Quantity: a physical property of an object, material, location or phenomenon which can be expressed numerically, measured or computed.

Function from *physical entities* to space of *quantity values*.

```
(Quantity length)
(domain length SpatialPath)
(subclass SpatialPath PhysicalEntity)
```

Note distinction between *quantity* and *quantity value*.

```
(= (length (across MyTable))(meter 3.25))
```

PhysicalEntity is a 'catch-all' class name for anything that can be said to possess a quantity. Some ontologies (OBO, DOLCE) require this to be separated into continuants vs. occurrents, with different treatments of time.

```
(= (temperature (atTime O t))(Kelvin 341))
(= (temperature O t)(Kelvin 341))
```

In CLIF these can be treated simply as alternative notations:

```
(forall ((q Quantity)(x PhysicalEntity)(t time))
     (= (q (atTime x t))(q x t) ))
```

(forall (fun x)(if (not ((domain fun) x)) (*DomainError* (fun x)))

(forall (x y)(iff (subclass x y)(forall (z)(if (x z)(y z)))))

(forall (x y)(if (and (Quantity x)(domain x y))(subclass y *PhysicalEntity*)))

(forall ((x Quantity) y)(if (domain x y)(subclass y *PhysicalEntity*)))

Scale: a structured set of measures used to record quantity values.

Scales come in several types. A given quantity may be measured using a variety of scales, but usually all of the same type. A given scale measures a unique quantity.

Note, there is an issue about whether a scale can have vector values or more complex values (eg quarternions). We restrict ourselves to scalar scales.

Scale is formalized as a function from the set of scale measures to quantity values, eg (meter 3.25), where the measures are numbers (the most common case.)

A Scale with a *total order* defined on it is an OrdinalScale. (The term NominalScale is used to refer to scales which are not Ordinal.) Eg Rockwell hardness.

An OrdinalScale which has an operator of multiplication by real numbers is an IntervalScale. Eg Celsius temperature. (Also called AffineScale.)

A RatioScale is an OrdinalScale which has an absolute zero and a *division* operator. A RatioScale with the non-negative reals as its measureSet is a NormalScale.

(Terminology from S. S. Stevens, see http://en.wikipedia.org/wiki/Level_of_measurement)

```
(NormalScale meter)
(= (measureSet meter) NonNegativeReal)
(measures meter length)

(forall ((s Scale))(exists ((q Quantity))(measures s q) ))

(OrdinalScale RockwellHard)
(= (measureSet RockwellHard) (RealInterval 0 200))

(forall ((s OrdinalScale)
  (forall ((x (measureSet s))(y (measureSet s)))
    (if (< x y)(< (s x)(s y))
)))

(forall ((s IntervalScale)
  (forall ((x (measureSet s))(y Real))
    ((measureSet s)(times y x))
))

(subclass OrdinalScale Scale)
(subclass IntervalScale OrdinalScale)
(subclass RatioScale IntervalScale)
(subclass NormalScale RatioScale)

(= OrdinalScale AffineScale)

(forall ((s RatioScale))(= (unit s)(s 1)) ))
(forall ((s RatioScale)
  (forall ((x (measureSet s))(< (s 0) x))
)
```