Towards a Data Model for Update Propagation in MR-DLM

Sheng Zhou[1], Nicolas Regnauld[1] and Carsten Roensdorf[2]
Ordnance Survey Research[1]
Ordnance Survey GeoData Management Group[2]

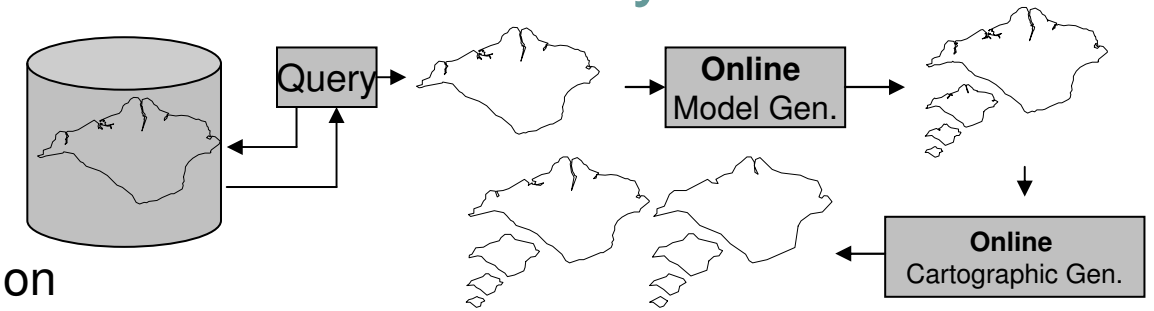
11th ICA Workshop on Generalisation and Multiple Representation, 20th to 21st June 2008, Montpellier, France

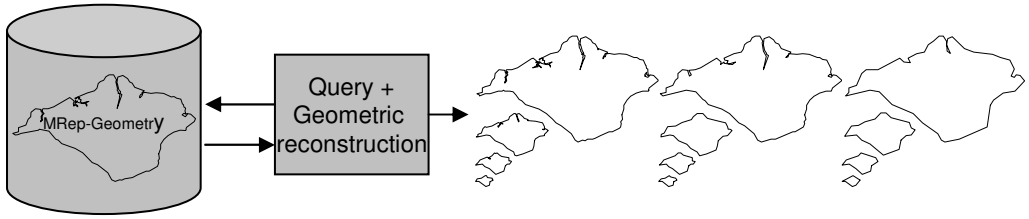
Multi-representation: the issue

- A conceptual issue
  - One geographic phenomenon, multiple viewpoints
  - A (geometric/graphic/etc.) representation for each viewpoint
- A SDB perspective
  - Different users require different representations
  - The more representations a SDB can supply, the more useful it becomes

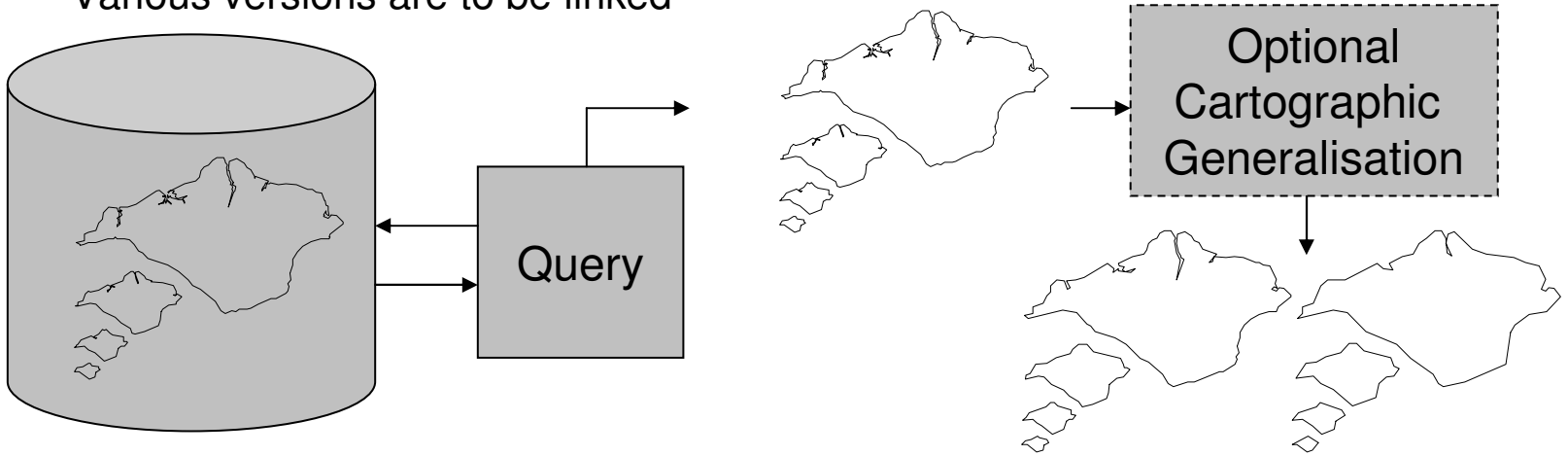
Multi-Representation: a reality check

- Two ideal (but currently less than practical) solutions:
  - Online Generalisation
    - Performance restriction
  - Pre-computed MRep-Geometry
    - Difficult to construct





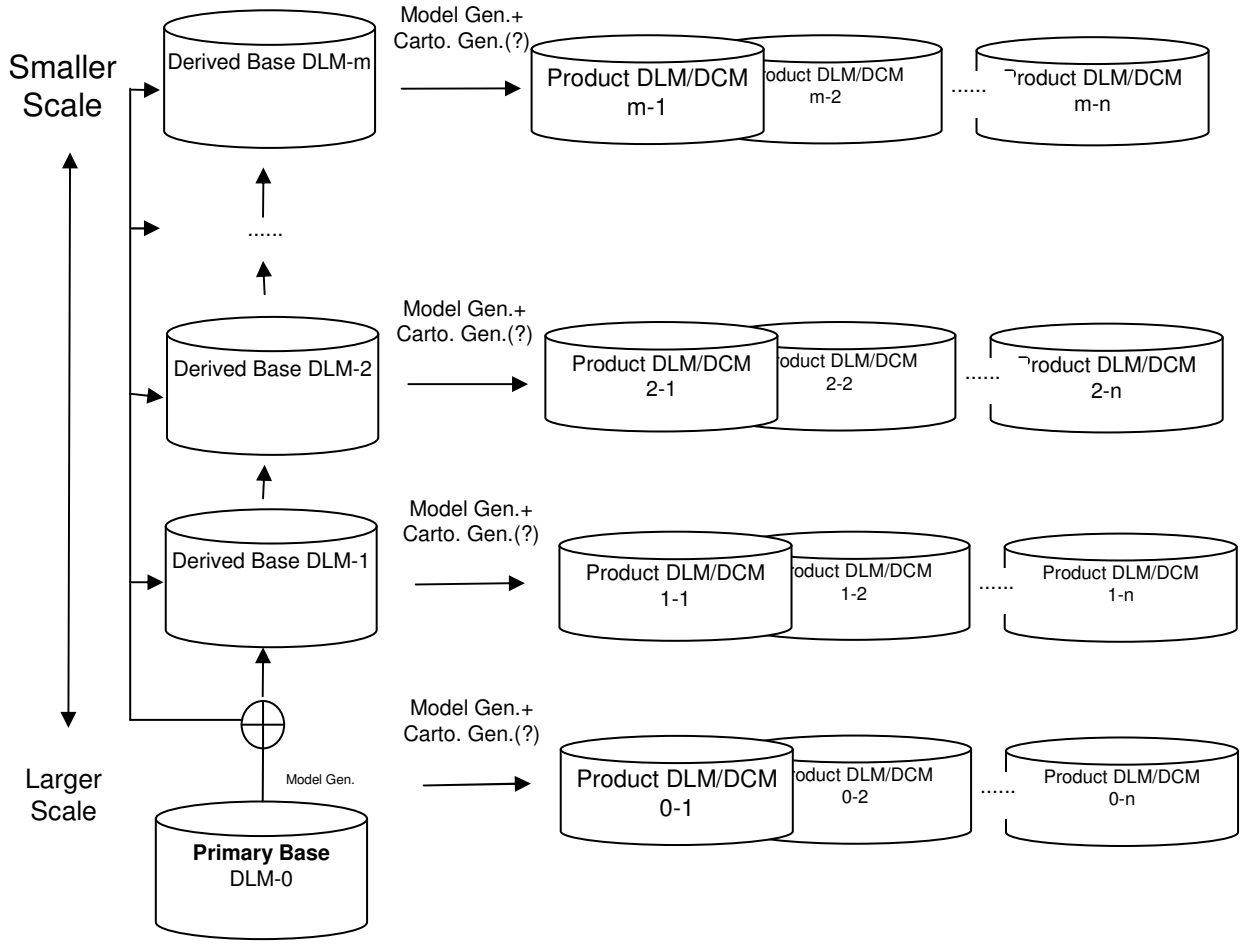
- A more practical solution: a Multi-version approach
  - A finite number of versions are generated for each feature
  - Various versions are to be linked



Generating Multiple Versions/Representations

- Match & Link
  - Matching and linking different representations of the same feature in different *existing* datasets
  - More difficult to handle derivation
- Link by generalisation
  - Generating additional (less detailed) representations from a base representation via generalisation
  - Linkage and derivation are inherent

An example of Multi-Version MR-DLM/MR-SDB

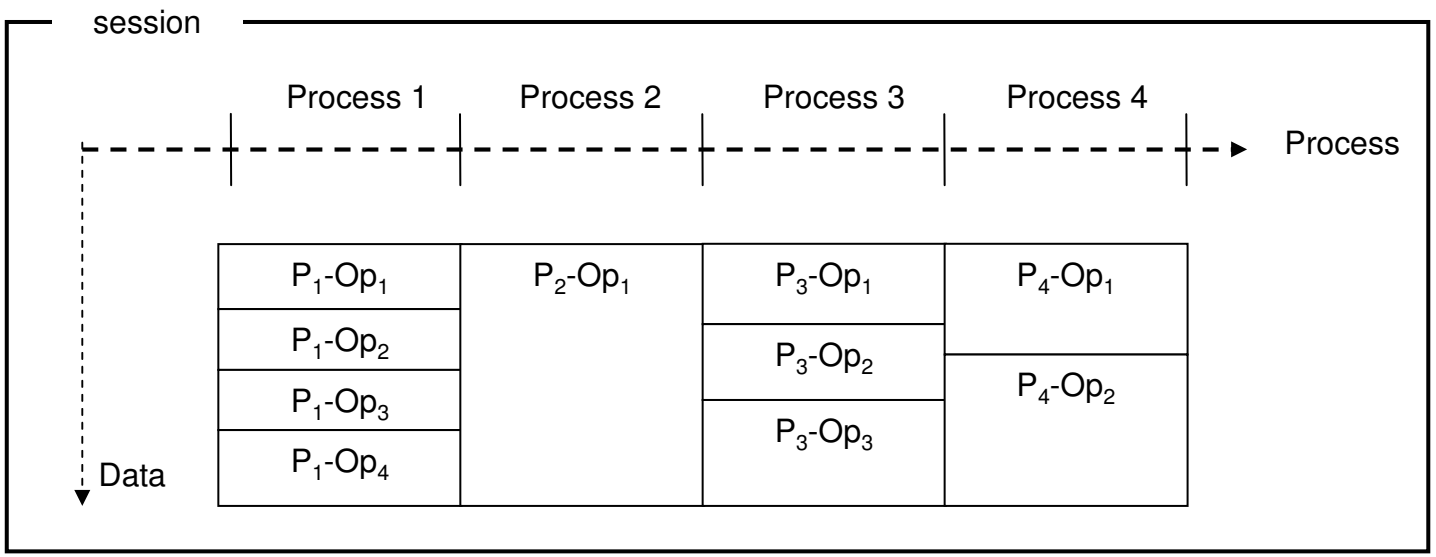


Maintaining Multiple Versions

- The nature of MR-DLM maintenance
  - Changes usually occur at one scale
  - Updates will be propagated to other scales
  - Local update preferable
- Information requirements of update propagation
  - Minimum: linkage between source and generalised/derived
  - Desirable: application-specific operational information (algorithm types, parameters, etc.)

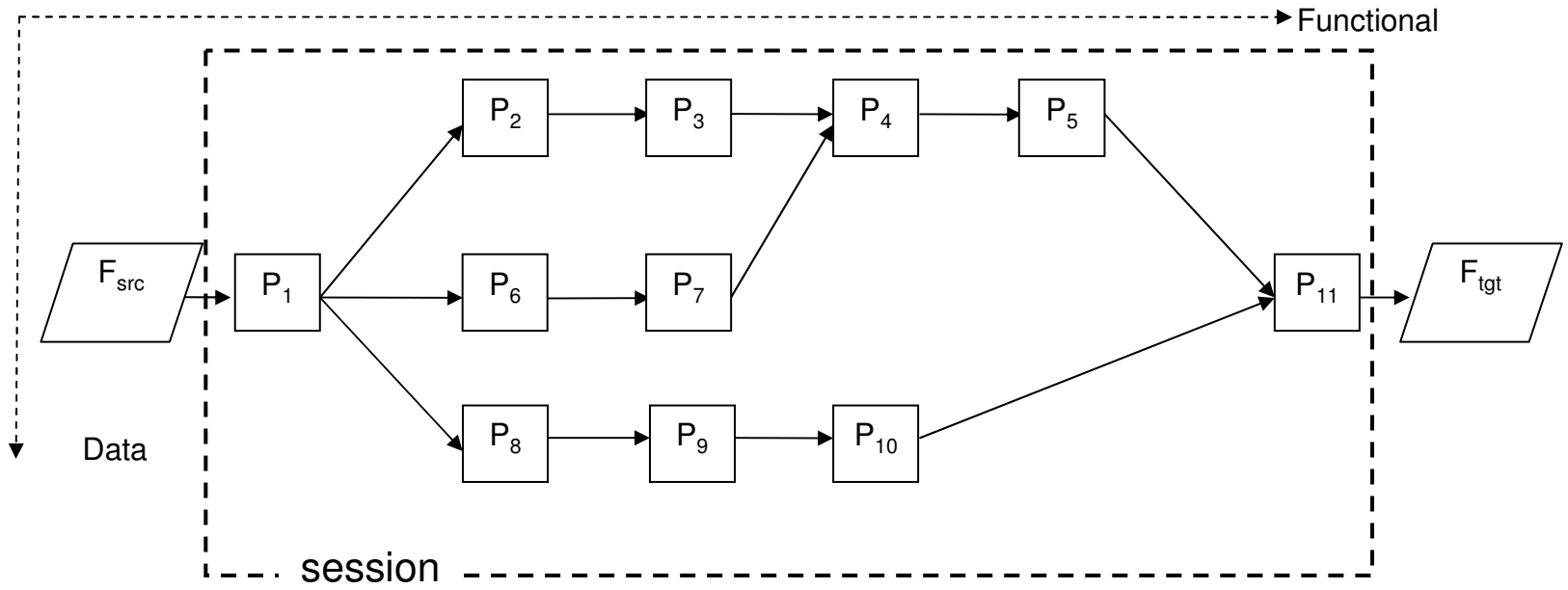
A Model of Generalisation Process

- A hierarchy of Session/Process/Operation
  - A session consists of a set of prioritised processes
  - A process consists of one or more parallel operations
  - Operations are the basic functional units of generalisation



Session as a DAG of Processes

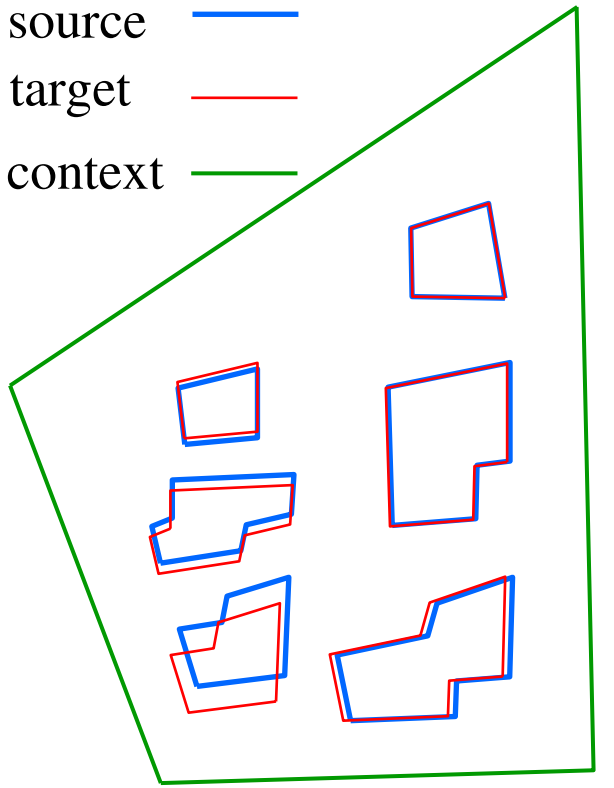
- "Precedence" represents the execution priority of processes, e.g.:
  - P_2 precedes P_3



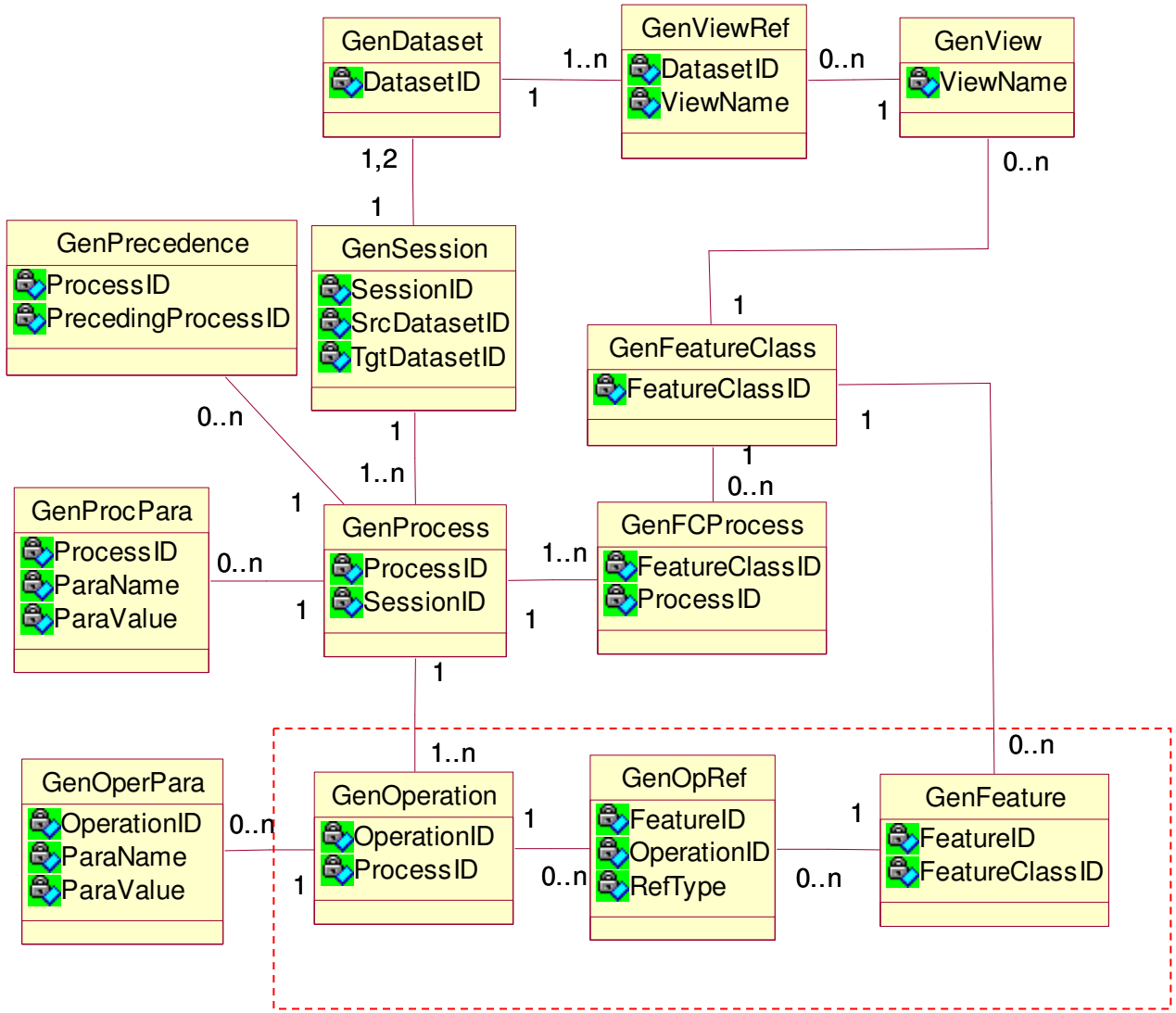
Processes, Operations and features

- Division of a process into operations:
  - Current design: parallel and independent, i.e.
    - Partition on data dimension
    - Functionally identical to parent process
  - Potential alternative:
    - Priority/precedence for operations:
      - partition on both data and functional dimensions
- Role of features in an operation
  - Source: the original
  - Target: the generalised
  - Context: participating but not manipulated

Example: Building displacement

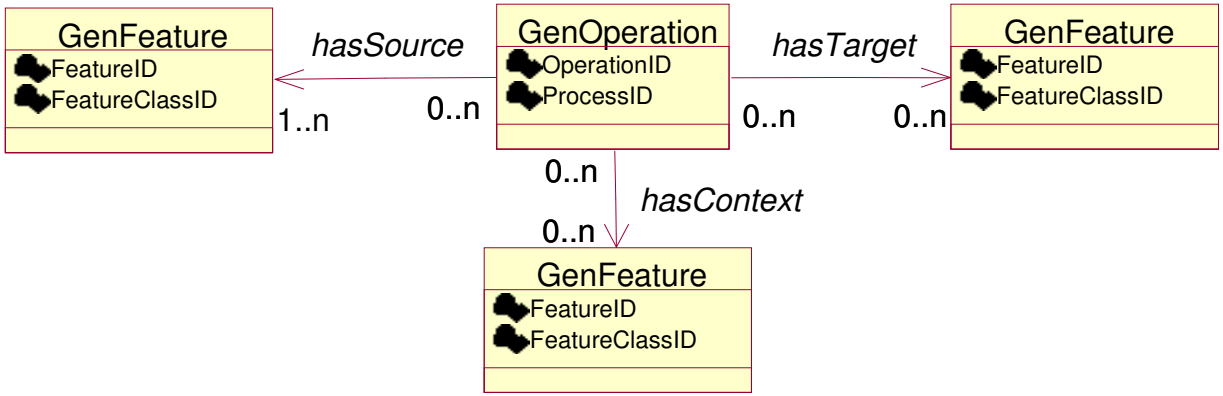


Generalisation Logging – the data model

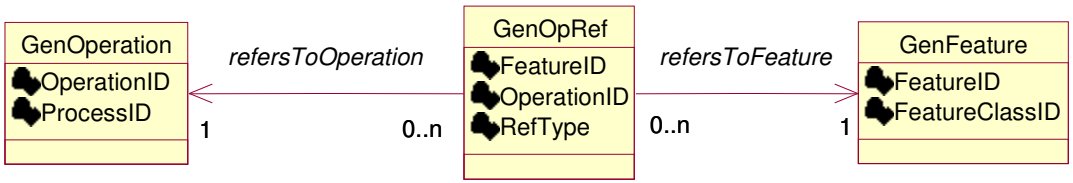


Linking Features and Operations

- Relations between features and operations



- A relational realisation

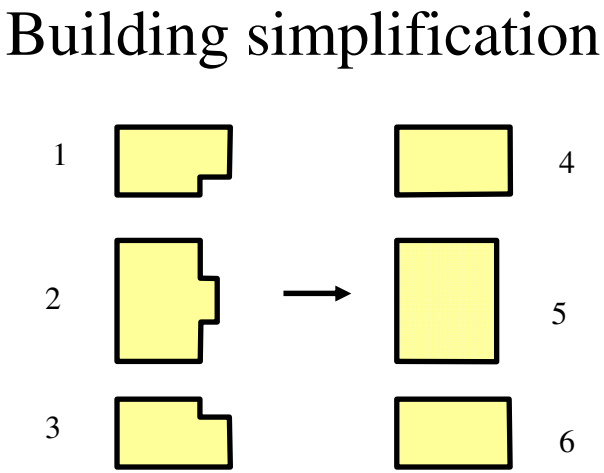


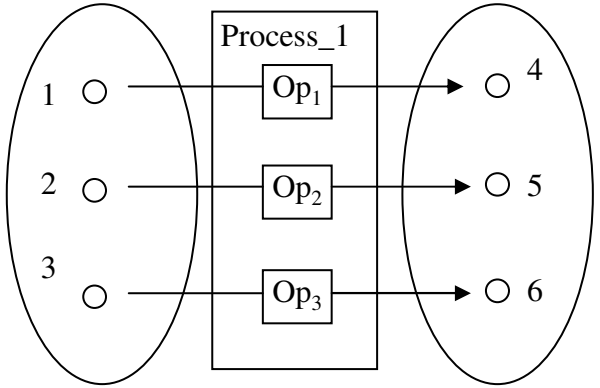
- Op. Ref. types: Source, Target, Context, Failed, etc.

What we can do with this model?

- Tracing the full generalisation history of each feature
  - Generalisation operations applied to the feature
  - Parameter values used
- Tracing the full generalisation history of each feature class (good source for automatic learning)
  - Useful when a new feature is added
- Re-constructing execution sequences of processes in a session from precedence information (by “topological sorting”)
- Storing and retrieving source, target or contextual features of a generalisation operation/process/session

Example A: 1:1 Source-Target Mapping



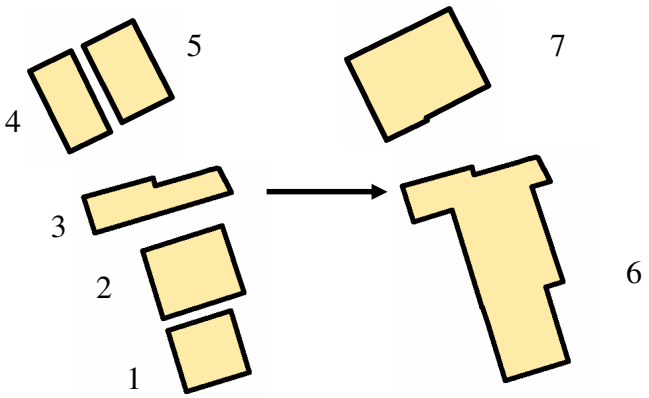


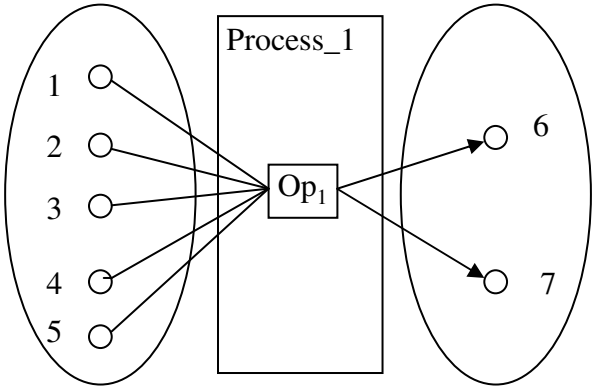
Source table		Target table
FeatureID		FeatureID
1		4
2		5
3		6

ProcessID	OperationID	FeatureID	ReferenceType
1	1	1	S
1	1	4	T
1	2	2	S
1	2	5	T
1	3	3	S
1	3	6	T

Example B: M:N Source-Target Mapping

Building aggregation





Source table		Target table
FeatureID		FeatureID
1		6
2		7
3		
4		
5		

ProcessID	OperationID	FeatureID	ReferenceType
1	1	1	S
1	1	2	S
1	1	3	S
1	1	4	S
1	1	5	S
1	1	6	T
1	1	7	T

What's next

- Theoretical:
  - Update management
    - Versioning to handle functional changes
      - E.g. new/alternative/improved algorithms
    - Versioning to handle data changes
      - E.g. addition/deletion/modification of features
  - Formalisation of the model (?)
  - A formal model for multi-version based multi-representation
- Practical:
  - API for handling generalisation logs
  - Experiment (MRep transportation network)

Question Time