on the type of analytics you're performing, you can potentially widen the scope of what is "recent enough"— and therefore "correct enough"— for the project.

This can introduce tricky factors into your data quality: the same data can meet the required recency needs for one project while simultaneously failing another's. It's important to recognize this and account for it in aggregated results.

ACCURACY

Correct data is factually accurate.

Ensuring accuracy is a huge part of data quality, because all the analytics in the world can't deliver benefits if the data they're given is just flat-out wrong.

Part of ensuring accuracy is simple validation, which is essentially checking whether the data's format and value even allow the possibility of the data being factually accurate.

The other, trickier part of accuracy is determining whether the data is a true reflection of the real world.

Manual verification against a third party or against the source is the most obvious way to do this, but it's also the most unrealistic in practice. It's extremely high-effort, resource-intensive, and slow.

Automated verification against a third party is one of the more popular strategies. It's much quicker than manual verification, and with vendors who specialize in providing verification as a service it's much easier to get started.



Automated verification does depend on the quality of the verifying database, but its biggest drawback is the limits of the data for which it is available.

If you want to use automated verification for person data, company data, or for data which has a centralized universal standard (think Dun & Bradstreet numbers), you have plenty of options.

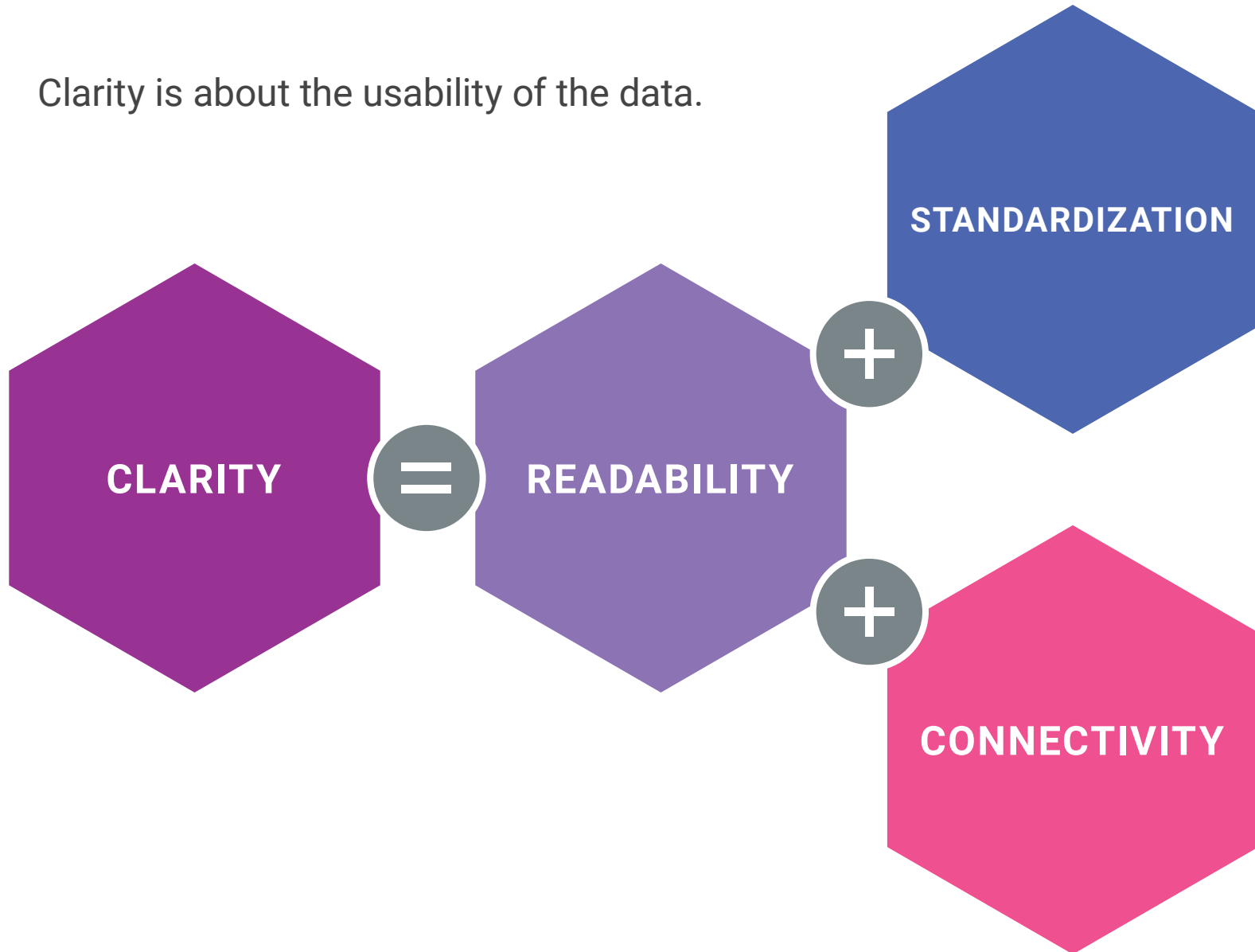
But if your data isn't something that verification service providers handle, or they are ruled out for some other reason, you have to find another option.

One of the more attainable, if you ever record data about the same thing more than once, is to check your data against itself. This can mean you have multiple systems, or you could have multiple records about something in a single system.

Cross-checking data against itself allows you to at least establish a degree of consistency in your data. Data that is contradicted by other records can be thrown out or verified by some other process.

# CLARITY

Clarity is about the usability of the data.



CLARITY

READABILITY

Readable data is comprehensible, specifically comprehensible to the intended readers. And a data reader could be a human, a machine, or both. Most likely, it's multiples of both.

Almost no data is only relevant to exactly one thing; most data will be put to use for different purposes, and it'll often have to be consumed differently for those purposes.

That means high-quality data has to be ready to be presented in multiple forms, which might not have very much in common with each other—or with the way the data is being stored. To build this variety of readable structures, high-quality data has to be very clearly defined and prepared.

CLARITY

## STANDARDIZATION

The basic concept of standardization is straightforward: when data is standardized, data of the same type is directly comparable, without having to do a lot of cleansing or transformations on whatever form the data is being stored in.

At its most fundamental level, you have to conceptualize type as, of course, data type: integer, string, boolean, date, etc.

To be standardized at this level, all dates must be comparable with other dates, strings with other strings, and so on.

You also need to consider business types: a string can be a name, but it can also be a street address, or a job title. A date can be a birthdate, or an expiration date, or a login date.

But you can't just stop there. You have to consider type at a higher level of abstraction, in terms of types of entities.

At this level, you need standardization like customers that are comparable with other customers, and products that are comparable with other products.

This doesn't mean that every entity of a type needs to have an absolutely identical structure, of course; a clothing product vs a food product, or an individual customer vs a business customer might (and probably will) require different models.

But it does mean that comparable entities should be structured in the same way to the extent possible. Personal names, for instance, should be fundamentally treated the same way for any entity that has a personal name. Individual customers should be comparable to individual employees, because both are people.

To achieve this, there needs to be a certain level of unity in your data quality initiative overall, so that different projects and different domains approach comparable entities in a standardized way.

This is where data governance—which often goes hand-in-hand with data quality—can play an important role. A consistent data dictionary can play a major role in standardization; so can associated meta-dictionary information like how standardization tools should be configured for different types of data.

Using a central business model to drive a data quality project, instead of attempting to work off the sources as-is, also

## CLARITY

contributes significantly to standardization. Different projects can use different combinations of sources, but if they're all referencing the same business view of the world, then it becomes much easier to standardize cross-project data.

CLARITY

CONNECTIVITY

Data connectivity is the understanding and representation of the meaningful relationships in your data.

Relationships between what? Between people, places, and things—the real-world entities that your data represents.

That includes both the relationships within entities, and the relationships between entities.

“Understanding the relationships within entities” is basically the idea of entity resolution: if you have multiple records about an entity, you should be able to recognize that fact, and understand the relationships between the data in the records well enough to synthesize those records into a single representation of the entity (or at least to generate a single-record representation of the multiple records).

Understanding relationships between entities is equally simple conceptually, but harder to execute in practice. Data quality should not only establish explicit relationships (where, for example, the name on a policy indicates a relationship between that policy and the person by that name), but implicit relationships.

Implicit relationships are relationships that the data suggests, but does not outright state: for example, if two people with the same last name live at the same address at the same time, they're likely to be family members, and they are definitely members of the same household.

These types of relationships can be detected even when each person's record doesn't mention the other, simply through the meaning of the data in the records. To achieve high quality in connectivity, these relationships need to be detected and created when they don't already exist.





# GOALS

To get value out of your data for a particular project, you need to apply data quality to data that's relevant and available.

We can't give you an exact outline of what relevant and available mean. That's going to vary from company to company and even project to project.

**Availability** is going to be affected by factors like:

- Company security policies concerning data access
- Regulatory rules concerning data access
- Operational needs of the systems
- Budget/effort/time available to create access

While **relevance** is often more affected by factors like:

- The subject of the project
- The scope of the project
- The confidence level required for the results



As you expand data quality within your company, your definitions of relevant and available are likely to change, as you discover data and connections (or a lack thereof) that you didn't know about before.



# CHECKLIST

Here's a summary guide of things to ask yourself about a data quality project as you get started. This list will help you start collecting information that you'll need to know to figure out your data quality measurables.

Remember that we always recommend a data-first perspective, and that might mean you need to revise your initial answers to these questions later. That's OK, and in fact is good: it shows that you're really looking at the data, not just seeing what you want to see.

Who uses the data that you are going to perform the data quality on?  
What are their projects?

Does all of the data that you want to evaluate exist?

What systems is the data located in?

Can you get data out of all of the above systems?

*This includes direct access to the system, but you could also read from a backup, from an Enterprise Service Bus, etc. to get the system's data without accessing the system itself.*

If not, is there any way to start getting data from those systems?

*This may involve applying to be granted security access to the system, or using a new tool (either your DQ tool or something else) to get access to the system.*

How recent does each of these types of data need to be for the projects that you are running data quality for?

What options do you have for determining the recency of a piece of data?

*If the record that a piece of data is in doesn't have an update timestamp, you may have to be creative. Does the database itself timestamp updates? Is there any recency information you can infer from the data?*

How much of the data you are working with is stable vs steady vs volatile?

What are your options for checking the factual accuracy of the data?

What data types do you expect to find?

What business types do you expect to find?

What entities do you expect to find?

How are these entities related to each other from a business perspective?

Which pieces of data and/or entities need to be directly comparable?

What form do the users of this data need it to be in?



## RELATED READING

[Proactive Data Quality: A Definition and Review of Application Points](#)

[Data Quality Vendor Evaluation Checklist](#)

[The Trailblazer Approach to Data Quality: An Introduction & Definition](#)

**MIO**soft®

## READY TO HELP

If you're ready to start seeing the 3 Cs in your data, be sure to check out MIOsoft. Our data quality software is designed to help you bring completeness, correctness, and clarity to your data by combining powerful data preparation and cutting edge entity resolution with a proactive, success-focused approach.

Are you working with a lot of data? No worries: MIOvantage is designed to scale, so no matter how much data you need to improve, MIOvantage has the capabilities you need to stay current.

## STANDING BY

Interested in working with MIOsoft for your data quality?

Get in touch!

[sales@miosoft.com](mailto:sales@miosoft.com)