

A top-level categorization of types of granularity

C. Maria Keet

KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
tel: +39 0471 016127 – fax: +39 0471 016009 – keet@inf.unibz.it

Abstract. Multiple different understandings and uses exist of what granularity is and how to implement it, where the former influences success of the latter with regards to storing granular data and using granularity for automated reasoning over the data or information, such as granular querying for information retrieval. We propose a taxonomy of types of granularity and discuss for each leaf type how the entities or instances relate within its granular level and between levels. Such distinctions give guidelines to a modeler to better distinguish between the types of granularity in the design phase and the software developer to improve on implementations of granularity. Moreover, these foundational semantics of granularity provide a basis from which to develop a comprehensive theory of granularity.

INTRODUCTION

Granularity deals with articulating something (hierarchically) according to certain criteria, the granular perspective, where a lower level within a perspective contains information or knowledge (*i.e.*, entities, concepts, relations, constraints) or data (measurements, laboratory experiments etc.) that is more detailed than contents in the adjacent higher level. Conversely, a higher level ‘abstracts away’—simplifies or makes indistinguishable—finer-grained details. A granular level, or level of granularity, contains one or more entity types and/or instances. What granularity comprises can differ between research disciplines that tend to emphasize one aspect or the other. It combines efforts from philosophy, AI, machine learning, database theory and data mining, (applied) mathematics with fuzzy sets, fuzzy logic, and rough sets Yao (2005, 2007), for example (Peters *et al.*, 2002; Yao, 2004; Zadeh, 1997; Zhang *et al.*, 2002). Several usages of granularity capture subtle, but essential, differences in interpretation, representation, and/or emphasis. For, *e.g.*, data clustering fuzziness or roughness can be desired for *allocating* entities to their appropriate level. Reasoning over granulated data and information and *retrieving* granulated information requires clearer distinctions that utilize precise and finer-grained semantics of granularity to obtain correct behaviour of the application software, due in part because it emphasizes a *qualitative* component of granularity, albeit not ignoring the quantitative aspects. For instance, reasoning with a partonomy (hierarchy based on the part-of relation) requires different functions compared to aggregating—calculating—precisely 60 seconds into a minute, 60 minutes into an hour and so forth.

The aim of this chapter is to elucidate foundational semantics of granularity, that is, identifying the different *kinds* of granulation hierarchies and, hence, *ways of granulation* of a subject domain. The outcome of this analysis is a top-level *taxonomy of types of granularity* that has a first main branching between scale-based and not scale-based granularity—or, roughly, quantitative and qualitative granularity—and has subsequent more detailed distinctions up to the current eight leaf types in the taxonomy. Each of these types of granularity has its own set of constraints, requirements for representation, and consequences for implementation. These differences can be used to model granulation hierarchies (also called granular perspectives) that are based on the *mechanism of granulation*; hence, the types of granularity are reusable both within a particular subject domain and across domains. This, in turn, simplifies reuse of program code or query formulation—as opposed to repetitive hard-coding—through enabling one to recognise and deal in a consistent way with different kinds of recurring granularity structures.

The remainder of this chapter is organised as follows. I first consider related works and analyse four different viewpoints on granularity. The taxonomy of types of granularity is introduced and elaborated on in the section after that, which also contains examples for usage in modeling and implementation. Last, we look at future trends and close with conclusions.

BACKGROUND

To provide an overview of the state of the art, we have to take a two-pronged approach. First, we briefly consider the scope of disciplines where granularity has played an important role, which reveals implicitly the various approaches toward granularity and differences in emphases of aspects of granularity. Second, these approaches will be thoroughly analysed and illustrated with examples in the second subsection.

Related works

This section contains a summary of existing approaches and solutions to deal with granularity, which range from informal usage of granularity without software support to formal partial theories for either qualitative or quantitative aspects of granularity. The literature can be assessed along three dimensions, being (i) Formal characterisations of granularity and ontological approaches, (ii) Engineering solutions that mainly emphasise the quantitative aspects of granularity, and (iii) Informal approaches to (biological) granularity. The first two items can be grouped as granular computing with the aim of *structured thinking* and *structured problem solving*, respectively, where the latter can be divided into methodologies and processes (Yao, 2005, 2007). When the former is clear, the latter ought to fall in place with ease to, in turn, solve the methodologies and processes. Conversely, the former has to match reality sufficiently to be applicable, thereby closing the loop. Such informal considerations will have to be taken into account when developing a generic theory of granularity.

Modelling: subject domain semantics and ontology. All contributions have implicit or partially explicit assumptions about the basic entities and relations that have to do or are needed for handling granularity, be it for modelling of a particular subject domain or ontological and formal investigations into the nature of such entities. These are depicted in *Figure 1* with indicative labels, such as the underspecified *precedes* relation (be it \prec or \preceq) between adjacent levels of granularity or granules, and the *links* relation to somehow relate the different hierarchies. In anticipation of the next section, we have added an appropriate place for where the taxonomy of *TypesOfGranularity* can fit.

Looking first at informal approaches, Tange *et al.* (1998) constructed granular perspectives and levels for medical practice based on term usage in literature that was intended for text mining and categorisation of scientific literature. Hierarchies, such as *Physical examination - Lungs - Auscultation* that informally combines a process, structural part of the human body and “type of observation”, have to be organised more clearly and preferably in an ontologically consistent manner. Other examples of informal approaches in biology and medicine are (Elmasri *et al.*, 2007; Grizzi & Chiriva-Internati, 2005; Hunter & Borg, 2003; Ribba *et al.*, 2006; Salthe, 1985).

On the border of biomedicine and formal approaches is Kumar *et al.*’s granularity for human structural anatomy and relatively simple *gran* function (Kumar *et al.*, 2005; Keet & Kumar, 2005). They have *GR* as the ordered set of levels of granularity applicable to a domain and *U* denoting the set of biological universals, so that *gran(x)* returns the level of granularity where the universal of interest resides. It assumes that granulated domain knowledge already exists and it requires patchwork in the logic, design, and implementation, as demonstrated with the 9 granular perspectives (granulation hierarchies) for infectious diseases (Keet & Kumar, 2005). This can be addressed using contextual information, *i.e.*, proper management of granular perspectives, thereby avoiding inconsistencies in the software system. A separate issue is its bottom-up

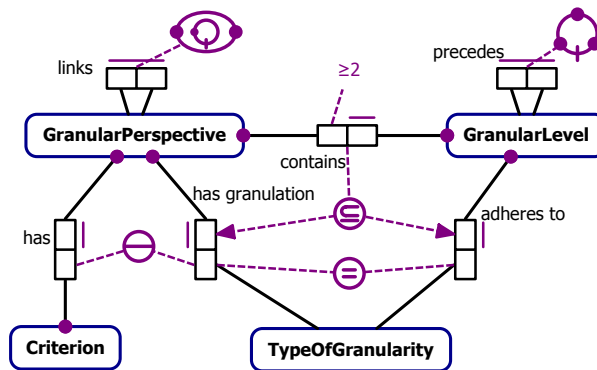


Fig. 1. Graphical depiction of the main entity types for granularity: A ‘granular perspective’ (granulation hierarchy) must contain at least two ‘granular levels’ and for each such related instances, both have, or adhere to, the same ‘type of granularity’ (mechanism for how to granulate) and a perspective can be identified by its type of granularity and its ‘criterion’ for granulation.

development of granular levels limited to human beings, which are not reusable in an expanded subject domain such as all eukaryotes instead of only humans. For computational implementations, however, an underlying domain-independent logically consistent theory of granularity is an imperative to meet requirements such as reusability, flexibility, and interoperability. Similar issues can be observed for conceptual data modelling for multi-representation geo-spatial databases in geographical information systems (Fent *et al.*, 2005; Fonseca *et al.*, 2002; Parent *et al.*, 2006a). An interesting addition for geographic information systems was made explicit by Camossi *et al.* (2003). They elaborated on the requirement for cross-hierarchy conditional information retrieval; for instance, where one has two granular perspectives, one for administrative regions and one for river sizes, then a realistic query is “if one makes a map with granularity at the Province-level then only rivers with a flow $\geq 10\,000$ litres/min should be included in the map”. One easily can imagine a similar example in the medical domain—though, to the best of my knowledge not raised yet—as, *e.g.*, “if the medical doctor needs a day-by-day view of the growth of the cancer in patient1, then deliver the tissue samples” as opposed to delivering cell cultures (Cell-level in a anatomy granular perspective with the Hour-level using time granularity) or microarrays (genes linked to the Minute-level).

Formal approaches motivated by engineering usefulness are restricted to a partial account of granularity and incorporate modelling decisions suitable for the engineering scope, such as data warehouse design (Kamble, 2004; Luján-Mora *et al.*, 2006; Malinowski & Zimányi, 2006), UML (Abelló *et al.*, 2006), and databases as linguistic corpus (Fagin *et al.*, 2005), and therefore are not easily transportable to other implementation scenarios such as Geographic Information Systems (GIS) and ontologies. Also, they have specified a large set of one-off functions and data manipulation operators only at the design or implementation layer, requiring a re-coding of functions, such as for calendar hierarchies and products sold; compare *e.g.*, *GMD*, *MSD*, *MADS*, and *MultiDimER* (Kamble, 2004; Fagin *et al.*, 2005; Parent *et al.*, 2006a; Malinowski & Zimányi, 2006), or see Euzenat & Montanari (2005) for an overview on theories of and functions for time granularity and Ning *et al.* (2002) for a particular example.

Hobbs (1985) has introduced several core components of granularity and Bittner & Smith (2003) have developed an ontologically-motivated formal “theory of granular partitions” (TGP) based on mereology. The TGP is relatively comprehensive and useful for granular levels, but it is limited to mereology, does not address the types of aggregation commonly used with data mining and conceptual data modelling, has no functions, no mechanism to deal with multiple granulation hierarchies for different perspectives, and does not allow for the kind of granularity and abstraction commonly used in biology or Mani’s (1998) folding operations in linguistics. There are few contributions on granularity from philosophy, which is addressed mostly within

themes such as hierarchical systems and emergent properties in biology (Cariani, 1997; Edmonds, 2000; Salthe, 1985; Wimsatt, 1995; Yao, 2005) where the main emphasis is on use of levels of detail to demarcate models to achieve better scientific explanations of natural phenomena and to address the limitations of those models with different theories for different levels of detail. Thus, it does not focus specifically on the ontological status or nature of what granularity is.

Granular Computing. Other types of implementations exist in different research disciplines, such as data mining and clustering techniques, which are grouped recently under the term Granular Computing, which focuses primarily on computational problem solving aspects. It combines efforts primarily from machine learning, data mining, and (applied) mathematics with fuzzy logic and rough sets. Lin (2006) summarises several example usages and Bargiela & Pedrycz (2006) and Yao (2007) describe background and trends. In this context, the comprehensive description of granule and granulation by Zadeh (1997) is useful for grouping together several notions about granular computing: “Informally, granulation of an object A results in a collection of granules of A , with a granule being a clump of objects (or points) which are drawn together by indistinguishability, similarity, proximity or functionality... In general, granulation is hierarchical in nature.”. The notions of similarity, equivalence, and indistinguishability relations have been well investigated with set-based approaches (Bittner & Stell, 2003; Chen & Yao, 2006; Hata & Mukaidono, 1999; Keet, 2007a; Mencar *et al.*, 2007; Peters *et al.*, 2002; Skowron & Peters, 2003; Yao, 2004). However, this set-based approach has issues that can be better addressed with mereology proper (Abelló *et al.*, 2006; Bittner & Smith, 2003; Keet, 2008a). The research programme of rough mereology (see, *e.g.*, Polkowski (2006); Polkowski & Semeniuk-Polkowska (2008) for recent results) has, from an ontological (Varzi, 2004; Keet & Artale, 2008) and logical (Pontow & Schubert, 2006) perspective, a comparatively weak mereology component, because it is tightly coupled with the set-based approach and remains close to Lesniewski’s pioneering work without considering newer mereological theories. For instance, modifying General Extensional Mereology, or a mereotopological version (Varzi, 2007), with the orthogonal roughness dimension would be an interesting and useful avenue to investigate.

Characteristic for these Granular Computing approaches is the applied mathematics, data-centric view, and quantitative aspects of data for problem-solving tasks, although the notion of “computing with words” (Zadeh, 1997, 2002; Broekhoven *et al.*, 2007; Mendel and Wu, 2007) clearly moves in the direction of subject domain semantics. The notion of a granularity *framework* with formally defined perspectives and levels is absent, but there are notable steps in that direction. Skowron & Peters (2003) have granule g as primitive, although they use it only for attaching lower and upper approximation bound to it. Yao’s (2004) comprehensive partition model based on rough sets¹ has been discussed before (Keet, 2008a). It is mainly quantitative, but recognises the need for qualitative aspects and seeks to accommodate a criterion for granulation although it does not have proper levels and only a lattice of sets. Rough sets’ approximation spaces can be lifted up to the ontological layer by, first, to conceptualise the various possibilities to augment any crisp theory admitting normal sets. We have drawn three options for such an *orthogonally positioned* extension to *Figure 1* in *Figure 2*. Observe that each option admits to a different ontological commitment, with the top two most different from the bottom one: the former assumes roughness (or, in analogy, fuzzyness) to be an *optional* property for granulation whereas the latter imposes that roughness is in some sense at least *mandatory*, if not *essential*, to any granular level; in the current paper, we do not commit to such a strong ontological commitment because one can identify granularity also for non-rough crisp data, information, and knowledge. In later work, Chen & Yao (2006) put more emphasis on granular perspectives, called “multiviews”, and a lattice as flexible granulation hierarchy. Qiu *et al.* (2007) make steps from set extension to concept, name a granular level a “granular world” that denotes a set of “concept granules”, have a mapping function to go from the finer- to the coarser level, and the union of such levels is a “full granular space”, which corresponds to what we refer to here as

granular perspective where one always must have *is_a* as relation between entities. This clearly moves in the direction of the TOG (Keet, 2008a), although it is limited to taxonomies only, and, most notably, misses a granulation criterion, a specification of the relation between the levels, and quantitative granularity.

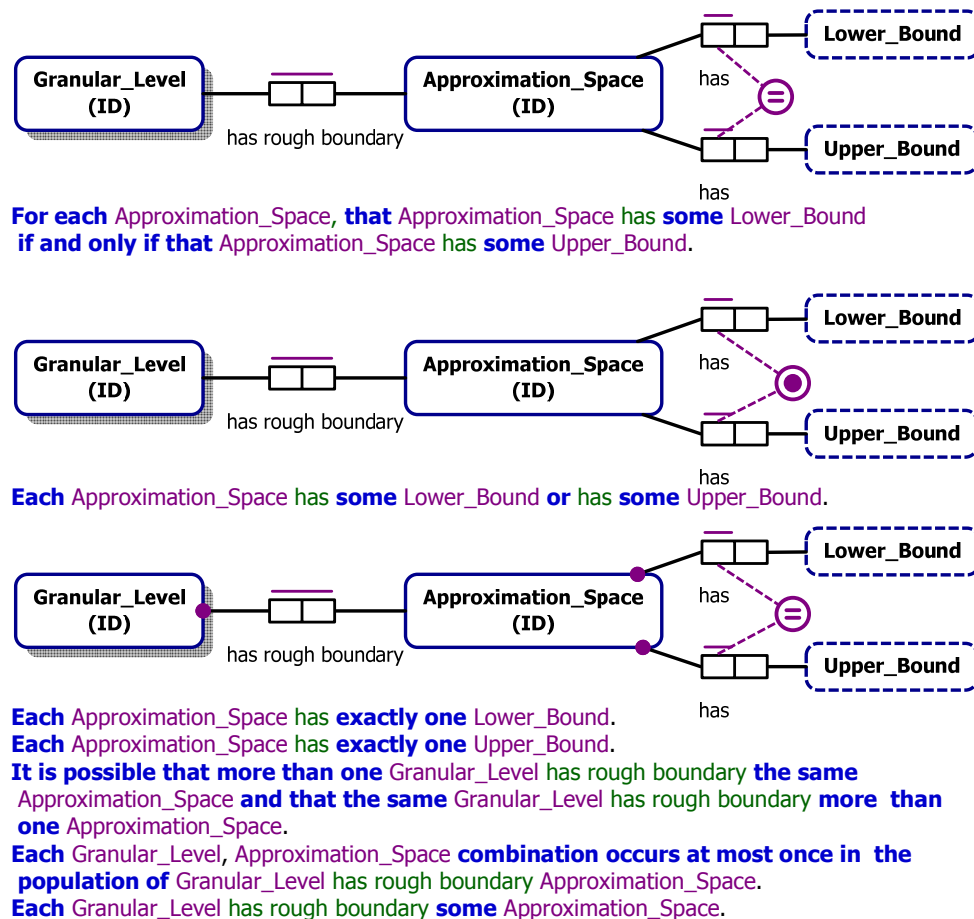


Fig. 2. Three options for adding the orthogonal notion of roughness with approximation values for the levels (or, in analogy, fuzziness) to granular levels. The top figure requires a bounded space when one specifies one of the values, the middle one permits specifying either one or both bounds, and the bottom one adds further mandatory constraints to impose approximation spaces with both lower and upper bounds for each granular level.

The principal other formal approaches with a computational scope for granularity are (rough and/or fuzzy) clustering, fuzzy sets, and to a lesser extent the combination of rough sets and fuzzy sets into fuzzy rough sets. The latter considers, among others, a fuzzy similarity relation to add another dimension with *degree* of similarity, and attribute reduction of fuzzy attributes using rough sets (refer to Chen *et al.* (2007) for recent results). Within the scope of fuzzy sets, one can distinguish the purely quantitative focus from the computing with words with, *e.g.*, linguistic fuzzy models for classification in the context of macroinvertebrate habitat suitability in aquatic ecosystems (Broekhoven *et al.*, 2007), which involves manual adjustment by domain experts of the numerical membership functions associated to the “linguistic values”. The named sets in the linguistic fuzzy model, however, do not yet deal with levels of granularity among the sets. Hierarchical and fuzzy clustering (Dawyndt *et al.*, 2006; Kaplan *et al.*, 2005; Vernieuwe *et al.*, 2007), among many, have quantitative granularity implicit in the mathematical models, which, ideally, should be made explicit to benefit transparency and to improve reusability. Limitations of such parameter-adjustable generated hierarchies are discussed by Zhou *et al.* (2005). They developed an algorithm to generate just that hierarchy whose data points group together into set-

extensions of the universals in the Gene Ontology. This, as well as Tsumoto's (2007) mining with a diagnostic taxonomy for headaches, could be interesting in conjunction with classification in logic-based ontologies and bottom-up generation or validation of the granular levels in a granular perspective. None of them, however, addresses that there are different kinds of granulation hierarchies and how and why they come about from a subject domain perspective.

Hata & Mukaidono (1999) explore granulation with fuzzy logic, where three classes of fuzzy information granulation are distinguished. First, their example for fuzzy Kleene classes uses informal granulation through mixing granulation criteria: first a transformation function from a colour gradient to a x-y plot with the usual range $[0,1]$ and then to select parts of the line. Second, their fuzzy probabilistic classes formalise the informal usage of more detailed attributes to calculate an overall probability for an event, which needs further investigation to take a structured approach to 'component-probabilities' of aggregated 'whole probabilities'. The third fuzzy information granulation is based on fuzzy Lukasiewicz classes, which, for the given example, amounts to fuzzy mereology (detecting anatomical parts in images of the whole brain), that, when worked out in greater detail, could be an interesting combination of qualitative with quantitative granularity and traditional bio-ontologies in the Semantic Web with fuzzy OWL-DL (Straccia, 2006); put differently: adding the orthogonal dimension for fuzziness alike depicted for roughness in *Figure 2*.

Thus, major themes addressed for computational problem solving are quantitative granularity and—like with DWHs and GIS—it takes a data-centric or ontologically poor linguistic approach toward granularity, whereas for conceptual data modelling, ontologies, and the Semantic Web, there is also the need to deal with both qualitative aspects of granularity and with the conceptual modelling and ontological analysis layers. In addition, while the mathematics-rich Granular Computing is good for mathematical foundations, it is relatively poor in incorporating subject domain semantics and lacks mechanisms for *how* the levels and hierarchies come about for a particular domain, such as administrative boundaries or human anatomy. Practically, many granulation hierarchies have been described in the above-cited literature across disciplines. For instance, we have cartographic maps that represent cities, provinces, regions, and countries, and a simple time granularity may start from second, to minute, hour, and day. With the latter hierarchy we can move from the chosen lowest level to the minute level by aggregating 60 seconds—that is, one uses a mathematical function—which cannot be devised consistently for administrative areas (*e.g.*, 'each province must be an aggregate of 15 cities' or 'exactly 5 provinces must comprise a region' does not hold in any country). By analysing hierarchies and, moreover, differences in emphases when modelling a hierarchy, common and differentiating characteristics can be uncovered. This analysis is the topic of the next subsection.

Analysis of different emphases regarding granularity

Granularity deals with organising data, information, and knowledge in greater or lesser detail that resides in a *granular level* or *level of granularity* and is granulated according to certain *criteria*, which thereby give a perspective—also called view, context, or dimension—on the subject domain, henceforth called *granular perspective*. A lower level within a perspective contains knowledge, information, or data that is more detailed than the adjacent higher level. Conversely, a higher level 'abstracts away', simplifies, or makes indistinguishable, finer-grained details. A granular level contains one or more entities, that is, representations of entity types or their instances; note that granular level is sometimes called granule, but we reserve *granule* to denote a cell or 'part of the pie'. Several interpretations of granularity and diagrammatical representations are shown in *Figure 3*, capturing subtle, but essential, differences in interpretation, representation, and/or emphasis. These differences in viewpoints are discussed in the remainder of this section. Successively, the emphasis will be on entity types & instances, the relation between levels and their contents, the perspective & criteria for granulation, and on consequences of choosing

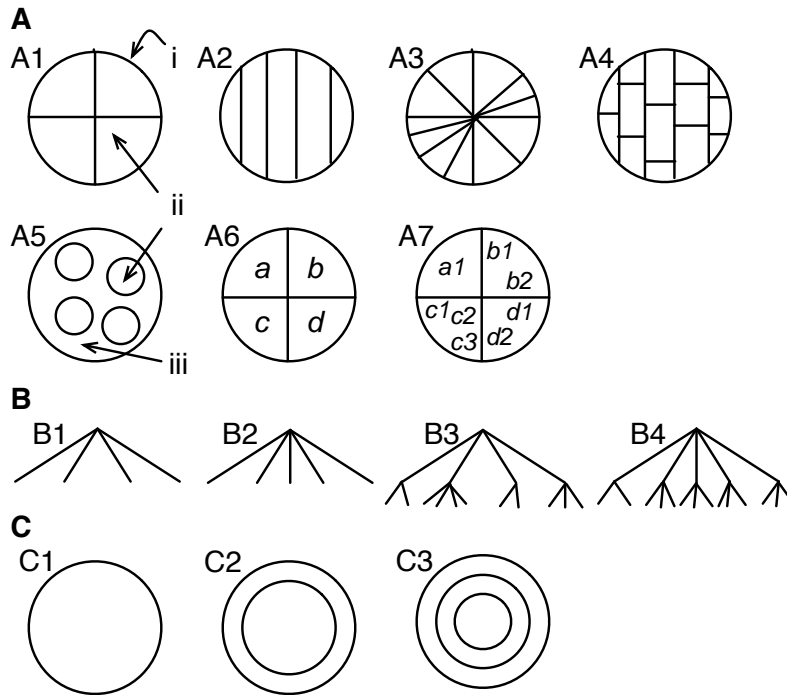


Fig. 3. Several graphical representations of granularity. A1/A5: (i) is the domain or a level with (ii) a partition and (iii) possible non-included rest depending on the interpretation. B1-B4 may be alternative representations of A1-A4. See text for explanation.

a particular formal representation. The main distinctions are summarised at the end of this section.

Emphasis on entity types and their instances. We first consider *Figure 3: A1-A5*. The circles A1-A4 in *Figure 3* are examples where the circles can represent the subject domain or a granular level. This gives four possible interpretations.

- i. If it represents a subject domain, then the four respectively five parts in A1 (A2) are finer grained than the circle, *i.e.* each one provides more detail about the domain than a plain circle (C1). With \prec denoting a strict order, then $A1 \prec C1$ and $A2 \prec C1$ hold.
- ii. If it represents a granular level, it shows the four (A1) respectively five (A2) granules resulting from granulating the contents of a level where each level is disjoint exhaustively granulated (fully divided). Without further clarification, it cannot be excluded that one of the granules denotes **Everything else**, or, if there is always one entity (/type) (A6) or possibly more (A7) entities in each granule (see also below on A6 and A7).
- iii. If the circles A1 and A2 are the same domain or granular level, then a different grid corresponds to granulation according to different perspectives or criteria on the same domain.
- iv. If A3 (resp. A4) is at a lower level of granularity compared to A1 (A2), then $A3 \prec A1$ ($A4 \prec A2$, respectively) and the granules of A1 (A2) are fully divided into more granules in A3 (A4), thereby representing finer-grained divisions that can be made when more details are taken into account, but which are indistinguishable at the level of A1 (A2).

From the possible interpretations assigned to A1-A4, A5 suggests that it contains four granules and empty space that falls outside the four nested smaller circles. However, it equally may be an inappropriately used ER diagram or Venn diagram requiring additional clarification to disambiguate its exact meaning regarding levels and granulation, such as if the four circles are disjoint exhaustive or not. Important to realise is that *Figure 3-A* has *implicit* the granular perspective with its criterion how to granulate: there is some criterion x why A1 has four parts

and another criterion y such that A2 has five, but x and y are assumed in the representation and the criterion for granulation—hence also the granular perspective—is omitted.

Figure 3: A6 and A7. The two circles containing a, \dots, d can represent a fundamental distinction on how to model granularity. Both A6 and A7 represent a populated A1, but depending on the interpretation of the figure, a, \dots, d in A6 can denote entity types or instances, and the indexed a_1, \dots, d_2 in A7 then denote instances.

i: If a, b, c , and d are entity types, then

- 1) *without* granulation as in C1, $\{a, b, c, d\}$ is an unordered set of entities of the domain of interest, thus $a, b, c, d \in D$ although in C1 they are indistinguishable;
- 2) *with* granulation, as in A6, $\{a, b, c, d\}$ are distinguishable and found to be distinct. Moreover, there is *exactly one* entity type in each granule, which can be either by design—one granule, one entity—or accidental in that there may be an e that also fits in the granule where, *e.g.*, a resides but is not included due to either unintended omission or known incomplete coverage of the domain.

ii: If a, b, c and d in A6 are instances, then either

- 1) the current 4 granules are accidental in the sense that at a time $t_1 > t_{present}$ there may be more or less than 4 granules because at least one of the objects may have ceased to exist or a new one added, or
- 2) at t_1 where an object has ceased to exist, there is an empty granule; hence, one commits to the original granulation for a set of instances. Thus, it can be that at time t_2 , where $t_2 > t_1$, that granule is not empty anymore.

In both options, the outcome of granulation is dependent on the instances present at the time of identifying or creating the granules.

iii: If it is the case of i-2, then either

- 1) A7 shows the corresponding instances of the entity types in A6, or
- 2) there was an unordered set of instances $\{a_1, \dots, d_2\}$ that were grouped according to some criterion. Based on their similarity, they are grouped into their corresponding classes as in A6 that may or may not correspond to universals.

iii-1 and iii-2 are different only in the starting point, one being the entity types and the other instance-motivated. Some of the points in this and the previous paragraph will be illustrated with an example; the new topic it introduces about measurement scales will be discussed afterwards.

Example 1 Let the circles A1 and A2 each represent a human population, where its parts (granules) are labelled with, respectively:

a1: (Single, Married, Divorced, Widowed)

a2: (Newborn, Child, Adolescent, Adult, Elderly)

Hence, A1's criterion is **Marital status** and A2's criterion can be **Life stage**. Considering A7, let a_1, \dots, d_2 be (not necessarily exhaustive) instances of months, then each granule could represent the quarters:

a7: (Quarter1, Quarter2, Quarter3, Quarter4)

This indicates that the domain or coarser-grained level C1 is **Year**; if there were semesters, then an intermediate level x would have to be added, such that $A7 \prec x \prec C1$ and x having two granules for **Semester1** and **Semester2**. More importantly, this granulation uses a 'smallest element': in this closed-world assumption, **Month** is chosen as the arbitrary atom (or *Urelement*) that is aggregated in such a way that each aggregate denotes a set extension of a class. Another example of this concerns phone points, where the phone point instances are granulated into **Cell**, **Land line**, **Direct line**, and **PABX** where a class was created from sets of phone points (Kamble, 2004). Thus, this relies on set theory for representing granularity, where each set neither necessarily must be the set-extension of a universal nor a defined class.

Another aspect of the figures in A is, *e.g.*, granulating temperature using a measurement scale in a lower grain size of integer degrees 19, 20, 21, for isotherms and moving up

to a higher level where *Isotherm20* suffices with coarser-grained rounding off (see also *Example 3*). In both cases, the *same* thing is granulated with more or less detail. This interpretation is prevalent in GIS for making a grid over land plots. Analogous is the case with spatial and time scales that for, *e.g.*, humans cover factor differences of 10^{15} for spatial and 10^9 for time, ranging from proteins (in nm) to height of humans (in m) and from μs for Brownian motion to decades for lifespan of a human, respectively (Hunter & Borg, 2003). \diamond

It is essential to note that when granulating according to a *scale*, one defines a *smallest unit* or a *standard unit* from which other levels are generated using a mathematical formula, according to which the domain is to be granulated. This is another, less problematic, granulation compared to trying to figure out the relation between *Tissue* and *Cell* or *Cell* and *Organelle*, if and how developmental stages of an organism have granularity, or characterising the type(s) of components of the *Second messenger system* that comprise distinct objects, its parts, processes, events etc. The second section of *Example 1* above deals with larger or smaller parts of arbitrary scales, but each level still concerns values according to the same arbitrary scale. Non-scale-dependent finer levels involve *other* types of entities, as, for instance, a biological cell is not equal to a tissue slice of 0.05mm thin. The latter puts a higher emphasis on the criterion for granulation and its levels than on the entities and instances one may find at a certain level of detail. This is especially useful for biological granularity, because of the incomplete knowledge of the domain that prevents disjoint exhaustive categorisations of its contents and its emphasis on qualitative information and knowledge as opposed to quantitative data. Granularity comprises both methods, but they involve fundamentally different granularity between coarser and finer levels.

In addition, while a1 and a7 in *Example 1* may seem alike, they are not: members of a population are different from elements in a set. With the latter, granularity *depends* on its instances: with another set of instances, the levels of granularity, ordering of the elements in a level, and perspectives may turn out to be different, and therefore can be time-inconsistent. In contradistinction, granulation involving an entity type identified with a collective noun like (human) population and how one can group the *members* of the population: from time t_0 to a later time t_1 the instances (members of the human population) have changed, but this does neither affect the principle/criterion nor the levels.

Emphasis on relation between entities and levels. Continuing with the possible interpretations, we proceed to *Figure 3: B1-B4*. Two first basic observations are that

- i. B1-B4 correspond to A1-A4, where the top equals the circle and each edge leads to a node (cell) at the end of each line. The tree structure is favourable when depicting multiple granular levels, because it is more concise than the figures in A (compare A3 and A4 to B3 and B4, respectively).
- ii. The lines in B emphasise the relation between levels of granularity, or at least between its entities (/types) residing in coarser- and finer-grained levels.

Point ii highlights the point of departure or focus—the relations involved—but is ignorant about which types of relations are relevant for granularity, both regarding the relation between the entities in different levels and how granular levels relate to each other. Committing to one type of relation or the other can imply an ontological commitment how one formally represents granularity (see below); in particular, partonomic versus taxonomic (generalisation/specialisation) granulation that are used or considered regarding informal biological granularity (a.o., Degtyarenko & Contrino, 2004; Fonseca *et al.*, 2002; Zhang *et al.*, 2002; Pandurang Nayak & Levy, 1995; Kiriyaama & Tomiyama, 1993). Such deliberations for one type of relation or the other is a distinct issue from using arbitrary scales and puts in the background the entities (/types) in each level and how the contents is allocated to a level. In addition, it may be that there is a taxonomic division for contents within a level, as depicted in *Figure 7*. Using the *is_a* relation for granulation means that each layer in the tree with the same depth should correspond to a

granular level. However, this does not necessarily hold for granularity as perceived by domain experts. For instance, ‘folding’ deals with polysemy and underspecification in language and the so-called black-box usage in biology, which is illustrated for cell physiology and book ordering in *Example 2*.

Example 2 Combining different types of entities and relations between granular levels may be useful in particular for abstracting biological complex types like **Second messenger system** or **MAPK cascade**. With the former, its processes such as **Activation**, **GTP-GDP exchange**, **α -subunit release**, states like **Activated**, and components such as **Hormone receptor**, **G_s protein**, and **cAMP**, collapse together into one entity type **Second messenger system**. **MAPK cascade** is already used as a module in systems biology that at a higher level of abstraction is treated as a black box, containing (sub-)processes, inputs/outputs, parameters and their values, etc. (Sontag, 2004).

A variant not uncommon in hierarchical modeling of conceptual data models is to have, *e.g.*, an entity type **Book order**, where the ordering consists of several procedural processes and entities involving, among others, **Billing**, **Paying**, **Supplier**, and **Shipment**. \diamond

As the example shows, that what is a type of endurant at the higher level of granularity, is composed of a combination of endurant parts, processes, and states. For an implementation, it is possible to separate the different types of components into different granular perspectives and levels, but this does not capture what is meant with the higher-level entity type like **Second messenger system**. Put differently: if separated in a granular perspective of structural components and another one for processes, the “Second messenger system” at the higher level in each perspective is only a *partial* representation of the entity type. If one allows relating levels of granularity by folding with type shifting (Mani, 1998), then this complicates what the parts are and how they relate to the whole, but on the other hand, saves integrating or linking granular perspectives. Either way, the relation-view between levels and between the entities (/types) is there; which granulation relations can be used and how will be summarised in section 1.

Emphasis on the perspective and criteria for granulation. Last, *Figure 3: C1-C4* show three levels of granularity where a smaller circle denotes a finer-grained level. This is unlike the Russian dolls analogy, where a similar smaller doll is contained in the larger one, but alike dissecting an organism to see what organs are inside, zooming in on parts of the organ, the tissue, cells and so forth. Thus, the parts are different types of entities and one uses, *e.g.*, human structural anatomy to identify finer- and coarser-grained levels that contain, respectively, all types of organs, tissues, and cells. In addition, each level has its distinguishing characteristic, that is, the property of a level is emphasised. In contrast with the first approach, one looks first at the property or properties, decides on the levels, and only possibly subsequently allocates entities to the levels based on the pre-selected properties.

Less explicit is how these properties relate to each other, except that it must relate in some way to both the finer- and coarser-grained level. I call the unifying rationale that links these properties the *criterion*. For instance, within the domain of human anatomy, one can granulate according to different criteria, such as structural anatomy, functional anatomy, or the processes they are involved in. Subsequently, one can identify granular levels according to certain properties that have to do with structural aspects or with containment and so forth. Observe that this entails a commitment to a granulation relation.

Because of the property-focus, C1-C3 do not bear any information if the cascaded granulation of the contents in each level is disjoint or complete. It does suggest that each level of granularity has one *type* of entity, such the outer circle representing the **Cell**-level containing cell types, with a smaller circle the **Organelle**-level containing entities such as **Endoplasmic reticulum** and **Lysosome**. Although these examples may indicate the physical size is a criterion, this is not necessarily the case. For example, if one were to represent the phylogenetic tree in the diagrammatic representation of B or C, a **Mammal**-level has no physical size associated with its definition.

More generally, with the emphasis on the perspective and criteria for granulation, this approach is more useful for non-scale dependent granularity.

Emphasis on formal representation. The difference between scale and non-scale dependency mentioned in the previous sections roughly fits with Sowa’s (2000) epistemic and intentional granularity. Sowa bases his three types of granularity on Peirce’s three categories of Firstness, Secondness and Thirdness. Firstness then maps to actual granularities with axioms for discrete, continuous or lumpy aggregates (Sowa (2000) and below) and concerns the entities that populate a level. Secondness for granularity uses epistemic logics involving measurements, including error ranges, or axioms & measurements (Sowa, 2000) and corresponds to the scale-dependent granularity with fuzzyness and roughness in allocating objects to their level of granularity. The Thirdness for granularity, corresponds to intentional, which requires a three-place predicate relating “an agent a to an entity x for a reason r ” (Sowa, 2000), where a reason r depends on the perspective on takes. However, depending on how one uses granularity in a subject domain, devising *levels* does not require asking oneself questions if entity x has at least one atom as part, if there is an infinite regress of parts that is cut at the lowest level defined, or if the entity is lumpy, but the allocation of entities to a given level does use aggregates and entities. More precisely, in mereology an *Atom* is an entity that has no proper parts (1).

$$Atom(x) \triangleq \neg \exists y(y < x) \quad (1)$$

Then, there are three kinds of aggregates (with “ \leq ” as part-of and “ $<$ ” as proper-part-of). First, *Discrete*: everything has at least one atom as part (2); thus, that things can be subdivided up to the point where nothing is left but atoms.

$$\forall x \exists y (Atom(y) \wedge y \leq x) \quad (2)$$

Second, *Continuous*: everything has at least one proper part (3), which permits indefinite subdivision, implying that there are no atoms,

$$\forall x \exists y (y < x) \quad (3)$$

Third, *Lumpy*: some things are atoms, some are continuous (4). (Sowa, 2000).

$$\exists x Atom(x) \wedge \exists y \forall z (z \leq y \rightarrow \exists w (w < z)) \quad (4)$$

Thus, representing granularity using mereology may have *but does not require* atoms as ‘ultimate part’ or Urelement that is used for set theory-based granularity. Observe also that Urelement can, in fact, be defined in terms of atoms, where Urelement is the “atom at the finest-grained level in a granulation hierarchy”, provided that ‘granular level’ and ‘granulation hierarchy’ are defined (*e.g.*, as in the TOG by Keet, 2008a). Both set theory and mereology have their advantages and disadvantages for representing granularity that better approximates reality. Ease, difficulty, or even impossibility, to identify an Urelement is illustrated in the following example.

Example 3 Let us take calendar entities and set-theory based granularity. Entity types such as *Week*, *Month*, *Quarter*, and *Year* can be defined based on a chosen Urelement *Day* and then can be represented by distinct sets of days. However, if we take isotherms, then what has to be chosen as Urelement? If one uses *Degree* as smallest element to build coarser-grained isotherms, then with a set as the extension of *Isotherm20*, like $\{15, 16, 17, 18, 19, 20, 21, 22, 23, 24\}$, and where *Isotherm20* is a subtype of *Isotherm* that has other subtypes (such as *Isotherm30*), there are two problems: the extension is not the entity type and the numbers are not degrees but integers (see also Johansson, 2004b). Within the subject domain of biology, identifying or choosing a smallest element is more challenging. In one scenario, a general practitioner who is not interested in smaller entity

types than tissue will make Tissue the Urelement (atom) to populate the lowest level, but this would also mean that all higher-level entities are composed of tissue *only*: we *know* this is biologically incorrect and thereby not a good representation of reality. Moreover, if one takes the lowest level of the Foundational Model of Anatomy (FMA) (Rosse & Mejino, 2003)—*i.e.*, Biological macromolecule, which does not include other molecules without which a human body cannot survive, such as H_2O —and deem that all coarser-grained levels up to Body are varying sets of macromolecules, then a body changes identity each time a molecule is synthesised/metabolised, which happens continuously, resulting in the situation that a body has no enduring identity but is in flux². In a similar fashion, entomologists study the same ant colony over time, even though ants were born and have died. More generally, regardless if a set-theoretic logical theory or model is logically valid and corresponding knowledge base in a legal state, basing reasoning on represented knowledge that is not adequately grounded in the reality it aims to represent can lead to undesirable outcomes for patients, ecosystems and the like. \diamond

Both ways of representing granularity, through *is_a* with set theory and mereological *part_of*, are from a logical viewpoint mostly interchangeable (Pontow & Schubert, 2006), but not from an ontological viewpoint as the intended meaning captured in a formalisation is distinct. This difference has been recognised earlier by Salthe (2001) and are not considered to be *competing* interpretations of granularity, but *both* considered as distinct, valid ways of understanding granularity. One does not have to force one type of granularity in the straightjacket of the other; doing so anyway always will deprive another type of granularity from representing nature as accurate as possible.

Moving to the notion of ‘thirdness’, reason *r* might be useful for granular perspectives in non-scale dependent granularity: although it is not necessarily modelled as a triadic predicate, separating and reusing the reason, or criterion, benefits scaling up the granularity framework. Such differences in types of granularity have, at the meta-level, a major effect on granulation relation between entities (/types) residing in different granular levels, because scale-dependent levels are identified and ordered according to a combination of a property and an arbitrary scale whereas non-scale-dependent levels are ordered according to a combination of properties where level identification is less straightforward. Properties will be analysed in detail in the next chapter.

Main differences concerning approaches toward granularity. An attempt to merge the graphical representations depicted in *Figure 3* is shown in *Figure 4* for two granular perspectives, where the top ellipses are coarse-grained granular levels with less detail in larger cells—that is, conceptually more encompassing entities—than the two finer-grained granular levels (iii).

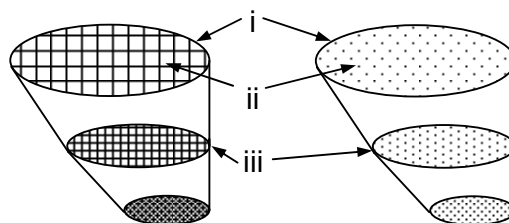


Fig. 4. Merging emphases on aspects of granularity (A-C of *Figure 3*): top ellipse (i) is a coarse-grained granular level granulated with less detail in larger cells or coarser-grained entities (ii) than in the finer-grained granular level (iii).

Summarizing, one can identify the 4 principal dimensions for types of granularity and the perception thereof:

1. Arbitrary scale versus non-scale-dependent granularity;
2. How levels, and its contents, in a perspective relate to each other;
3. Difference in emphases, being entity-, relation-, or criterion-focused;
4. The perception and (mathematical) representation, such as based on set theory versus mereology.

These differences do not imply one cannot switch from one to the other, represent one way into another, or let them work together orthogonally. When analysing of some subject domain, one apparently seamlessly shifts perspectives and alternately emphasises the criterion used for granularity and the partitioning within a level itself, or taking a type versus instance-inspired approach. Teaching a computer program to do so, however, requires a formal approach to implement it in a consistent manner that can be used and reused across different types of software applications.

TAXONOMY OF TYPES OF GRANULARITY

Given the types of granularity informally introduced in the previous paragraphs, they will be structured into a taxonomy of types of granularity in this section. Hereby it is emphasised that there is not one granularity, but several types—mechanisms of granulation—that have additional constraints extending the core **G**ranularity, **cG**, as root. *Figure 5* shows the top-level taxonomy, where the meaning behind the labels of the types are important, and *Table 1* summarizes the distinguishing characteristic at each branching point. This will be elaborated on in the next two sections by first showing a general bird’s eye view and then providing a description of the characteristics in detail in the second subsection. We consider several typical examples more comprehensively in the third subsection and also give an outline where and how the taxonomy of types of granularity can simplify implementations.

Overview of the top-level taxonomy

In this section, the types of granularity and the distinguishing characteristic at each branching point in the taxonomy are briefly described to give a general idea.

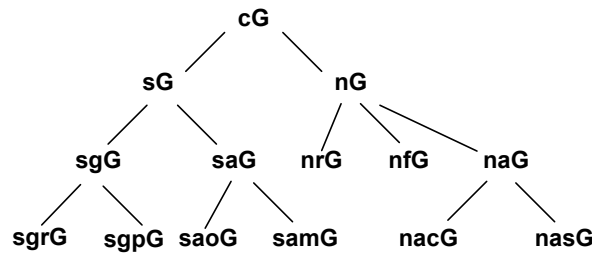


Fig. 5. Top-level taxonomy of types of granularity.

- ★ **cG**: core **G**ranularity, consisting of the basic characteristic common to all considered types of granularity and basic constraints.
- ★ **nG**: non-scale-dependent **G**ranularity, where other types of entities reside in each finer-grained level; subtypes have additional constraints and granulation relations.
- ★ **nrG**: levels of non-scale dependent **G**ranularity are ordered according to one type of relation in a perspective; *e.g.*, (structural-)*part_of*, (spatially-)*contained_in*. The primary types of granulation relations were identified and formally defined (Keet, 2008a; Keet & Artale, 2008) and include, at least, *is_a*, *participates_in*, *member_of*, and *proper_part_of* with its subtypes *contained_in* and *involved_in*.

Table 1. Distinguishing characteristics at the branching points in the taxonomy of types of granularity depicted in *Figure 5*.

Branching point	Distinguishing feature
sG – nG	scale – non-scale (or, roughly: quantitative – qualitative)
sgG – saG	grain size – aggregation (or: scale <i>on</i> entity – scale <i>of</i> entity)
sgrG – sgpG	resolution – size of the entity
saoG – samG	overlay aggregated – entities aggregated according to scale
naG – nrG – nfG	semantic aggregation – one type of relation between entities in different levels – different type of relation between entities in levels and relations among entities in level
nacG – nasG	parent-child not taxonomic and relative independence of contents of higher/lower level – parent-child with taxonomic inheritance

- ★ **nfG**: levels of **n**on-scale dependent **G**ranularity are ordered by simultaneous **f**olding ≥ 2 different (types of) entities, such as folding events and states, and consequently folding relations between those entities, upon going to a coarser-grained level; *e.g.*, the ‘black boxes’ in biology such as the **S**econd **m**essenger **s**ystem, the **A**bstraction **H**ierarchy, **E**R clustering.
- ★ **naG**: **n**on-scale-dependency with some form of **a**ggregation.
- ★ **nasG**: **n**on-scale-dependency using **a**ggregation of the **s**ame collection of instances of one type that subsequently can be granulated using semantic criteria. The class at a lower level is a subtype of the class at the coarser-grained level; *e.g.*, a collection of phone points and at the finer-grained level we have land-line and mobile phone points.
- ★ **nacG**: **n**on-scale dependency using **a**ggregation attributed to the notion of an entity generally labelled with a **c**ollective noun, has an existing semantics, the instances of the aggregate are different from instances of its members, and a change in its members does not affect the meaning of the whole; *e.g.*, **P**opulation with **O**rganisms of type **x**, or **T**eam as aggregate of its **P**layers.
- ★ **sG**: **s**cale-dependent **G**ranularity where the contents is structured according to a more or less obvious arbitrary scale; for **cG** and additional constraints. For instance, calendar hierarchy, rounding off of altitude lines on a cartographic map.
- ★ **sgG**: scale dependency with relation to **g**rain size, or resolution, scale-based zooming.
- ★ **sgrG**: scale dependency, taking into account **g**rain size with respect to **r**esolution; *e.g.*, **C**ell **w**all represented as line, as lipid bi-layer, and as three-dimensional structure, or a building on cartographic maps as polygon or as point depending on the resolution of the map.
- ★ **sgpG**: scale dependency, with **g**rain size and **p**hysical size of the entities; *e.g.*, sieves with different pore sizes that retains the entities or lets them through, a Euro coins separator, or two objects touching each other (*e.g.*, wallpaper and the wall).
- ★ **saG**: scale dependency with some form of **a**ggregation and its immediate parts are of one type.
- ★ **samG**: scale dependency and using **a**ggregation of the **s**ame collection of instances of the same top type or **U**relement that subsequently can be granulated in various ways at lower levels of detail using a mathematical function; *e.g.*, **S**econd, **M**inute, and **H**our, where 60 seconds go in a minute.
- ★ **saoG**: scale-dependency the carving up of the same entity at each level according to a coarser or finer grid of which the cells can be **a**ggregated and lay **o**ver the representation of a material entity³. For instance, the earth with its isotherms, where the isotherms are in steps of 10 degrees, 5 degrees, 1 degree detail⁴.

The current version of the taxonomy of types of granularity roughly fits the distinction between quantitative and qualitative features and added versus inherent granularity. At some branching point in the taxonomic structure, more than one desideratum is used to distinguish between the subtypes, which can be remedied by introducing ‘fillers’ to ensure only one desideratum at a time is added. However, these fillers are not used anyway and unnecessarily enlarge the top-level structure, and therefore have been omitted. Other categorisations of types of granularity are

conceivable, but these are much less consistent and structured. For instance, aggregation versus ‘granularity by other means’, instead of the (non) scale dependency because one has both **saG** and **naG** each with their subtypes. However, using aggregation emphasises the internal structure of a level, how entities and instances relate, or is implementation-driven, but it does not take into account the properties *how* to make the distinction between types because having a remainder group of types of granularity does not capture the semantics adequately and then the same desiderata would re-appear in both branches thereby creating redundancy. In addition, using aggregation as distinguishing criterion *implicitly* makes a distinction between set theory and mereology, but this should be a representational issue only. Last, aggregation is underspecified, both with respect to its ontological nature and variants in implementations. There are two other options to categorise the types, being entity-focussed and human-centred, which might aid understanding of the types. However, because several types can be categorised twice, it does not serve to devise an unambiguous classification⁵. In contrast, the proposed taxonomy takes a purely *semantic*, ontological, approach, thereby also separating restrictions of (formal) representation and implementation from the intended meaning.

Characteristics in detail

A consequence of different types of granularity is the influence on the structure of the contents of each level, independent of the actual data source. For instance, completeness and disjointness: a grid is automatically disjoint and, depending on the level and implementation decisions, complete, which does not necessarily hold for contents of levels that has a **nG**-type of granulation. It also affects the allocation of contents into the levels and reasoning over it, which will be illustrated in section 1. A first formalization of the structural aspects of the contents of a level for each leaf type was presented in (Keet, 2006a). The drawback of that formalization is that it requires many primitives and, as it turns out, we can avail of the Theory Of Granularity, TOG (Keet, 2008a), to provide a more elegant formal characterization. However, it is outside the current scope to introduce the TOG. Put differently, (i) The TOG—or any other theory of granularity—does not have as prerequisite a comprehensive formalization of the taxonomy of types of granularity; (ii) A precise characterisation of the types of granularity is very useful to grasp several modeling decisions for development of a theory of granularity, but not mandatory; (iii) Only an implementation of comprehensive granular reasoning needs both. Therefore, in this chapter, we only formalise aspects where possible and clearly indicate which kind of predicates one would need in a theory of granularity. To be precise about the formalisation in this chapter, we need a few preliminaries about first order logic, such as described in (Hedman, 2004), classes, universals and particulars (*e.g.*, Earl (2005); MacLeod & Rubenstein (2005); Smith (2004))⁶, and have to introduce several basic granularity functions and relations. For the present purpose, it suffices to let GL denote *granular level*, which is a unary predicate, U as universal and PT a particular, \mathcal{DL} denotes the set of levels in the domain granularity framework (D^f , which contains all granular perspectives and their levels), and D^s denotes the subject domain (not to be confused with a specific data source, such as an instance of the FMA database). Given these preliminaries, (5) says there is a relation between an entity (/type) x and the level it resides in and with the *grain* function (6) we retrieve level y where x resides; an alternative notation for (6) is $\forall x(\text{grain}(x) = y \rightarrow D^s(x) \wedge GL(y))$. The $\text{assignGL}(x, y)$ function assigns an entity (/type) x to a granular level y , where $U(x)$ or $PT(x)$ and $GL(y)$ (7, 8).

$$\forall x, y(\text{in_level}(x, y) \triangleq ((PT(x) \vee U(x)) \wedge GL(y) \wedge \text{grain}(x) = y)) \quad (5)$$

$$\text{grain} : D^s \mapsto GL \quad (6)$$

$$\text{assignGL} : D^s \times GL \quad (7)$$

$$\forall x, y(\text{assignGL}(x, y) \rightarrow GL(y) \wedge (PT(x) \vee U(x)) \wedge \text{in_level}(x, y)) \quad (8)$$

Further, one can enforce that each entity must reside in a granular level with (9).

$$\forall x(D^s(x) \rightarrow \exists y(\text{grain}(x) = y \wedge GL(y))) \quad (9)$$

Put differently, it represents an ontological commitment that the world—or at least the subject domain under consideration—is granular, which may or may not be truthful to reality. Moreover, this constraint may be too restrictive for deployed information systems where for some part of the D^s either the granulation is not known or beyond the interest of the domain experts and software developers.

Characteristics of the eight leaf types. The 8 leaf types inherit characteristics from their parent type of granularity. There are several general conditions that the structure of the entities (/types)⁷ within a level must satisfy, which all types inherit from **cG**. Based on the previous section, one can draw a preliminary list of general characteristics for the entities (/types) that are granulated according to any of the subtypes of **cG**.

- i. The contents of a level can be either entity types or instances, but not both.
- ii. The entities (/types) in a particular level have at least one property (value) in common.
- iii. The entities (/types) are disjoint, but not necessarily exhaustive due to our gaps in knowledge of nature. Within a closed world assumption, they are disjoint exhaustive.
- iv. Provided an entity (/type) is not an orphan and the subject domain is covered fully with granular perspectives, it must reside in at least one granular level.
- v. An entity (/type) never can reside in more than one granular level within the same perspective and that entity (/type) is classified as the same (instance of) universal.
- vi. The entity (/type) in a granular level may reside also in ≥ 1 other levels, provided that each level the entity (/type) resides in is contained in a distinct granular perspective.

In addition, only lists of ingredients are given to formally characterise the eight leaf types, whereas an elegant, comprehensive formalization that relies on both the TOG and a generic foundational ontology, such as DOLCE (Masolo *et al.*, 2003), is a topic of current work. For instance, the underspecified “ \prec ” over levels can be defined (called *RL* in the TOG), several of its properties proven, and a clear distinction can be made between the relation between levels and the granulation relation between entities (/types) residing in the levels. Further, notions such as *endurant*, *ED*, *physical region*, *PR*, how they relate, and so forth have precisely defined meaning with constraints in DOLCE, whereas the intuitive primitives that will be suggested for several types of granularity may be cast in terms of DOLCE categories and constraints, too, after closer investigation. The respective characteristics for the eight leaf types of granularity are then as follows.

saoG. All instances in the ordered set belonging to a particular level are instances of the same type. The amount of whole instances is not necessarily determined by the size of the entity that is granulated (see section 1 for an example with granulating a lake). The instances are automatically disjoint because the granulation results in a grid. In addition, the instances within the same level make up the set-extension of its corresponding universal, such as a set of plots of km^2 where the amount of plots depends on both the entity that is granulated and on the decisions to include or discard ‘partial’ plots where a cell of the grid covers a larger area than the part of the entity. Combining these constraints, we need to represent, at least:

- a notion of region to represent the cells of the grid (*e.g.*, DOLCE’s region *R*) and that these cells are of the same type, so that if we denote the granulation region with, say, *granR*, then $\forall x(\text{granR}(x) \rightarrow R(x))$ and for a particular grid where the shapes are squares (with squares defined the usual way), then we have also that for that particular granular perspective $\forall x(\text{granR}(x) \rightarrow \text{Square}(x))$ holds;
- a way to relate the boundary of one cell to another and to state these cells are disjoint;
- the entity (/type) to which the grid is applied (*e.g.*, DOLCE’s *endurant ED*);

- that the endurant can be associated to each level in the granular perspective, but granulation regions of certain size are in one level of granularity only and those of a larger measure must be in a coarser-grained level than those of a smaller measure.

samG. All instances in the ordered set belonging to a particular level are instances of the same type and they are whole instances at that level. Further, there is an exact, known, number of instances that can be in that level. In addition, the entities and instances at the higher levels are ultimately composed of the chosen Urelement at the lowest granular level. In contradistinction with **saoG**, the set is grouped into particular amounts, like {Hour 1, ..., Hour 24} at the Hour-level y_i , which are ultimately built up from the same Urelement, such as Second at level y_{i+2} . Combining these constraints, we need to represent, at least:

- Urelement defined as the arbitrarily chosen atom at the most fine-grained level in the granular perspective, which may or may not be Atom *sensu* (1);
- a function to calculate the amount of elements that have to be aggregated and to relate that function to the level;
- the granulation relation between the aggregates in different levels versus the relation between granular levels;
- that the aggregates in different levels are the set extensions of different universals (as opposed to arbitrary aggregates).

sgpG. This involves a ‘zooming in’ and ‘zooming out’ factor, where at a coarser-grained level, *e.g.*, the wall and wallpaper touch each other, but at a greater magnification, there is wall-glue-wallpaper, and again in smaller detail, one looks at the molecules in the paper, glue, and wall. The zooming factor is like a grain size when relating levels of granularity, where *within one* level one can distinguish instances of, *e.g.*, $\geq 1\text{mm}$ but instances $< 1\text{mm}$, metaphorically, fall through the sieve and are indistinguishable from each other, but are distinguishable at lower levels of granularity. In practice, this is used, *e.g.*, when filtering substances with filters having different pore sizes and dialysis tubes. With **sgpG**, differences in physical size of the entities (/types) is *the* property for granulation. To characterise the content, we need a function, *e.g.*, *size_of*, which returns a value in, say, length, square or cubic size, which can be categorised as physical regions alike DOLCE’s *PR*. Thus, the instances of an entity type ϕ , *inst*(x, ϕ) recorded at some instance of *GL*, say, level gl_j , are physically smaller than the instances at a higher level (gl_i), with $gl_j \prec gl_i$, and thereby are related to each other at least or possibly only by the relation that they fall within the same physical size range. Combining these constraints, we need to represent, at least:

- access to the measurement of the physical size of the objects, *e.g.*, with a function *size_of* : $PT \mapsto PR$ for the measured region (*PR*) of an object (*PT*), and comparison of measured regions so that, in rudimentary form, $size_of(x) < size_of(y)$ provided we have $in_level(x, gl_j)$, $in_level(y, gl_i)$, and $gl_j \prec gl_i$;
- that this measurement is taken by using a direct or indirect measurement property of the entities (/types);
- a value range for each level in the granular perspective and to ensure that those ranges of coarser levels are larger than those of finer-grained levels;
- that, consequently, the (type of) entities in different levels are different—following the first item, then $\forall x, y (inst(x, \phi) \rightarrow \neg inst(y, \psi))$ —and can also be different within the same level.

sgrG. The entities in reality associated with the levels are the same in the coarse- and finer-grained levels, but their representations change according to pre-defined resolutions. That is, the real world entity is the same universal or its instance, but one chooses to represent them as if they were instantiating different universals; *e.g.*, Cell wall as circle, lipid bi-layer, three-dimensional structure, or also considering the movements of the lipids and proteins. In this case, the resolution-motivated representation is a figurine where that at a coarser-grained level is, in

fact, a proper part of the figurine at the finer-grained level, as, *e.g.*, a point is a proper part of a polygon and circle a proper part of a sphere. This is common in cartography and GIS in general, where at a greater resolution, a street is represented as a single line, two parallel lines, or even more detail. Thus, there are several mappings from the same entity to different figurines, resulting in the situation where ordering the figurines with respect to the resolution (hence, also their attributes) are more important than the actual entity. To capture this multi-representation, we could introduce a primitive $rep_of(x, y)$ to denote the relation between the real-world entity and its coarser- or finer-grained representation $Rep(y)$. Combining these constraints, we need to represent, at least:

- associate a value to the granular level to represent the resolution applicable to the level;
- the entity (/type) that has a multi-representation (*e.g.*, DOLCE's endurant ED) and thereby that the endurant can be associated to each level in the granular perspective;
- the figurines with, say, $Rep(x)$ (but preferably a more detailed characterisation);
- the association of the entity (/type) with the figurines with, say, $rep_of(x, y)$ where $ED(x)$ and $Rep(y)$;
- that given any $rep_of(x, y_i)$ and $rep_of(x, y_j)$, then they must be in different levels;
- that given the previous item, if $in_level(y_i, gl_i)$ and $in_level(y_j, gl_j)$ and $gl_j \prec gl_i$, then $Rep(y_i)$ is a proper part of $Rep(y_j)$.

nrG. The entities in a level are of a different type, but all are of the same category, such as all being non-agentive physical objects (*NAPO*) or processes (*PRO*) and so forth. For instance, at the **Cell**-level, there are many *types* of cells, but they are all of the category *NAPO* structural component (Hemal cell, Leukocyte, ...), or function (Hormone excretor, Insulin excretor, ...), and so forth, or a **Protein unit structure**-level with items such as α -helices and β -sheets. Thus, the entities are structured in a hierarchy where the direct children are in a lower level of granularity than its supertype. The characteristic of the **nrG** type is the type of relation between entities, which is of the same type throughout; the currently suggested types of relation were given in section 1, which is denoted here with granulation relation GR . In the level, without further specification, the entities can be in an unordered set. It may be, however, that the content has some other additional structure within the level alike a **nasG**, or another **nrG** structure, as illustrated in *Example 4* and *Figure 7*. Alternatively, one can group the unordered set such that it takes into account the additional tree (or other) structure in the level, where each granule correspond to a different branch. Either way, the entities are disjoint thanks to the underlying structure in the data source. Combining these constraints, we need to represent, at least:

- the granulation relation GR between the entities (/types) that relate these entities (/types) residing in adjacent levels;
- that for each granular perspective only one granulation relation is used;
- the permitted granulation relations by which one can granulate the data.

nfG. The entities in a level can be of different kinds, such as folding *NAPOs* with their processes and states, combining types of entities into *one* entity residing in an adjacent higher level. It is not the case that the entities contained in the lower granular level is an (un)ordered set, but the entities (/types) are always related to at least one other entity (/type) within that level. For instance, the hierarchical modeling to improve comprehension of large conceptual data models that was illustrated in *Example 2* and different folding operations—that is, what is folded and how—can be identified, concerning perdurants and endurants and some of their subtypes (this has been elaborated on in (Keet, 2008a, 2007b)). Combining these constraints, we need to represent, at least:

- the assertion that the entities (/types) in a level are related to each other;
- the granulation relations between the entities (/types) and their relations in the finer-grained level as the domain of the relations on the one hand and the single entity (/type) they are are folded into in the coarse-grained level as the range on the other hand;

- the permitted granulation relations between the coarse-grained entity (/type) and the finer-grained entities (/types) and relations it expands into.

nacG. Like **samG**, all instances in the set belonging to a particular level are all of the same type and at that level they are whole instances. It is not necessarily the case that the amount of instances in a particular level is known and can be computed. For instance, **Sports team** does have a predefined amount of instances of **Player** per team, but sales department members of a company do not have to have always the same amount of members. The instances that are member of such populations change over time but the entity (/type), generally labelled with a collective noun, and its meaning endures. Thus, looking at the structure of the data in a level, it is at least an unordered set but can be an ordered set of instances, and the instances populating the set can vary over time, although the entity (/type) keeps its identity. It might be possible, to have not an (un)ordered set but a taxonomy or other additional aggregation within the level alike a **nasG** or **nrG** structure, such as an employee hierarchy (with **Junior sales person**, **Senior sales person**, **Trainee**, **Manager**, etc), or aggregated by the organisational unit (**teamA1**, **teamA2**, etc). Combining these constraints, we need to represent, at least:

- that the instances in the level instantiate the same type, *i.e.*, $\forall x(inst(x, \phi))$ for a particular level, or, at the type-level, that they are subsumed by a root entity type;
- that this type is, at least, a subtype of enduring *ED*, such as a social object and not defined by its extension;
- a notion of ‘membership’ of the entities (/types) in the finer-grained level as members of the entity (/type) in the adjacent coarser-grained level, such as through the meronymic *member_of* as granulation relation.

nasG. The structure of the data is like **samG**, but if one combines the subsets at each level, then the amount of unique instances residing in *each* level is always the same amount as they are instances of the chosen counting element. For instance, at level gl_1 there are 100 phone points and in a gl_2 , such that $gl_2 \prec gl_1$, the 100 phone points may be divided into three subsets Land line, Mobile, Phone over IP each with, say, 2, 35, 63 elements of the original set, respectively, hence, **Mobile** \subset **Phone point**. There may be a gl_3 with **Classic cell phone** and **Skype mobile phone** that granulates **Mobile phone** points and that each have 20 and 15 elements in the set, respectively, which adds up to the 35 elements for **Mobile** of the higher level gl_2 (assuming that **Classic cell phone** \cap **Skype mobile phone** = \emptyset , although in certain cases there may be a ‘rest group’). Thus, at each level there are subsets with instances as elements of the set that, depending on the granulation criterion, are disjoint. Combining these constraints, we need to represent, at least:

- define the counting element as the arbitrarily chosen atom at the most coarse-grained level in the granular perspective, which serves to count the number of instances in all levels;
- that the number of instances at a given time are the same for each level in the granular perspective, hence, are fully partitioned at each level;
- for all instances that are member of a class ϕ in a finer-grained level, their coarser-grained representations are instances of ψ residing in a coarser-grained level, *i.e.*, $\forall x(inst(x, \phi) \rightarrow inst(x, \psi))$ so that taxonomic subsumption (*is-a*) may be a relation by which to granulate the data;
- following from the previous point and proper taxonomy development, then the instances at some level gl_2 have in their representation either at least one more attribute or more constrained attribute values than their respective representation in the coarser grained-level gl_1 .

One may opt for the design decision to demand from the chosen criterion that the sets never overlap, or, for ‘just in case’, create two subtypes of **nasG** where one does allow overlapping sets and the other subtype does not. It does not merit a subtyping because the core ontological aspect is the same, but it may be useful for software systems to distinguish between these two cases.

Some types of relation between the entities or instances within a level can be combined, because one does not have to take into account that some are granulated according to arbitrary scale and others are not. (The (non-)arbitrary scale division is relevant for the relations between levels, but do not always act out on the relation between entity types or instances contained within a level.) **nasG**, **nacG**, **nrG**, and **sgG** may be unordered sets, **samG** and **saoG** may be ordered sets, and **nfG**, **nrG**, **sgG**, and **nacG** can have a more complex additional orthogonal structure of the data inside the level that itself may be subject to a granular structure. This, among other topics, will be illustrated in the next section.

Sample usage of the types of granularity in modeling and implementation

We return to several typical examples that passed the revue in section 2, which now can be cast in the light of a type of granularity so as to both be more precise about the way of granulation and to hint toward finer-grained as well as orthogonal properties that can facilitate implementations and stimulate further analysis toward extending the top-level taxonomy of types of granularity in the scale-dependent branch in particular. After analysis typical examples, we sketch a simplification for implementing granularity thanks to the reusability of the types of granularity.

Granular perspectives and their type of granularity. In this section we take a closer look at examples for, primarily, **saoG**, **samG**, and **nrG**, and their modeling options and novel inferences in particular.

Content of a level with arbitrary scales. For granularity type **saoG**, each level is granulated alike a grid with cells that may or may not be exhaustive for its contents. In the remainder of this paragraph we take a closer look at consequences of constraining granulation to be exhaustive versus permitting non-exhaustiveness that was briefly mentioned for the **cG** type at the start of the previous section. Consider a GIS application used for, *e.g.*, laying a grid on a lake as depicted in *Figure 6-A* for a coarser-grained level, which ensures each part of the lake is covered by the grid. Alternatively, one might want to apply a rule alike “when $> 50\%$ of a cell is occupied it must be covered by a grid cell”, shown in *Figure 6-A'* that consequently discards parts of the lake that occupy $< 50\%$ of a cell. This leads to a second question and a consequence: if the discarded cells in *Figure 6-A'* should be taken into account at a lower level. If one does, then one arrives at a granulation as depicted in *Figure 6-B*, if one does not (*Figure 6-B'*), then the parts at the lower level do not make the whole at the coarser-grained level as can be observed from the difference between moving from *A'* to *B'* instead of from *A* to *B* to *B'* (the seven shaded squares would have been absent moving from *A'* to *B'*). The discarding rule means that granulation is not exhaustive for we have thrown out a remainder. This type of impreciseness is characteristic for any coarse-grained level and applies to scales for features such as surfaces, volumes, isotherms, and isobars. Developments in rough set theory, rough mereology, and fuzzy logic might serve as an appealing implementation method, and is further elaborated on in (Chen *et al.*, 2007; Hata & Mukaidono, 1999; Keet, 2007a; Klawonn & Kruse, 2004; Peters *et al.*, 2002; Yao, 2004; Zadeh, 1997). Yet differently, instead of including part of the shore of the lake, one can divide the lake by making the grid *inside* the coloured area only, but then one would have to deal with incomplete cells, that is, cells of different size within one granular level, which complicates computation and would not solve the aforementioned boundary problems. Clearly, this is straightforward for **samG**, thanks to the measurement scale and the explicit requirement for a function to move between entities residing in different levels.

Another type of (non) exhaustiveness occurs with less obvious scales, which was briefly illustrated in *Example 1*. For instance, if it had only (Baby, Child, Adolescent, Adult) based on age, then Elderly is omitted, hence, that the granulation is either non-exhaustive or assumed to be included in Adult and thereby meeting the exhaustiveness criterion. Because we have created the scale, we can easily decide one way or the other. Alternatively, we could have two levels, one

coarse-grained with (Young person, Old person) that partitions a population between persons ≤ 40 year and > 40 years old and a finer-grained one with (Baby, Child, Adolescent, Young adult, Mature adult, Senior, Elderly) using the age brackets 0-5 years, 5-10, 10-20, 20-40, 40-55, 55-70 and > 70 years, respectively. Moreover, bringing the requirement to decide on such issue to the fore, which can be done thanks to the types of granularity, this can be made explicit during the software design phase so as to enhance possibilities for transparent use, reusability, and interoperability of granulated information systems.

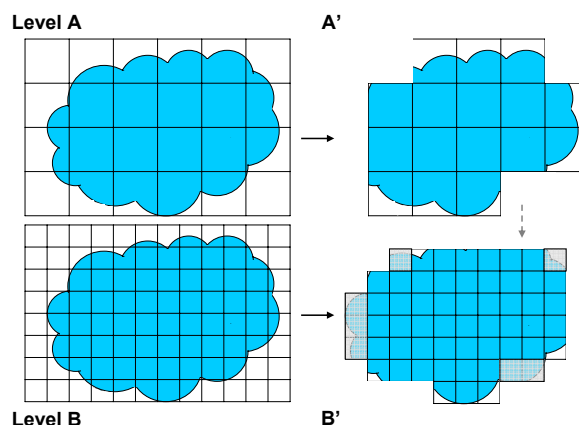


Fig. 6. Grid with cells partitioning a ‘lake’ according to different rules.

Non-scale-dependent content of a level. One might conceptualise non-scale-dependent, qualitative granularity as squeezing in a grid-structure in a level—one entity type, with or without its instances for each ‘cell’ in the grid—but this ignores the relations between parent/children in the hierarchy and would then be a changeable grid and no (pairwise) disjoint tree, in particular for information in biology because disjointness and exhaustiveness is aimed for but rarely achieved⁸. In addition, disjointness depends on the categorisation one is accustomed to, where it may be that the types and their instance satisfy more than one category; *e.g.*, LAB streptococci are types of lactic acid bacteria and types of cocci (sphere-shaped) that aggregate in grape-like bunches. Of greater interest is the interaction between the granulation relations and levels of granularity that adhere to **nfG** or **nrG**. In this setting, granularity provides an additional layer to infer more knowledge than is possible separately. Loading a domain granularity framework with data is—or should be—structure-preserving with respect to the data source so that granularity *enhances* the domain data. Advantages of this approach are illustrated in the next example that uses **nrG**-granulated human anatomy by the parthood relation throughout whilst at the same time exploiting a taxonomic structure within in the granular levels.

Example 4 The Foundational Model of Anatomy (Rosse & Mejino, 2003) uses both *is_a* and *part_of* relations between anatomical entities. Let us take parthood for granulation, then the taxonomic structure can be preserved in the levels, as depicted in *Figure 7-B* for cells. The FMA lists that Blood has as parts: Plasma, Erythrocyte, Neutrophil, Eosinophil, Basophil, Lymphocyte, Monocyte, Platelet, B lymphocyte, T lymphocyte, Natural killer cell, Granular leukocyte, and Leukocyte. Relying on this unordered set alone, one cannot know if it is exhaustive: 1) the list was created manually and some entity type may have been omitted by accident, 2) an ontology adheres to the open-world assumption, and 3) the development tool, FMA-Protégé, does not include axioms for disjoint exhaustive. Combining the taxonomy subsumed by Cell and intersecting it with the parts of Blood, it is immediately evident that both the parent and child types of Non-granular leukocyte are part of blood, but not Non-granular leukocyte itself, even though logically it should be. In addition, two cell types

that are directly subsumed by **Non-granular leukocyte** are **Peripheral blood mononuclear cell** and **Lymphoblast**, but they are not listed as parts of blood. Monocytes are definitely part of blood, whereas lymphoblasts are “immature lymphocytes” and either not non-granular leukocytes or should be subsumed by **Lymphocyte**. Either way, the structure-preserving loading of granular levels brings afore such areas for improvements. Obviously, one would want to take advantage of the already encoded structure of the taxonomy and have returned something alike *Figure 7-B* instead of *A*. \diamond

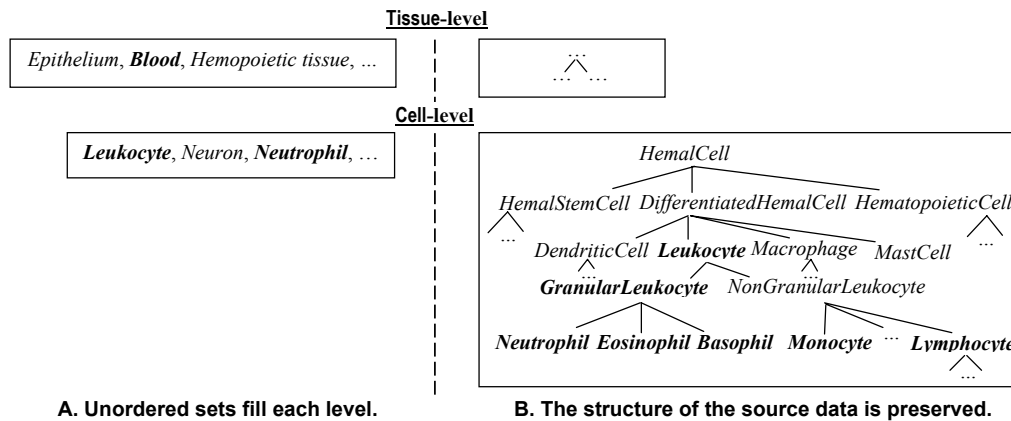


Fig. 7. Two levels with examples of their contents, unordered as in the FMA versus structure-preserving; entities subsumed by **Hemal cell** present a section of the FMA where terms in bold-face are listed (as in A) as part of blood.

This example can be automated at least in part, from which the need has arisen to have a set of usable and reusable types of granular queries (Keet, 2008a). An important advantage of this structure-preserving approach—in particular in conjunction with automation—is that when presenting the combination of taxonomy with granularity demarcations, one gets for free the detection of inconsistent or incomplete knowledge in either the taxonomy or in the partonomy. Thus, conflicting information is highlighted, can be used for formulating research questions, and be investigated.

Example 4, however, implicitly illustrated another issue. Lower levels may contain many more entities—1 human body, 12 organ systems, 300+ cell types, 100000+ proteins (Hunter & Borg, 2003)—which is difficult to understand if these entity types (let alone their instances) were represented as an unordered set for each level. This can be pruned through conditional selections and intersections as done in the example (see also the follow-up examples, below). For instance, if one searches the contents at the **Cell-level** combined with a particular selection of, say, **Blood** at the **Tissue-level**, then the types returned contain only the entity types in the selected levels & type, indicated in bold face in *Figure 7-B*.

A related facet of utilising the structure of the contents compared to an unordered set for each level concerns the ‘size’ aspect, where it is crucial that the relations between the entities at different levels are not destroyed when applying non-scale-dependent granularity to the subject domain. The typical problem it otherwise raises is that of chicken anatomy where the chicken egg at the **Cell-level** is physically larger than some of the chicken’s organs and body parts, such as **Head**. Without the parthood relations between the anatomical entities, one might erroneously assume that cells are part of body parts, hence the chicken egg part of head. Conversely, when one maintains the parthood relation and using **nrG**-granulation, then such inferences cannot be drawn because there is explicitly no path from chicken egg to chicken head. Aside from preventing incorrect inferences, one can obtain more benefits from using the underlying structure, in addition to those already illustrated in *Example 4*, which are demonstrated in the following

example that also takes into account more than one granular perspective with corresponding type of granularity.

Example 5 Whereas *Example 4* looked at the finer-grained levels, here we also consider coarse-grained levels of **Blood**. (1) Represents three levels in the mode of transmission perspective for infectious diseases (Keet & Kumar, 2005), and (2) is taken from the FMA partonomy; thus, **Blood** is positioned at the intersection of two levels in distinct perspectives, being **Mode of transmission** and **Anatomy**, and one can derive (3) from (1) and (2) by traversing the levels ‘up’. (4) Is another branch in the FMA: one branch descends to **Blood** and another one to **Skin-associated lymphoid tissue (SALT)**, and both are ultimately part of the **Hemolymphoid system**.

- 1: Blood involved_in Person-to-person involved_in Direct contact
- 2: Blood part_of Hematopoietic system part_of Hemolymphoid system
- 3: Hemolymphoid system involved_in Direct contact
- 4: Hemolymphoid system has_part Lymphoid System has_part

Non-lymphatic lymphoid system has_part Skin-associated lymphoid tissue

However, one cannot conclude that **SALT** is involved in transmission via **Direct contact**, but it does pose hypotheses on involvement. In fact, **SALT prevents** infectious agents to enter the vascular system (hence, blood). Although the involvement is different, new combinations may be identified and suggest directions for new research.

Traversing the partonomy downwards, one can infer that at least one of **Blood’s** 13 parts must be involved in transmission of infectious agents because blood is. This is already supported by scientific evidence: transmission of hepatitis C virus via **Erythrocytes** (Widell *et al.*, 1996) and West Nile Virus via blood **Plasma** (Hollinger & Kleinman, 2003). Consequently, one may wonder if it is the whole cell or if one can isolate parts of cells that are involved. The latter has been established manually with, *e.g.*, *Listeria* infections at the **Organelle**-level (sub-cellular) and nucleation of actin filament polymerization (Rodal *et al.*, 2005). \diamond

Note that the assumption in the above example implies a reductionist viewpoint, and philosophically encounters the problem of infinite regress; however, the finest-grained level that is defined for the subject domain demarcates a *finite* regress. It is possible that when the implementation predicts involvement of a lower level it either is not known, hence an epistemological issue where the system generates new research questions, or for good scientific reasons involvement of a lower level is not possible due to a systems-level complex combination of events and substances. Either way, using biological granularity in combination with ontologies can speed up the discovery process because it combines existing information—and gaps therein—in a novel way, thereby offering a new view on the same information.

Simplifying implementation by using the types of granularity. In addition to allocation of entities in the appropriate level(s) of granularity, *retrieving* granulated information—just that part of the user’s interest—is important. Of the various types of queries (Keet, 2008a,b), we will take a closer look at one of the requirements: retrieving the contents of a granular level with the *getC* function. This is conceptually straightforward, but hides many details, in particular the need to use the structure of the contents given the different types of granularity to ensure correct behaviour. Let us first define the function as follows. Let gl_i be a particular granular level such that $gl_i \in \mathcal{L}$, with $i \leq n_{max}$, n_{max} the total amount of levels declared, hence, corresponding to the amount of elements in the set \mathcal{L} , and y_1, \dots, y_n are the entity types or instances residing in that particular level, *i.e.* $y_j \in U$ or $y_j \in PT$ with U the set of universals and PT the set of particulars in the granulated information system, and \mathcal{E} denotes the collection of universals or particulars that reside in a single granular level.

Goal: retrieve the contents E , entities (/types), of a selected granular level gl_i ; input is the selected level, where $gl_i \in \mathcal{L}$ (and \mathcal{L} the set of levels) and output is a set of

entities (/types), $E \in \mathcal{E}$, that takes into account the structure of the contents in the level. **Specification:** $getC : \mathcal{L} \mapsto \mathcal{E}$.

For instance, $getC(gl_2) = \{y_1, \dots, y_n\}$. $getC$ takes a particular granular level as argument and returns the contents of that granular level, *irrespective of how the contents themselves may be structured*. Figure 8 depicts this graphically for two levels, reflecting that the structure of the levels' contents varies according to the type of granularity they adhere to. The current characterisation of $getC$, however, does not guarantee preserving the structure of the source data like conveniently depicted in Figure 8. Although this could be ignored here and deferred to the implementation stage, it can be solved relatively easily by nesting other functions specific for each type of granularity. Before we resolve this, salient problems are illustrated if it were ignored.

Example 6 Consider the domain granularity framework for infectious diseases (Keet & Kumar, 2005), we have nine granular perspectives where we focus first on $gp_9 =$ Predisposing factors that has two levels. Retrieving contents of both (1 and 2) ⁹ below, one has to note that (2) is now merely an unordered set without further structure among the entity types in the level, but its simple approach ignores that in the data source the environmental factors are in a different branch of the taxonomy than the four types of living habit predisposing factors; that is, there is not one top type in gp_9gl_2 . Its coarser-grained level gp_9gl_1 contains the entity type **Environment** that in the subject domain subsumes **SocEnv**, **PolEnv**, **EcoEnv**, and **BioEnv**, whereas **Living habits** subsumes **Diet**, **Stress**, **Smoking**, and **PersHyg**.

1: $getC(gp_9gl_1) = \{ \text{LivingHabits, Hereditary, Environment, Age} \}$

2: $getC(gp_9gl_2) = \{ \text{SocEnv, PolEnv, EcoEnv, BioEnv, Diet, Stress, Smoking, PersHyg} \}$

3: $getC(gp_8gl_2) = \{ \text{Congestion, Red hepatization, Grey hepatization, Resolution} \}$

If, on the other hand, we would have selected $gp_8 =$ Pathological process, one of its levels (gp_8gl_2), and retrieve the contents (3), then there is an internal structure among the entities within the level and not only between the types in adjacent levels, for they are successive sub-processes of the **Inflammatory process** of pneumococcal pneumonia. However, neither (2) nor (3) reveals that the former is part of a taxonomy and the latter represent successive processes. \diamond

The problems illustrated in *Example 6* are caused by inadequate usage of the original data. To solve this, we first need to look at the types of granularity and their influence on $getC$. Having recorded the type of granularity used for each granular perspective—in the TOG, this is achieved with the *has_granulation* and *adheres_to* relations—and the structure of the contents of each type of granularity (section 1), we can obtain both the type of granularity and the content structure upon using $getC$. Therefore, the specifics for retrieving the contents of each type of granularity can be solved automatically and has to be defined only *once* for each leaf type of granularity used in the application. Let us use a function, tgL (an abbreviation of *type of granularity that the level adheres to*), which is constrained as $tgL : \mathcal{L} \mapsto TG$, where TG is one of the types of granularity, and \mathcal{L} as before. Given the types of granularity and their corresponding content structure (section 1), their impact on nested functions for $getC$ are as follows.

- ★ For **saoG** we have a grid at each level, hence $getC$ typically will retrieve this grid, which is a 2D representation fixed according to its coordinates. Typically, one wants to retrieve the associated representation of the material entity or its cartographic map, too; hence retrieval with $getC$ will contain at least two sub-functions to handle this.
- ★ **samG** has its instances within a level as an ordered set, and does not need further processing for retrieval.
- ★ **sgpG**: entities with additional data about their size, which can be retrieved, *e.g.*, as a two-column table.
- ★ **sgrG**: textual representation and corresponding figurines, which can be retrieved as with **sgpG** but with two attributes—label of the entity and figurine—for each object.

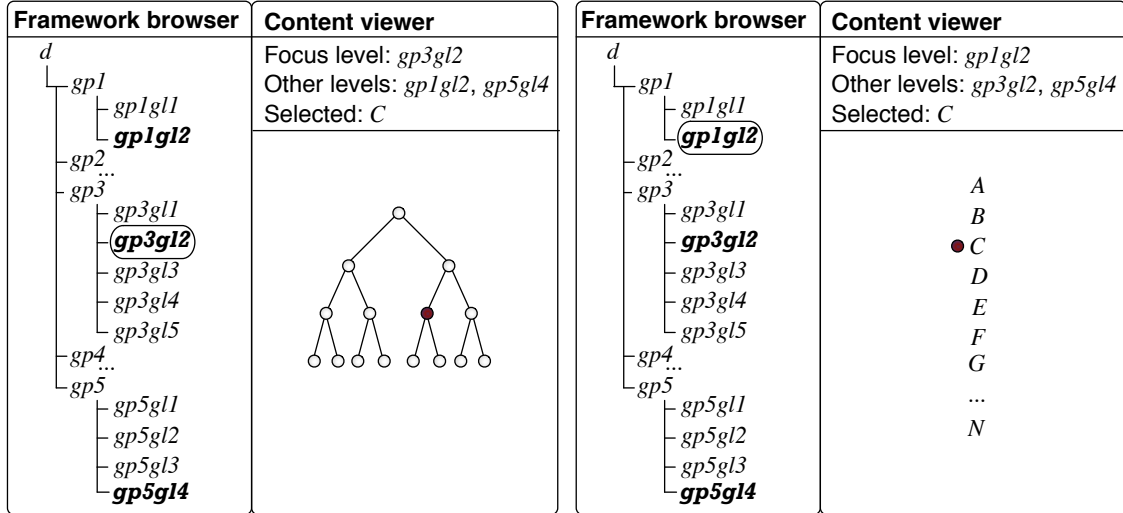


Fig. 8. Left: Selection of levels, with the contents of gp_3gl_2 depicted; Right: Selection of levels, with the contents of gp_1gl_2 depicted; Left: Selection of levels, with the contents of gp_3gl_2 depicted. The black dot indicates that the entity type labelled with C is selected and contained in the hierarchy in gp_3gl_2 and in an ordered list in gp_1gl_2 .

- ★ **nasG** types have unordered sets and do not need further processing for retrieval.
- ★ Depending on the implementation, **nacG** can be an unordered set that is aggregated or have additional ‘subgroups’ in a level. Members at the lower level can be a) aggregated as an unordered set, b) ordered taxonomically, or c) another representation, like a graph with the positions of sports team players on the field. Consequently, finer-grained behaviour of the sub-procedures of $getC$ depends on the data source.
- ★ **nfG**: includes relations among entities within a level, which is useful together with the $getC$ to retrieve all the entities and its relations. (This works only if those entities do not have relations with other entities beyond the level they reside in, else an additional verification is needed that checks that the candidate entity to retrieve is not in another level within the same perspective.) The minimal structure of the representation of the contents are triples with $\langle entity, relation, entity \rangle$, which can be listed as unordered set, or rendered in some graphical representation.
- ★ Using $getC$ with a level adhering to **nrG**-type granulated entities: one may want to take into account their respective supertypes at the adjacent coarser-grained level, and then aggregate the subsumees in the branch into a granule in the focal level. An example of this is the query to retrieve the cells from the **Cell**-level that are part of blood, thereby omitting the other types of cells residing in the **Cell**-level. Hence, $getC$ uses at least a recursive query to retrieve hierarchically organised content and may use entity selection to retrieve a subset of the contents.

Once the content is retrieved, it can be used for further processing, such as intersecting contents of two levels. Regarding *Example 6*, we can ensure retrieval of the sequential processes for hepatization thanks to **nfG**-type of granularity where the procedure requires retrieval of the within-level relations as well, and thanks to **nrG**-type for the predisposing factors granular perspective, use the recursive query with its auxiliary functions, illustrated in *Example 7*.

Example 7 The predisposing factors are of the granularity type **nrG** and the goal is to answer queries such as “given the predisposing factor **Environment** at level gp_9gl_1 , retrieve the contents at level gp_9gl_2 ”; *e.g.*, condensed in (1). If the supertype is unknown beforehand, the query needs a preliminary step to retrieve the parent type of the selected entity; *e.g.*, “retrieve the granule of **Stress**” where **Stress** is subsumed by **Living habit** (2).

- 1: if $\text{grain}(\text{Environment}) = \text{gp}_9\text{gl}_1$ and $\text{grain}(x) = \text{gp}_9\text{gl}_2$
then $\text{getContent}(\text{gp}_9\text{gl}_2) = \{ \text{SocEnv}, \text{PolEnv}, \text{EcoEnv}, \text{BioEnv} \}$ and $\text{is_a}(x, \text{Environment})$
 - 2: if $\text{grain}(\text{Stress}) = \text{gp}_9\text{gl}_2$ and $\text{is_a}(\text{Stress}, x)$ and $\text{grain}(x) = \text{gp}_9\text{gl}_1$ and $\text{grain}(y) = \text{gp}_9\text{gl}_2$
then $\text{getContent}(\text{gp}_9\text{gl}_2) = \{ \text{Diet}, \text{Stress}, \text{Smoking}, \text{PersHyg} \}$ and $\text{subsumes}(x, y)$
- More examples can be found in (Keet, 2008a). \diamond

A suggested procedure that demonstrates the nesting of this function in *getC* is included as *Algorithm 1* (*selectL* is an auxiliary level-selection function). The goal of each **case**-option—*what* to do to retrieve it—is the same but actual operations depend on the software implementation, such as using a recursive query in STRUQL or a method in a C++ program; *i.e.*, its practical realisation depends on how the data, information or knowledge is organised in the type of application. Thus, to achieve the *purpose* of *getC*, one has to *use the type of granularity to which a level adheres* and the finer-grained application-specific procedures it requires to retrieve the content. Clearly, once one has defined the procedure for retrieval, such as the need for recursive query for **nrG**-granulation, one can re-apply this to a different granular perspective that uses the same granulation, thereby avoiding re-analysis and promoting transparency of the software.

Algorithm 1 Retrieving a level’s contents by taking into account the content structure

Require: $x \leftarrow \text{selectL}(x)$

procedure *getC*(x)

- 1: $\phi \leftarrow \text{tgL}(x)$
 - 2: **switch**
 - 3: **case** $\phi = \text{samG}$: $\ll \text{see text for details} \gg$
 - 4: RESULT \leftarrow query and sort the set
 - 5: :
 - 6: **case** $\phi = \text{nrG}$: $\ll \text{see text for details} \gg$
 - 7: RESULT \leftarrow recursive query over granulation relation *GR*
 - 8: **end switch**
 - 9: **return** RESULT
-

FUTURE TRENDS

The top-level taxonomy of types of granularity as proposed in this chapter makes explicit the main distinctions between the different ways how people granulate data, information, and knowledge, and, tentatively, how it may exist in reality. This is, however, only a first step in characterising the foundational semantics of granularity, for which there are two principle directions of further research concerning the topic. First, upon closer inspection, there may be uncovered more detailed distinctions between the types so as to refine the taxonomy with a fourth or even fifth layer. This in particular for the scale-dependent branch, because more theoretical and practical results are known compared to the non-scale-dependent branch (*e.g.*, with an ontology of measurement, usage of attribute values in clustering, rough and fuzzy sets and fuzzy logic). In addition, the interplay between and concurrent use of quantitative with qualitative granularity (Bender & Glen, 2004; Keet, 2007a; Zhou *et al.*, 2005) may reveal additional insights in dependencies between the two principal ways of granulation. Second, there is the need to formalise the taxonomy elegantly with as few primitives as necessary and to result in a logically consistent theory. To this end, it is necessary to have a theory of granularity that adequately addresses, among others, what a granular level is, what the nature of the relation between levels is, and what constitutes a granular perspective (granulation hierarchy). To the best of my knowledge, the most comprehensive proposal that lays bare premises and their (logical) consequences to define a theory of granularity has been proposed by Keet (2008a), which could be used in the

endeavour to formalise the taxonomy, although a comparatively ‘lightweight’ theory might also suffice for scalable information systems.

Both directions, however, focus on more theoretical development as opposed to “*Applications for Advanced Human Reasoning*” as the book title indicates. Clearly, the taxonomy already can be used to better represent granular perspectives *manually*, hence it contributes to *human* reasoning for we have obtained a better understanding of granularity. However, efficient *computational* applications that use the types of granularity—be it *de novo* or as enhancements to existing granulated information systems—have yet to be developed. The latter may be seen as a shortcoming by engineers on the short-term, but a sound theoretical basis is a necessity for each field of study, which has been a noted point for improvement for granular computing already¹⁰. Moreover, the manual examples in section 1 unambiguously demonstrated several of the benefits one can harvest by first identifying the type of granularity, in particular for granular information retrieval. This may provide an incentive to commence novel usage of the types of granularity or to use it to verify existing granular perspectives in, *e.g.*, geographic information systems (Keet, 2009) so as to facilitate information system integration.

CONCLUSIONS

The aim of this chapter was to elucidate foundational semantics of granularity. Ontological distinctions between different types of granularity were identified based on differences in (i) scale- and non-scale-dependent types of granularity; (ii) How levels, and its contents, in a granular perspective relate to each other; (iii) Difference in emphases, being entity-, relation-, or criterion-focused; and (iv) its representation (mathematical or otherwise). Based on the differences uncovered, a top-level taxonomy of types of granularity was developed and it was characterised for each of the eight leaf types of granularity how content (entity types, object, and their relations) residing within levels adhering to a particular type of granularity relate to each other within and across adjacent levels. These types of granularity can guide a conceptual data modeler to better distinguish between the different hierarchies and the software developer to improve on implementations of granularity, in particular when used for reasoning over the data or information. For instance, one can discover implied relations between entities/instances by positioning orthogonally a taxonomy and a partonomy, and make valid inferences with relation to (spatial) inclusion of ecological and/or GIS data. Last, sample contents of a level of granularity were illustrated with examples from several subject domains and advantages of use and reuse of the taxonomy for granular information retrieval was demonstrated.

Current research is focussed on a formalisation of the taxonomy and a case study in the subject domain of agriculture for defining granular perspectives using the types of granularity.

Notes

¹For an excellent recent review paper on rough sets and several extensions, consult Pawlak & Skowron (2007a,b), where the latter also contains an overview of approximation spaces conceptualized by Pawlak in a series of papers from the early 1980s onwards.

²Four-dimensionalism of perdurantists is outside the scope.

³The grid is over the representation of a *material* entity, because one cannot put a grid over the representation of a non-material entity like an organisation, but one can do this with, *e.g.*, a lake—that is, with GIS objects such as representations of entities on cartographic maps.

⁴This does not consider roughness or fuzziness of the measurement, which is an orthogonal issue.

⁵For instance, entity-focussed: i) Different real-world entities in different levels with **sgpG**, **nrG**, **nfG**, **nacG**, ii) The same real-world entities in different levels, reordered/restructured with **sgrG**, **samG**, **saoG**, **nasG**, **nrG**. Human-centred: i) Human-imposed granularity with **sgrG**, **sgpG**, **samG**, **saoG**, **nacG**, **nasG**, and ii) Not necessarily human-imposed granularity with **sgpG**, **nrG**, **nfG**, **nacG**, **nasG**.

⁶“Universals are a class of mind independent entities, usually contrasted with individuals, postulated to ground and explain relations of qualitative identity and resemblance among individuals. Individuals are said to be similar

in virtue of sharing universals.” (MacLeod & Rubenstein, 2005). Philosophically, universals are not always considered to be distinct from concepts (Earl, 2005), but, in practice, the term *concept* tends to refer to mind-dependent entities (Smith, 2004). A *class* is a set, which may correspond to a universal or a concept in its intension.

⁷The use of the word “structure” in the context of contents in granular levels refers to its organisation of the entities (/types), such as an ordered list or grid, which can coincide with structure sensu model theory if the TOG is applied to instance data but its underlying idea will also be used where the contents is type-level knowledge.

⁸This is primarily due to epistemological reasons: there are many things of nature we just do not know enough about, accumulation of knowledge about nature is in flux, and discoveries are not made following a balanced binary tree representation but as they come and where most funding is.

⁹The retrieved entity types are abbreviated here for brevity; see (Keet & Kumar, 2005) for the full terms like **Social environment** and **Personal hygiene**.

¹⁰It has been a noted shortcoming for Granular Computing numerous times by leading researchers like T.Y. Lin in his GrC’06 keynote speech and Yao (2005, 2007, 2008).

Bibliography

- Abelló, A., Samos, J. & Saltor, F. (2006). YAM²: a multidimensional conceptual model extending UML. *Information Systems*, 31(6), 541-567.
- Bargiela, A. & Pedrycz, W. (2006). The roots of granular computing. *IEEE International Conference on Granular Computing 2006 (GrC06)*, 10-12 May 2006, Atlanta, USA. 1, 806-809.
- Bender, A. & Glen, R.C. (2004). Molecular similarity: a key technique in molecular informatics. *Org. Biomol. Chem.*, 2, 3204-3218.
- Bittner, T. & Smith, B. (2003). A Theory of Granular Partitions. In: *Foundations of Geographic Information Science*, Duckham, M, Goodchild, MF, Worboys, MF (eds.), London: Taylor & Francis Books, 2003, pp117-151.
- Bittner, T. & Stell, J. (2003). Stratified rough sets and vagueness, In: *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science. International Conference (COSIT'03)*, Kuhn, W., Worboys, M., Timpf, S. (eds.). pp286-303.
- Broekhoven, E. van, Adriaenssens, V., De Baets, B. (2007). Interpretability-preserving genetic optimization of linguistic terms in fuzzy models for fuzzy ordered classification: an ecological case study. *International Journal of Approximate Reasoning*, 2007, 44: 65-90.
- Camossi, E., Bertolotto, M., Bertino, E., & Guerrini, G. (2003). Issues on Modelling Spatial Granularity. *Workshop on fundamental issues in spatial and geographic ontologies*, 23 Sept. 2003. Ittingen, Switzerland.
- Cariani, P. (1997). Emergence of new signal-primitives in neural systems. *Intellectica*, 25, 95-143.
- Chen, D., Wang, X. & Zhao, S. (2007). Attribute Reduction Based on Fuzzy Rough Sets. In: *Proceedings of the International Conference Rough Sets and Intelligent Systems Paradigms (RSEISP 2007)*, Kryszkiewicz, M., Peters, J.F., Rybinski, H., Skowron, A. (Eds.). Springer, Berlin, Lecture Notes in Artificial Intelligence 4585, 381-390.
- Chen, Y.H. & Yao, Y.Y. (2006). Multiview intelligent data analysis based on granular computing. *IEEE International Conference on Granular Computing (GrC'06)*. IEEE Computer Society, 2006, 281-286.
- Dawyndt, P., De Meyer, H., De Baets, B. (2006). UPGMA clustering revisited: A weight-driven approach to transitive approximation. *International Journal of Approximate Reasoning*, 2006, 42(3): 174-191.
- Degtyarenko, K. & Contrino, S. (2004). COMe: the ontology of bioinorganic proteins. *BMC Structural Biology*, 4, 3.
- Earl, D. (2005). The classical theory of concepts. *Internet Encyclopedia of Philosophy*. <http://www.iep.utm.edu/c/concepts.htm>.
- Edmonds, B. (2000). Complexity and Scientific Modelling. *Foundations of Science*, 5(3), 379-390.
- Elmasri, R., Fu, J., & Ji, F. (2007). Multi-level conceptual modeling for biomedical data and ontologies integration. *20th IEEE International Symposium on Computer-Based Medical Systems (CBMS'07)*. June 20-22, 2007, Maribor, Slovenia. pp589-594.
- Euzenat, J., & Montanari, A. (2005). Time granularity. In: *Handbook of temporal reasoning in artificial intelligence*, Fisher, M., Gabbay, D., Vila, L. (Eds.). Amsterdam: Elsevier 2005. pp59-118.
- Fagin, R., Guha, R., Kumar, R., Novak, J., Sivakumar, D., & Tomkins, A. (2005). Multi-Structural Databases. *PODS 2005*. Baltimore, Maryland, June 13-16, 2005.
- Fent, I. de, Gubiani, D., & Montanari, A. (2005). Granular GeoGraph: a multi-granular conceptual model for spatial data. In: *Proceedings of the 13th Italian Symposium on Advanced Databases (SEBD'05)*. Calí, A., Calvanese, D., Franconi, E., Lenzerini, M., Tanca, L. (eds). Roma: Aracne editrice, 2005, pp368-379.

- Fonseca, F., Egenhofer, M., Davis, C., & Camara, G. (2002). Semantic Granularity in Ontology-Driven Geographic Information Systems. *Annals of Mathematics and Artificial Intelligence*, 36(1-2), 121-151.
- Grizzi, F. & Chiriva-Internati, M. (2005). The complexity of anatomical systems. *Theoretical Biology and Medical Modelling*, 2, 26.
- Hata, Y. & Mukaidono, M. (1999). On Some Classes of Fuzzy Information Granularity and Their Representations. In: *Proceedings of the Twenty Ninth IEEE International Symposium on Multiple-Valued Logic* May 20 - 22, 1999, Freiburg im Breisgau, Germany. pp288-293.
- Hedman, S. (2004). *A first course in logic—an introduction to model theory, proof theory, computability, and complexity*. Oxford: Oxford University Press.
- Hobbs, J.R. (1985). Granularity. *International Joint Conference on Artificial Intelligence (IJCAI85)*, 1985, 432-435.
- Hollinger, F.B. & Kleinman, S. (2003). Transfusion transmission of West Nile virus: a merging of historical and contemporary perspectives. *Transfusion*, 43(8), 992-997.
- Hunter, P.J. & Borg, T. (2003). Integration from Proteins to Organs: The Physiome Project. *Nature*, 4(3), 237-243.
- Johansson, I. (2004b). The Ontology of temperature. *Philosophical Communications*, 32, 115-124.
- Kamble, A.S. (2004). *A Data Warehouse Conceptual Data Model for Multidimensional Information*. PhD thesis, University of Manchester, UK. 2004.
- Kaplan, N., Sasson, O., Inbar, U., Friedlich, M., Fromer, M., Fleischer, H., Portugaly, E., Linial, N., Linial, M. (2005). ProtoNet 4.0: A hierarchical classification of one million protein sequences. *Nucleic Acids Research*, 2005, **33**: D216-D218.
- Keet, C.M. (2006a). A taxonomy of types of granularity. *IEEE Conference in Granular Computing (GrC2006)*, 10-12 May 2006, Atlanta, USA. IEEE Computer Society, 1, 106-111.
- Keet, C.M. (2007a). Granulation with indistinguishability, equivalence or similarity. *IEEE International Conference on Granular Computing (GrC2007)*, San Francisco, November 2-4, 2007. IEEE Computer Society, 11-16.
- Keet, C.M. (2007b). Enhancing comprehension of ontologies and conceptual models through abstractions. *10th Congress of the Italian Association for Artificial Intelligence (AI*IA 2007)*, Rome, September 10-13, 2007. Basili, R., Paziienza, M.T. (Eds.), Springer-Verlag, Lecture Notes in Artificial Intelligence 4733, 814-822.
- Keet, C.M. (2008a). *A Formal Theory of Granularity*. PhD Thesis, KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy.
- Keet, C.M. (2008b). Toward cross-granular querying over modularized ontologies. *International Workshop on Ontologies: Reasoning and Modularity (WORM'08)*. 2 June 2008, Tenerife, Spain. CEUR-WS Vol-348, 6-17.
- Keet, C.M. (2009). Structuring GIS information with types of granularity: a case study. *Proceedings of the 6th International Conference on Geomatics*. La Habana, Cuba, Feb 10-12, 2009.
- Keet, C.M. & Artale, A. (2008). Representing and Reasoning over a Taxonomy of Part-Whole Relations. *Applied Ontology*, 3(1-2), 91-110.
- Keet, C.M. & Kumar, A. (2005). Applying partitions to infectious diseases. *XIX International Congress of the European Federation for Medical Informatics (MIE2005)*, 28-31 August 2005, Geneva, Switzerland. 2005. In: Connecting Medical Informatics and bio-informatics, Engelbrecht, R., Geissbuhler, A., Lovis, C. Mihalas, G. (eds.). Amsterdam: IOS Press. pp1236-1241.
- Kiriayama, T. & Tomiyama, T. (1993). Reasoning about Models across Multiple Ontologies. *International Qualitative Reasoning Workshop* Washington, May 16-20, 1993.
- Klawonn, F. & Kruse, R. (2004). The Inherent Indistinguishability in Fuzzy Systems. In: *Logic versus Approximation: Essays Dedicated to Michael M. Richter on the Occasion of his 65th Birthday*. Lenski, W. (ed.). Springer: Berlin / Heidelberg, Lecture Notes in Computer Science

3075, 6-17.

- Kumar, A., Smith, B., & Novotny, D.D. (2005). Biomedical Informatics and Granularity. *Comparative and Functional Genomics*, 5(6-7), 501-508.
- Lin, T.Y. (2006). Toward a Theory of Granular Computing. *IEEE International Conference on Granular Computing (GrC06)*. 10-12 May 2006, Atlanta, USA. IEEE Computer Society.
- Luján-Mora, S., Trujillo, J., & Song, I. (2006). A UML profile for multidimensional modeling in data warehouses. *Data & Knowledge Engineering*, 59(3), 725-769.
- MacLeod, M.C. & Rubenstein, E.M. (2005). Universals. *The Internet Encyclopedia of Philosophy*. 2005. <http://www.iep.utm.edu/u/universa.htm>.
- Malinowski, E. & Zimányi, E. (2006). Hierarchies in a multidimensional model: From conceptual modeling to logical representation. *Data & Knowledge Engineering*, 59(2), 348-377.
- Mani, I. (1998). A theory of granularity and its application to problems of polysemy and underspecification of meaning. In: *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixth International Conference (KR98)*, A.G. Cohn, L.K. Schubert, and S.C. Shapiro (eds.). San Mateo: Morgan Kaufmann. 1998. pp245-255.
- Masolo, C., Borgo, S., Gangemi, A., Guarino, N., & Oltramari, A. (2003). *Ontology Library*. WonderWeb Deliverable D18 (ver. 1.0, 31-12-2003). <http://wonderweb.semanticweb.org>. 2003.
- Mencar, C., Castellano, G., & Fanellia, A.M. (2007). Distinguishability quantification of fuzzy sets. *Information Sciences*, 177(1), 130-149.
- Mendel, J., Dongrui Wu, D. (2007). Perceptual Reasoning: A New Computing With Words Engine. *IEEE International Conference on Granular Computing (GrC2007)*, San Francisco, November 2-4, 2007. IEEE Computer Society, 446-451.
- Ning, P., Wang, X.S., & Jajodia, S. (2002). An Algebraic Representation of Calendars. *Annals of Mathematics and Artificial Intelligence*, 63(1-2), 5-38.
- Pandurang Nayak, P. & Levy, A.Y. (1995). A semantic theory of abstractions. In: *Proceedings of the International Joint Conference on Artificial Intelligence*, Mellish, C. (ed.). San Mateo: Morgan Kaufmann, 1995. pp196-203.
- Parent, C., Spaccapietra, S., & Zimányi, E. (2006a). Conceptual modeling for traditional and spatio-temporal applications—the MADS approach. Berlin Heidelberg: Springer Verlag. 2006.
- Pawlak, Z. & Skowron, A. (2007a). Rudiments of rough sets. *Information Sciences*, 177(1), 3-27.
- Pawlak, Z. & A. Skowron. (2007b). Rough sets: Some extensions. *Information Sciences*, 177(1), 28-40.
- Peters, J.F., Skowron, A., Ramanna, S., & Synak, P. (2002). Rough sets and information granulation. In: T.B. Bilgic, D. Baets, and O. Kaynak (eds.), *Proceedings of 10th International Fuzzy Systems Association World Congress*, Springer-Verlag, Lecture Notes in Artificial Intelligence 2715, 370-377.
- Polkowski, L. (2006). Rough Mereological Reasoning in Rough Set Theory: Recent Results and Problems. *Proceedings of Rough Sets and Knowledge Technology (RSKT 2006)*, Springer LNCS vol. 4062, 79-92.
- Polkowski, L., Semeniuk-Polkowska, M. (2008). Reasoning about Concepts by Rough Mereological Logics. *Proceedings of Rough Sets and Knowledge Technology (RSKT 2008)*, Springer LNCS vol 5009, 205-212.
- Pontow, C. & Schubert, R. (2006). A mathematical analysis of theories of parthood. *Data & Knowledge Engineering*, 59, 107-138.
- Qiu, T., Chen, X., Liu, Q., Huang, H. (2007). A Granular Space Model For Ontology Learning. *IEEE International Conference on Granular Computing (GrC2007)*, San Francisco, November 2-4, 2007. IEEE Computer Society, 61-65.
- Ribba, B., Colin, T., & Schnell, S. (2006). A multiscale mathematical model of cancer, and its use in analyzing irradiation therapies. *Theoretical Biology and Medical Modelling*, 3, 7.
- Rodal, A.A., Sokolova, O., Robins, D.B., Daugherty, K.M., Hippenmeyer, S., Riezman, H., Grigorieff, N., & Goode, B.L. (2005). Conformational changes in the Arp2/3 complex leading to

- actin nucleation. *Nature Structural & Molecular Biology*, 12, 26-31.
- Rosse, C. & Mejino, J.L.V. (2003). A reference ontology for biomedical informatics: the foundational model of anatomy. *Journal of Biomedical Informatics*, 36, 478-500.
- Salthe, S.N. (1985). *Evolving hierarchical systems—their structure and representation*. New York: Columbia University Press. 1985. 343p.
- Salthe, S.N. (2001). Summary of the Principles of Hierarchy Theory. November 2001. http://www.nbi.dk/~natphil/salthe/hierarchy_th.html (accessed: 10-10-2005).
- Skowron, A. & Peters, J.F. (2003). Rough sets: Trends and challenges - plenary paper. In: Wang, G., Liu, Q., Yao, Y., Skowron, A. (eds.), *Proceedings of RSFDGrC 2003: Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*. Springer-Verlag, Lecture Notes in Artificial Intelligence 2639, 25-34.
- Smith, B. (2004). Beyond Concepts, or: Ontology as Reality Representation. Varzi, A., Vieu, L. (eds.), *Formal Ontology and Information Systems. Proceedings of the Third International Conference (FOIS 2004)*, Amsterdam: IOS Press, 2004, 73-84.
- Sontag, E.D. (2004). Some new directions in control theory inspired by systems biology. *Systems Biology*, 2004, 1(1): 9-18.
- Sowa, J.F. (2000). *Knowledge representation: logical, philosophical, and computational foundations*. China Machine Press. 2000. 594p.
- Straccia, U. (2006). A Fuzzy Description Logic for the Semantic Web. In: *Capturing Intelligence: Fuzzy Logic and the Semantic Web*, Sanchez, E., ed., Elsevier, 2006.
- Tange, H.J., Schouten, H.C, Kester, A.D.M., & Hasman, A. (1998). The Granularity of Medical Narratives and Its Effect on the Speed and Completeness of Information Retrieval. *Journal of the American Medical Informatics Association*, 5(6), 571-582.
- Tsumoto, S. (2007). Mining Diagnostic Taxonomy and Diagnostic Rules for Multi-Stage Medical Diagnosis from Hospital Clinical Data. *IEEE International Conference on Granular Computing (GrC2007)*, IEEE Computer Society, 2007, 611-616.
- Varzi, A.C. (2004). Mereology. *The Stanford Encyclopedia of Philosophy*. (Fall 2004 Edition), Zalta, E.N. (ed.). <http://plato.stanford.edu/archives/fall2004/entries/mereology/>.
- Varzi, A.C. (2007). Spatial reasoning and ontology: parts, wholes, and locations. In: Aiello, M., Pratt-Hartmann, I. & van Benthem, J. (eds.) *Handbook of Spatial Logics*. Berlin: Springer, 945-1038.
- Vernieuwe, H., Verhoest, N.E.C., De Baets, B., Hoeben, R., De Troch, F.P. (2007). Cluster-based fuzzy models of groundwater transport. *Advances in Water Resources*, 2007, 30(4): 701-714.
- Widell, A., Elmud, H., Persson, M.H., & Jonsson, M. (1996). Transmission of hepatitis C via both erythrocyte and platelet transfusions from a single donor in serological window-phase of hepatitis C. *Vox Sang*, 71(1), 55-7.
- Wimsatt, W.C. (1995). The ontology of complex systems: Levels of Organization, Perspectives, and Causal Thickets. *Canadian Journal of Philosophy*, 20, 207-274.
- Yao, J.T. (2007). A ten-year review of granular computing. *IEEE International Conference on Granular Computing 2007 (GrC'07)*, IEEE Computer Society, 2007, 734-739.
- Yao, Y.Y. (2004). A partition model of granular computing. *Lecture Notes in Computer Science Transactions on Rough Sets*, 1, 232-253.
- Yao, Y.Y. (2005). Perspectives of Granular Computing. *IEEE Conference on Granular Computing (GrC2005)*, 1, 85-90.
- Yao, Y.Y. (2007). The art of granular computing. In: *Proceedings of the International Conference on Rough Sets and Emerging Intelligent Systems Paradigms*, 2007.
- Yao, Y.Y. (2008). Granular Computing: Past, Present, and Future. *IEEE Conference on Granular Computing 2008 (GrC'08)*, China, August 2008. IEEE Computer Society (in press).
- Zadeh, L.A. (1997). Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems*, 90(2), 111-127.

- Zadeh, L.A. (2002). From computing with numbers to computing with words—from manipulations of measurements to manipulation of perceptions. *International Journal of applied mathematics and computer science*, 2002, **12**(3): 307-324.
- Zhang, J., Silvescu, A., & Honavar, V. (2002). *Ontology-Driven Induction of Decision Trees at Multiple Levels of Abstraction*. Technical Report ISU-CS-TR 02-13, Computer Science, Iowa State University. 2002. <http://archives.cs.iastate.edu/documents/disk0/00/00/02/91/>.
- Zhou, Y., Young, J.A., Santrosyan, A., Chen, K., Yan, S.F., & Winzeler, E.A. (2005). In silico gene function prediction using ontology-based pattern identification. *Bioinformatics*, *21*(7), 1237-1245.