



# **Patterns - Dynamic Data Schema**

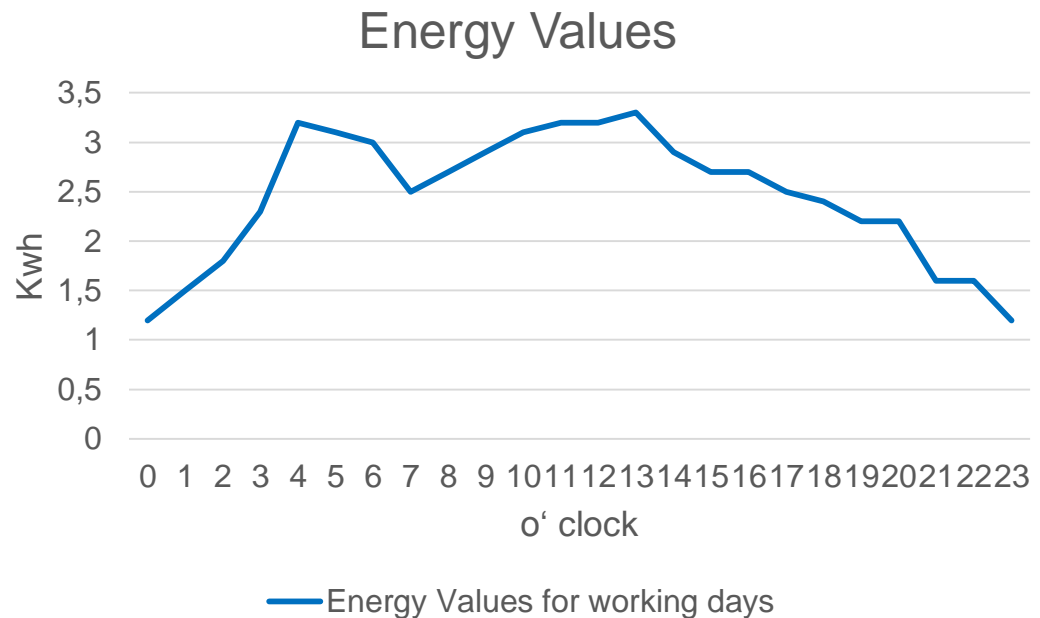
## **CityGML 3.0**

Kanishk Chaturvedi

Chair of Geoinformatics  
Technische Universität München  
[kanishk.chaturvedi@tum.de](mailto:kanishk.chaturvedi@tum.de)

# Need for supporting patterns

- ▶ As presented in 10th WP6 meeting, the Timeseries allows supporting absolute start and end points
  - Within which the attribute values can be mapped
  - Can be represented as tabulation of measured data
- ▶ For example, mapping of energy values of a building for every hour in a day

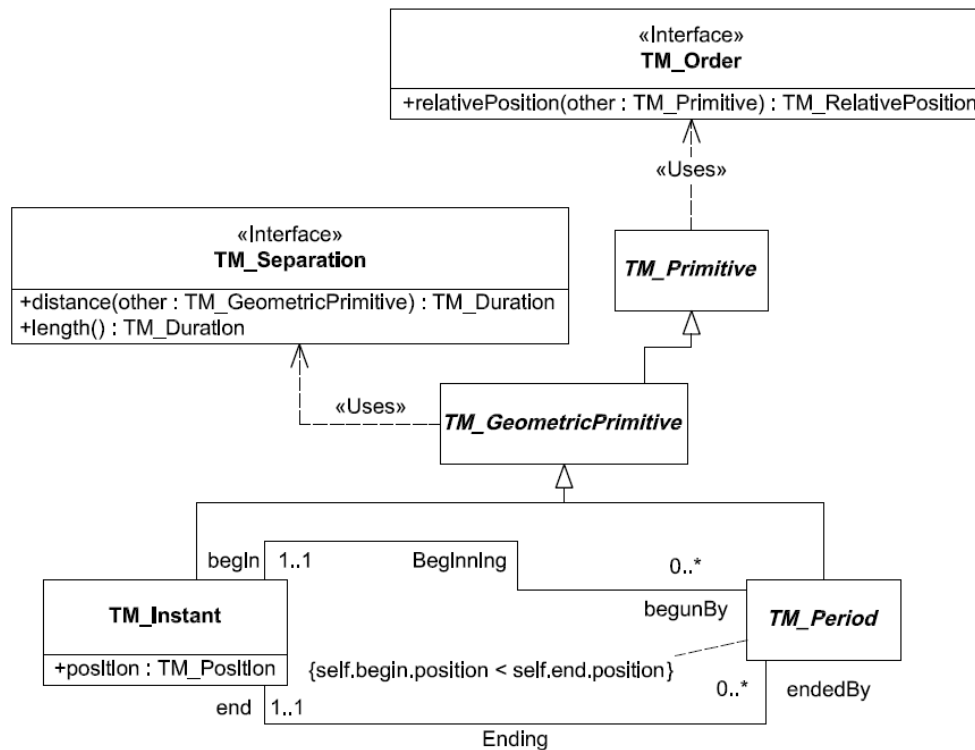


# Need for supporting patterns

- ▶ However, in many applications, it is not sufficient just to provide a means for the tabulation of time-value pairs.
  - They may require patterns to represent dynamic variations of properties based on statistics and general rules.
  - For example, during energy demand estimations, the energy values reflect specific patterns for individual weekdays and weekends.
- ▶ Such patterns should also be supported by the proposed dynamic data schema.

# Key Points (Absolute and relative times)

- ▶ ISO 19108 defines Absolute time points in two ways (TM\_Instant and TM\_Period)



Source : [ISO 19108 Temporal Schema]

# Key Points (Absolute and relative times)

## ► Possible representations of Absolute Time

- Time\_Instant (defines time position)
  - E.g., 2015-05-22T13:00:00 (Timestamp)
  - 2015-05-22 (yyyy-mm-dd)
  - 2015-05 (yyyy-mm)
  - 2015 (yyyy)
  - Monday?
- Time\_Period (having begin and end time positions)
  - 2015-05-01 to 2015-05-31
  - Possible to determine the length of the period or the temporal distance between begin and end points

# Key Points (Absolute and relative times)

## ► Relative Time Points

- TM\_Order in ISO 19108 is used to determine position of a time relative to another time position.
- These relative positions are based on the 13 temporal relationships identified by Allen[1]. Hence, it allows to perform comparative operations on time periods.
- However, metric or arithmetic operations can be very beneficial in defining the patterns in our schema.
  - E.g. 1-July-2015 + 1 Month = 1-August-2015
  - Or, 1-July-2015 07:00:00 + 1 Hour = 1-July-2015 08:00:00
- Such features are already available in Databases (such as Oracle), but not defined in ISO 19108 or GML. However, there is a mention about adding such features within the scope of Temporal DWG[2].
- ISO 19108 or GML can be extended to include such arithmetic operations.

Source[1] : ALLEN, J. F., Maintaining Knowledge about Temporal Intervals, Communications of the ACM, 1983, vol. 26 pp. 832-843

Source[2] : [http://external.opengeospatial.org/twiki\\_public/TemporalDWG/WebHome](http://external.opengeospatial.org/twiki_public/TemporalDWG/WebHome)

# Starting point of reference

- ▶ Repeating patterns within Outlook or Google Calendars
- ▶ Allows to schedule meeting repetitions as
  - Daily (Specific point of time every day)
  - Weekly (Specific point of time on specific day(s) every week)
  - Monthly
    - Specific day of the month
    - Specific day of the week
  - Yearly
- ▶ Repetition frequency
  - E.g. every 1 day or every 2 months etc.
- ▶ Termination of the repetitions
  - Define number of occurrences (after which the repetition would stop)
  - Define specific Date (after which the repetition would stop)

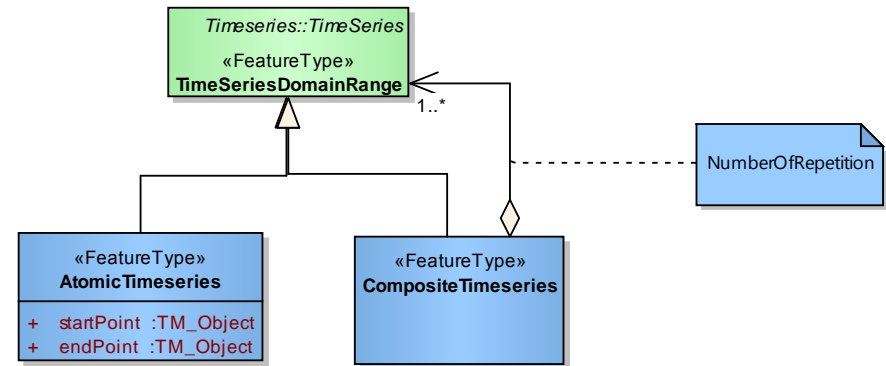
# Patterns

- ▶ Patterns can be defined by extending TimeSeriesDomain as
  - Atomic Timeseries, having
    - startPoint (Absolute Time Point or Period)
    - endPoint (Absolute Time Point or Period)
  - Composite Timeseries, having
    - Allows a TimeSeries to consist of multiple TimeSeries
    - Allows to define specific patterns
  
- ▶ A pattern can have multiple patterns
  - E.g., the daily energy values can be defined using a pattern.
  - Further, a pattern for a week can comprise of such daily patterns



# Patterns

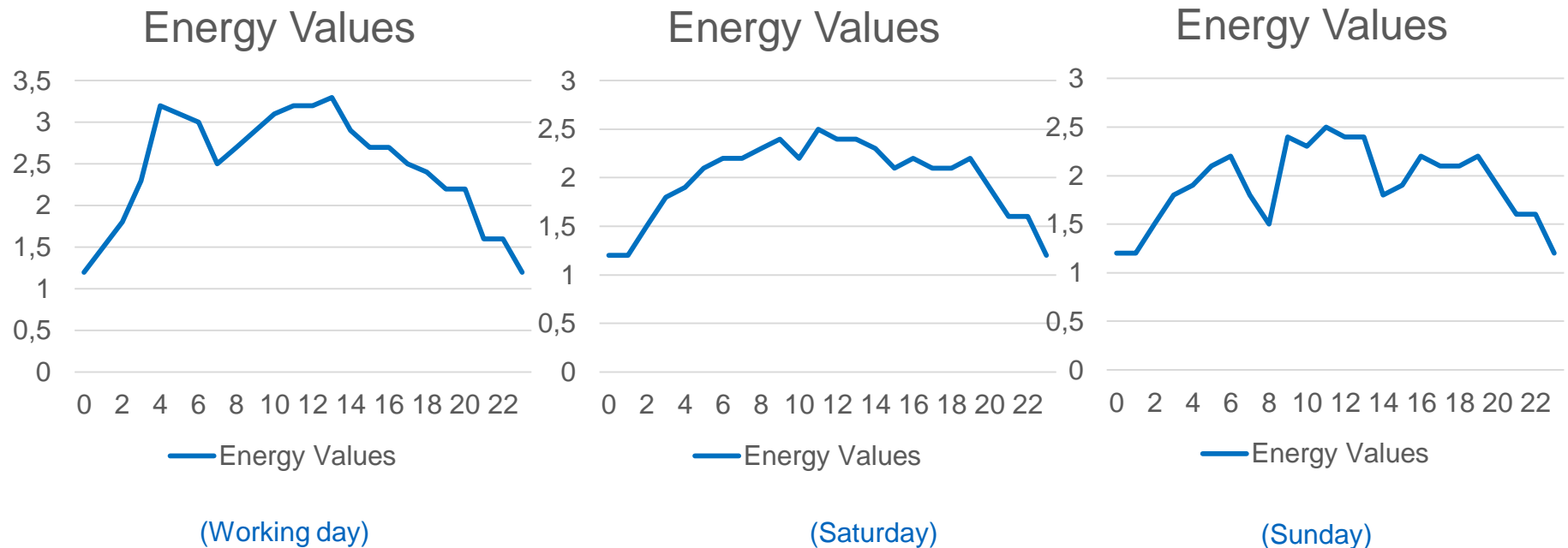
- ▶ Timeseries can be extended to support various kinds of patterns.
  - An atomic timeseries consists of a timeseries defined once.
  - The composite timeseries allows to define timeseries of arbitrary depths allowing nested patterns.
- ▶ Using the coverage approach, these patterns can be defined in the domain range and their values can be defined in the range set.



- ▶ Advantages:
  - Patterns can have sub-patterns of arbitrary depths
  - Using incrementor, any interval can be used for defining patterns, e.g.
    - Energy values for every 2 hours in a day, or
    - Comparison/pattern of energy demands for summers and winters over a period of 5 years

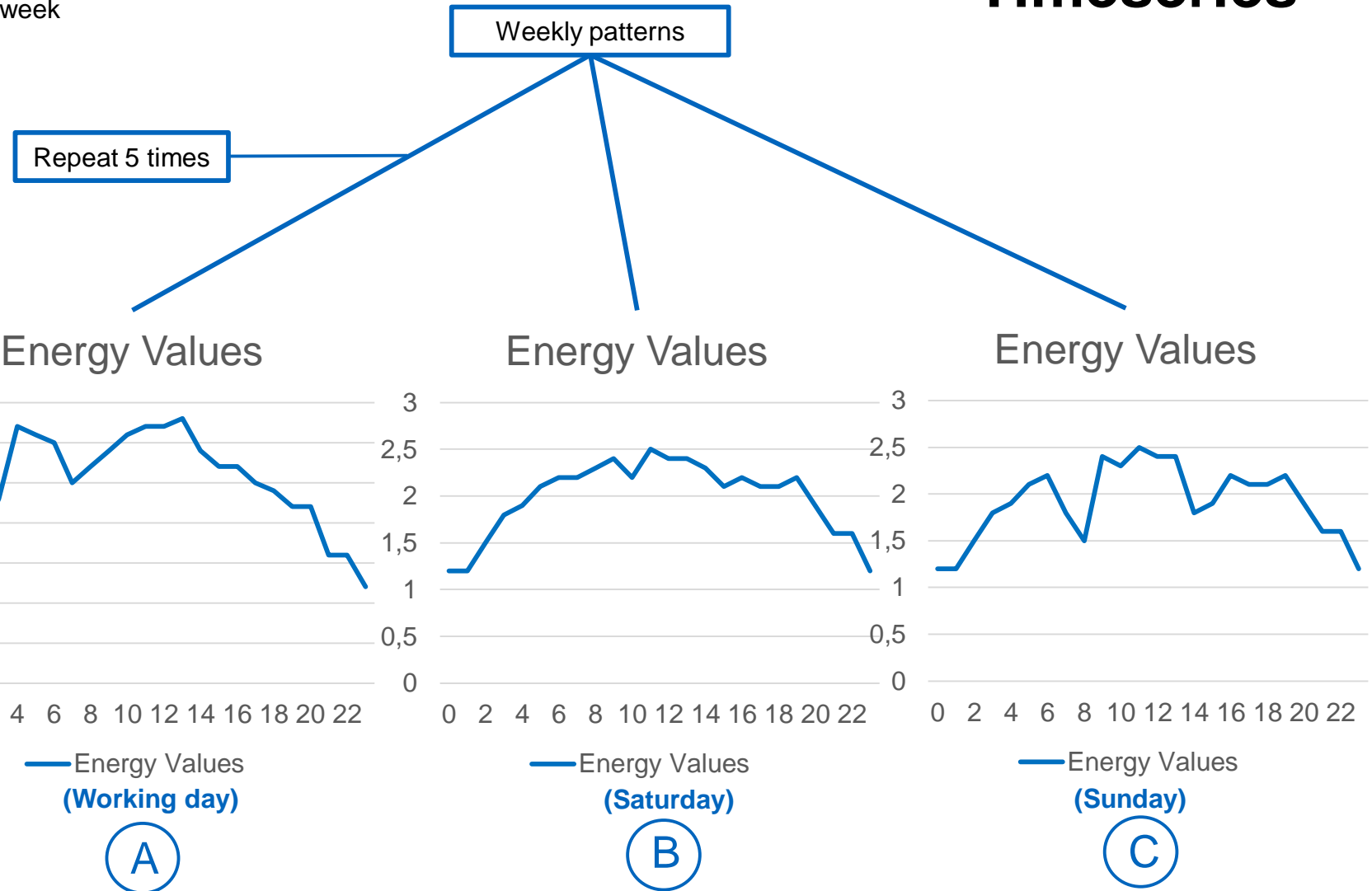
# Examples – Atomic Timeseries

- ▶ Atomic Timeseries can be defined once for specific time points/series.
- ▶ E.g, energy values for a weekday, a Saturday and a Sunday can be defined once as atomic timeseries.



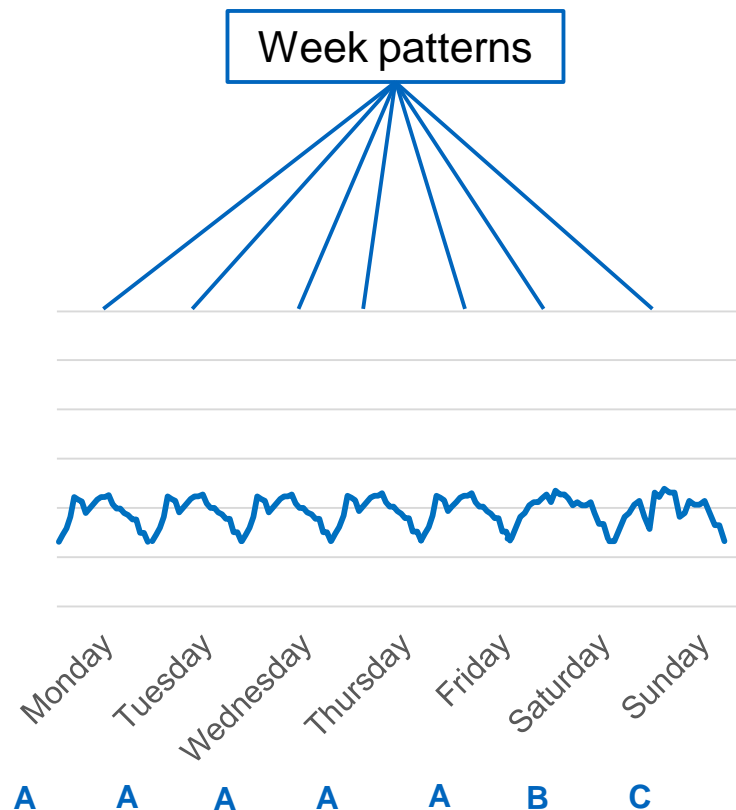
- ▶ However, composite timeseries allow repetitions of atomic timeseries for a number of times.
- ▶ E.g., the atomic timeseries of a weekday can have 5 repetitions to obtain the patterns for a week

# Composite Timeseries



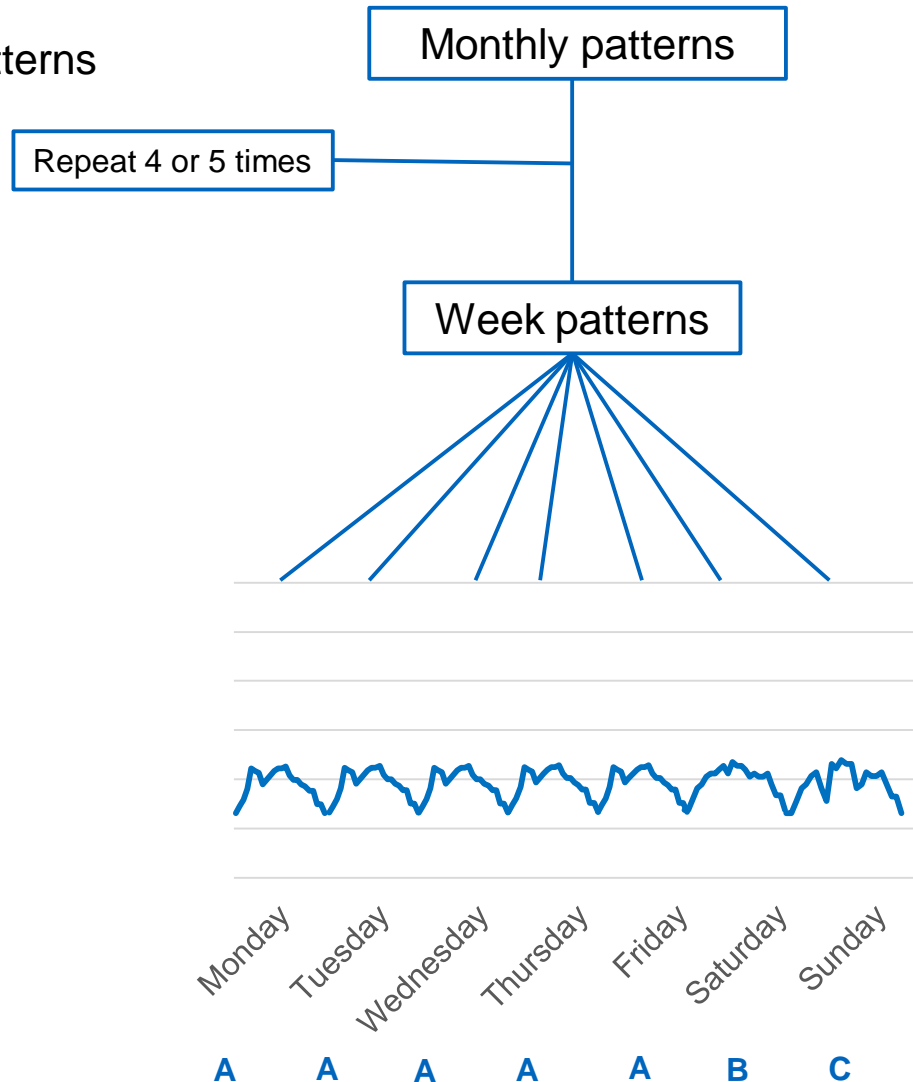
# Composite Timeseries (Complex patterns)

- ▶ Weekly patterns, consisting of
  - Five patterns for weekday
  - One pattern for weekend (Saturday and Sunday)



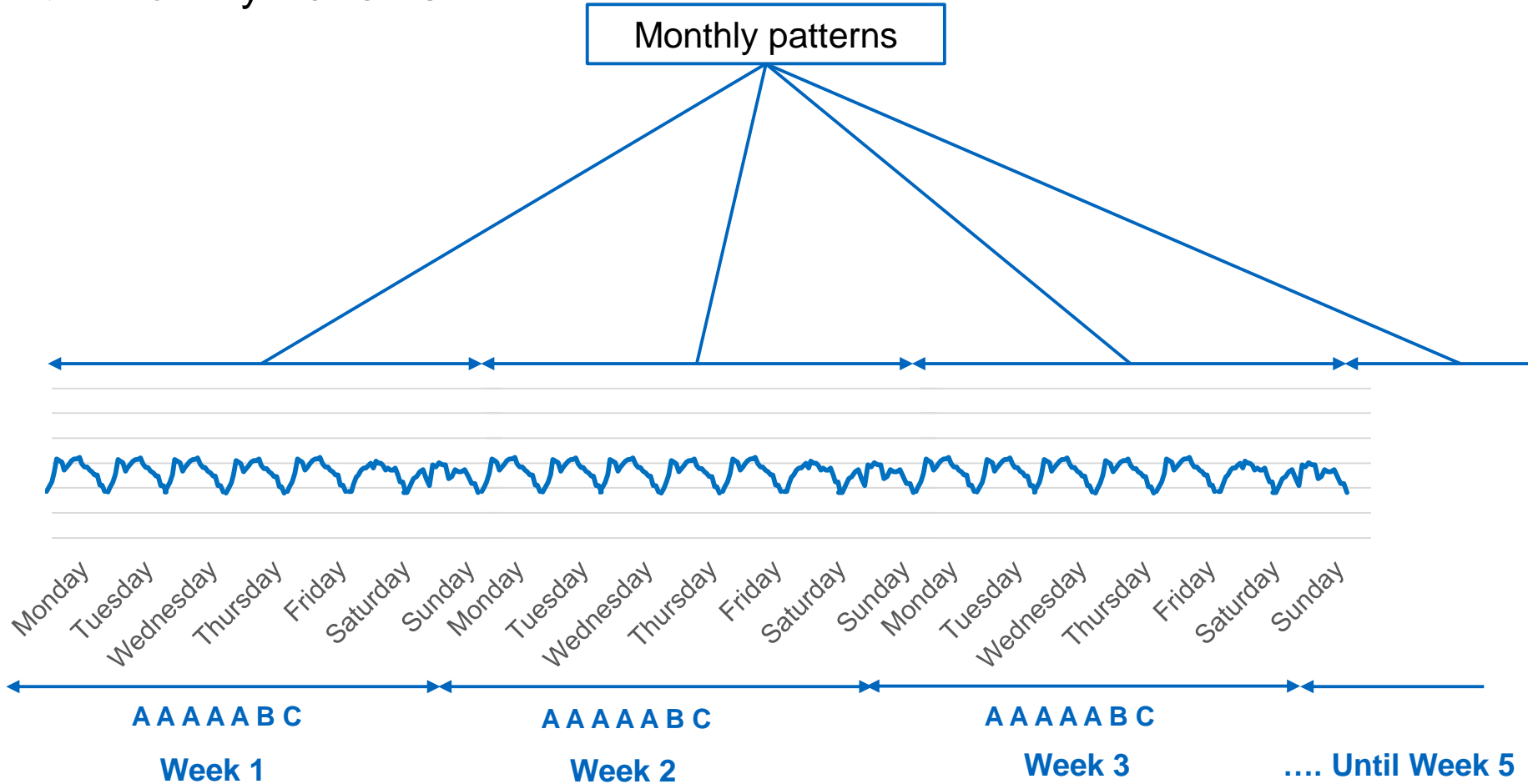
# Composite Timeseries (Complex patterns)

## ▶ Monthly Patterns



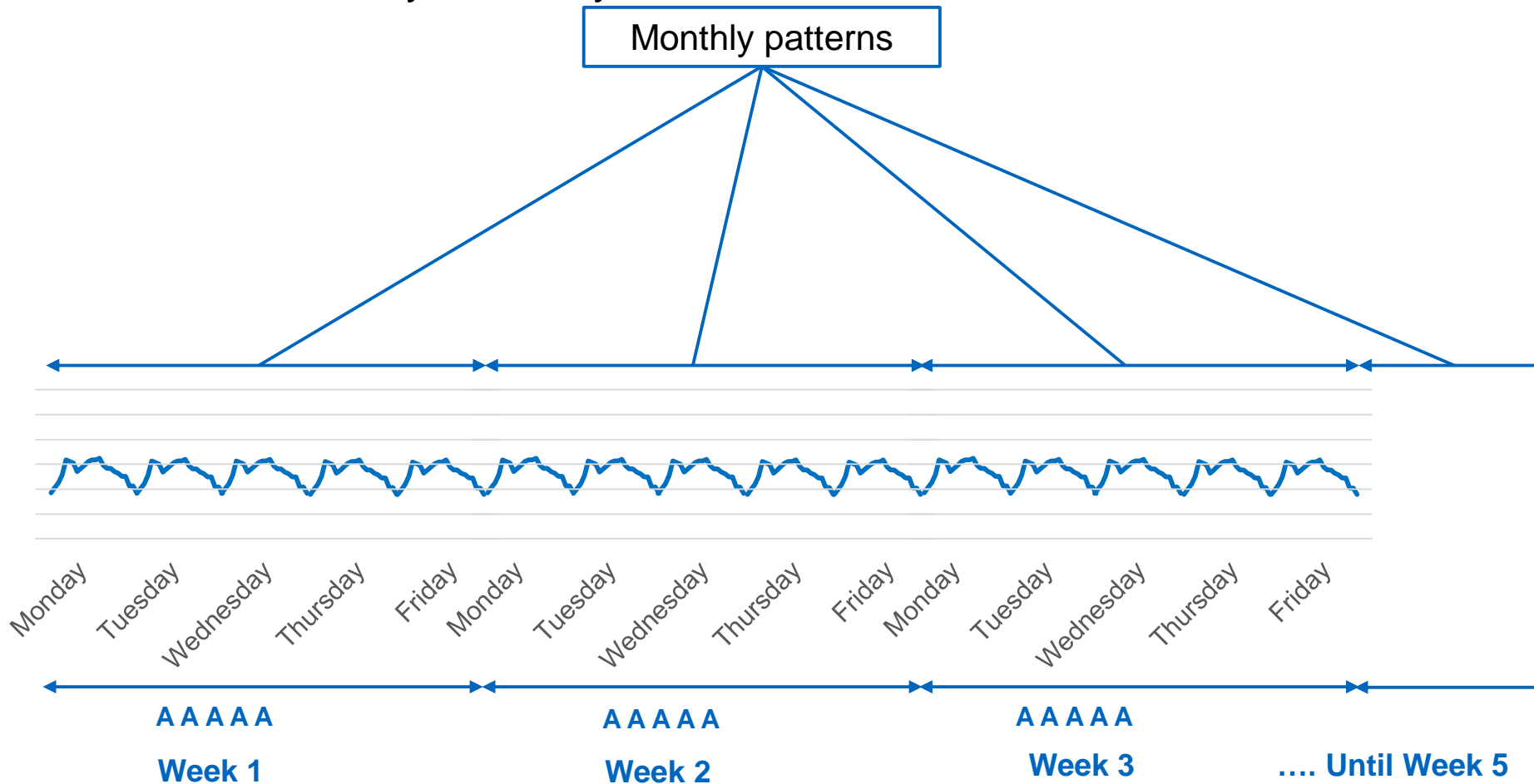
# Composite Timeseries (Complex patterns)

## ► Monthly Patterns



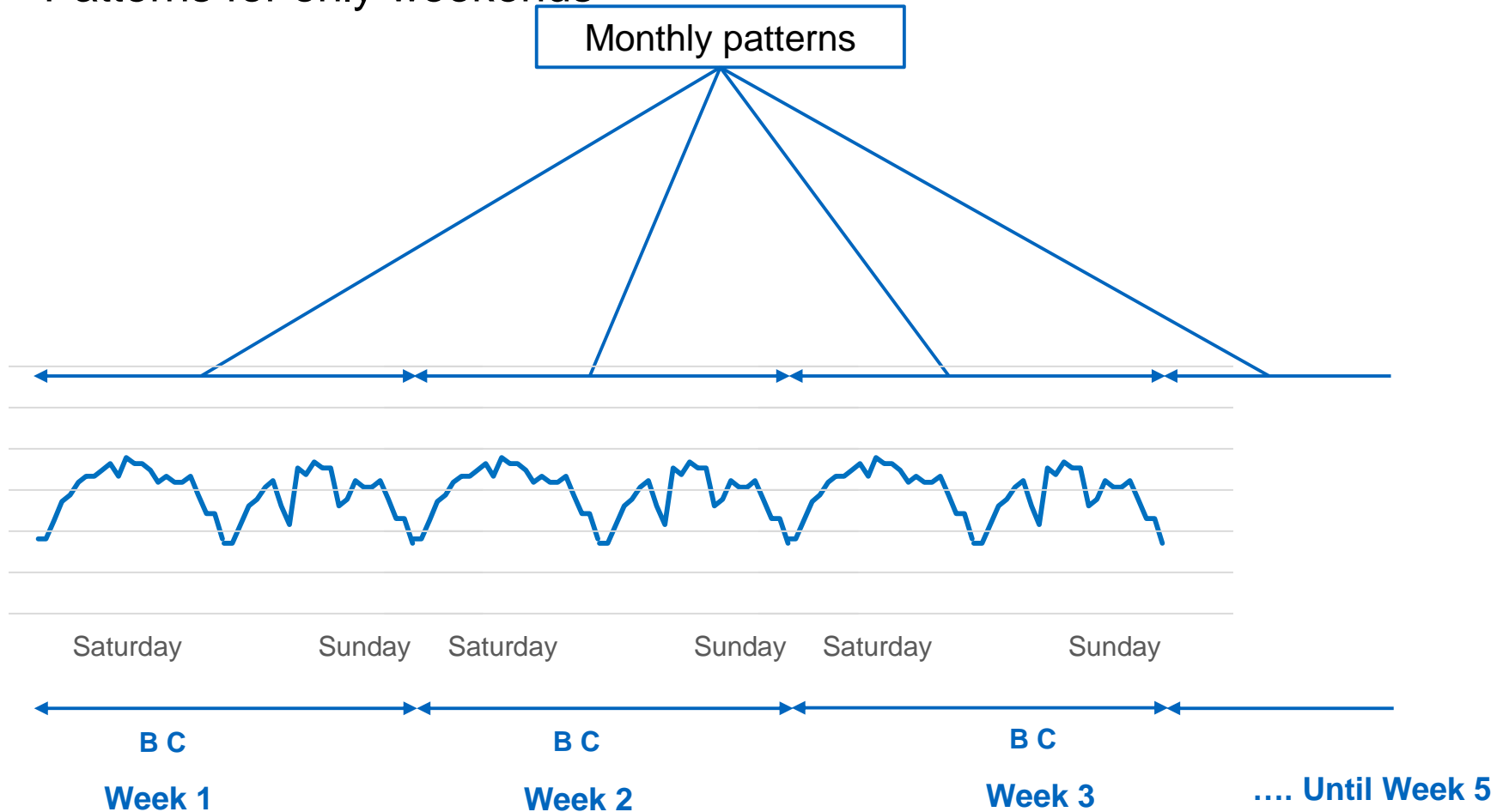
# Composite Timeseries (Complex patterns)

## ► Patterns for only weekdays



# Composite Timeseries (Complex patterns)

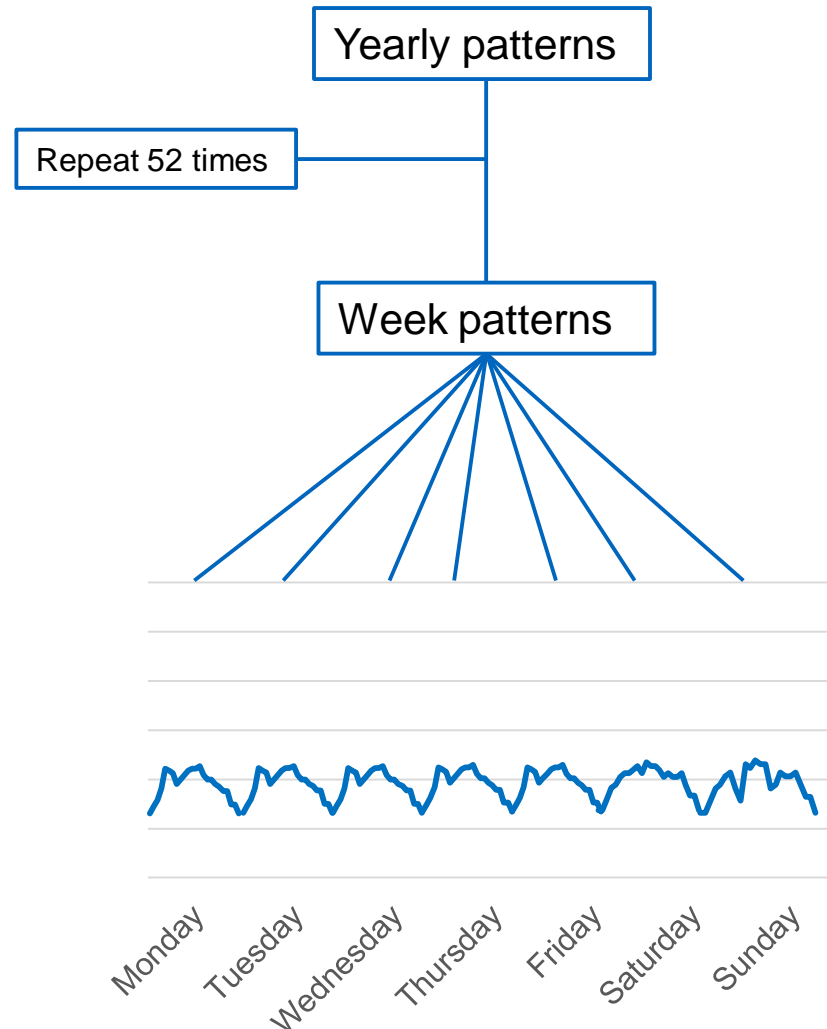
## ► Patterns for only weekends





# Composite Timeseries (Complex patterns)

## ► Yearly Patterns



# To Do

- ▶ Can external data sources like sensors can be referenced from the time-value-pair within patterns?