UNIVERSITÄT KOBLENZ · LANDAU

IWVI

BUSINESS PROCESS MANAGEMENT – EXERCISE

DIAGRAMED MODEL QUERY LANGUAGE (DMQL)

RECAP: GMQL

- Sketch out the underlying idea, before starting to model
- Stick to the correct types and number of parameters
  - Nest operators if necessary
- Make use of indents and variables, to avoid confusion
- Make sure you know what the output of a function looks like
  - Does it return a simple set or a set of sets?
  - What do the sets or subsets contain?
- Make use of the SELFUNION operator to turn a set of sets into a simple set

DMQL – BASIC IDEA

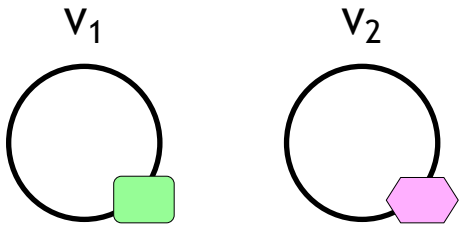
- A DMQL pattern is a graph, defined by a tuple

$$Q = (V_Q, E_Q, P_V, P_E, \delta, \varepsilon, G)$$

  - A pattern consists of vertices (V_Q) and edges (E_Q)
  - Vertex properties (P_V) and edge properties (P_E) define the vertices and edges
  - δ and ε are functions to assign the respective properties to the vertices and edges
  - Global rules (G) can be used to formulate constraints on patterns

DEFINITION OF VERTICES

- Set of vertices: $\{v_1,\ldots,v_n\}$
- Properties (for each vertex):
  - VID
  - VTYPES
  - VCAPTION



- Hint: If you define specific captions for vertices, your pattern will only find parts of the process with EXACTLY these captions

DEFINITION OF EDGES

- Set of edges: $\{e_1,\ldots,e_n\}$
- Properties (for each edge):
  - EID
  - ECAPTION
  - DIR
  - MINL, MAXL
  - MINVO, MAXVO
  - MINEO, MAXEO

  } minimal/maximal vertex and edge overlaps

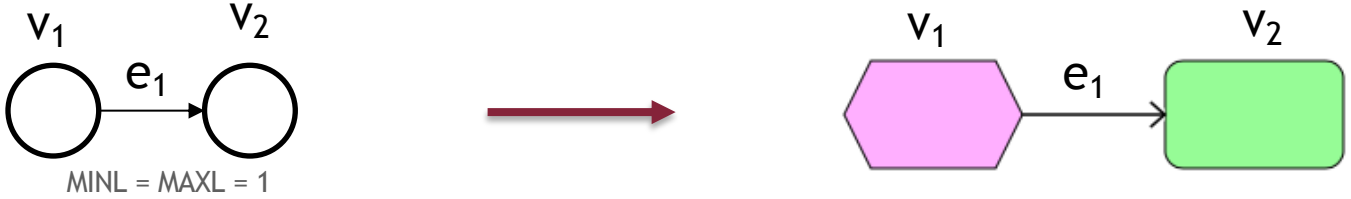
  - vtypesr, etypesr
  - vtypesf, etypesf

  } required/forbidden vertices and edges on path

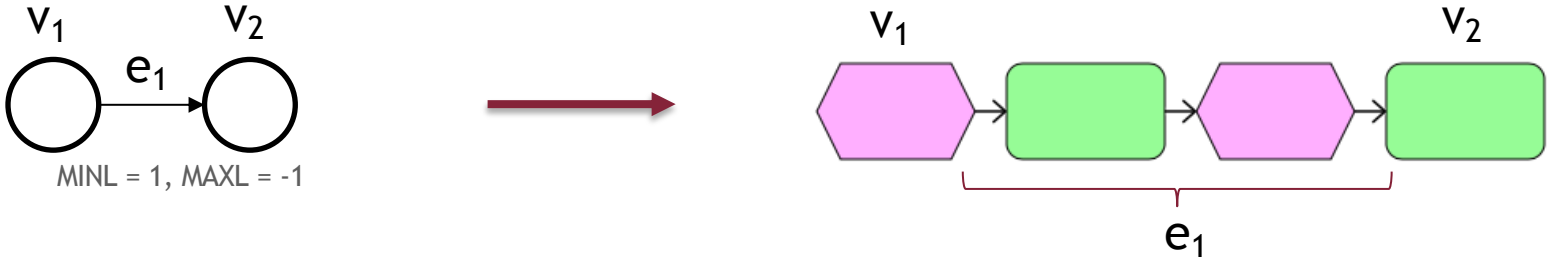
EDGES & PATHS

In DMQL, edges can be mapped to

- an edge of length one, i.e., an edge in DMQL is mapped to exactly one edge in the original model (isomorphism)



- a path, i.e., an edge in DMQL is mapped to a path of elements and edges in the original model (homeomorphism)

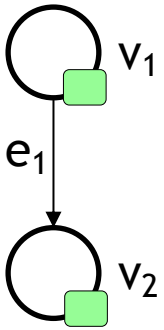


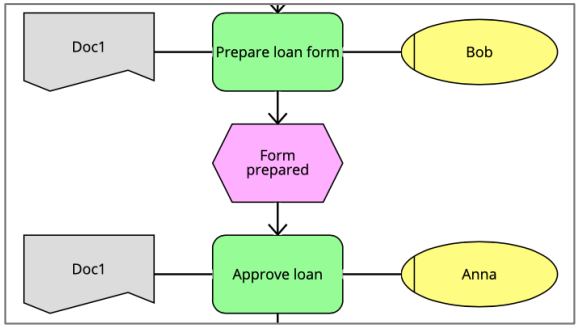
COMMON ERRORS: PATHS

- In DMQL you can make use of paths, if you want to define edges > 1
- MINL/MAXL define the length of these paths
- Try not to “hard code” these properties based on a given (exemplary) EPC, unless you specifically need to

Example:

If MAXL of e_1 = 2, then any pattern with a longer path would not be found



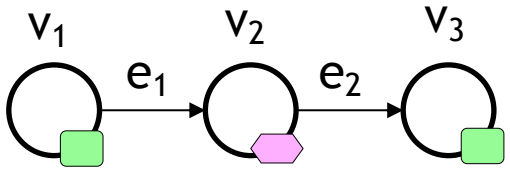


EXAMPLE

- Assume we want to design a pattern, that represents the following situation: A function is directly followed by an event and later on in the process there is another function (i.e., either directly following or somewhere in the future)

- e_1: MINL = 1; MAXL = 1

- e_2: MINL = 1, MAXL = -1



COMMON ERRORS: VTYPESR

The edge property *vtypesr* defines, which elements are required on a given path

- An edge of length 1 does not contain any elements
- Don’t include any elements in this set, unless you really require them on the path
  - Otherwise, your pattern might not find all results

GLOBAL RULES

- Global rules can be used to formulate constraints on patterns

→ relate properties of pattern elements to each other

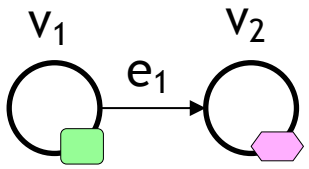
- Examples:
  - count outgoing/incoming/undirected edges of a vertex
  - compare captions

COMMON ERRORS – GENERAL

- All vertices and edges used in the DMQL pattern have to be defined (formally or in a diagram)
- Keep in mind, that a given EPC is often just an example of something your DMQL pattern should return
  - Don’t limit your query to this specific EPC, unless necessary
- Only include the minimum number of elements needed for your DMQL query
  - Otherwise, you might restrict your results too much

DMQL – EXAM HINTS

- Draw a diagram with the basic vertex and edge properties (ID, caption, type)



- You only have to define the properties needed for your specific pattern, everything that is not defined is assumed to be 0 or empty

UNIVERSITÄT KOBLENZ · LANDAU

IWVI

BUSINESS PROCESS MANAGEMENT – EXERCISE

INSTITUTE FOR IS RESEARCH

www.uni-koblenz.de