FREIE UNIVERSITÄT BOZEN
LIBERA UNIVERSITÀ DI BOLZANO
FREE UNIVERSITY OF BOZEN - BOLZANO

KRDB RESEARCH CENTRE
KNOWLEDGE REPRESENTATION MEETS DATABASES

Faculty of Computer Science, Free University of Bozen-Bolzano, Piazza Domenicani 3, 39100 Bolzano, Italy
Tel: +39 04710 16000, fax: +39 04710 16009, http://www.inf.unibz.it/krdb/

KRDB Research Centre Technical Report:

A formal approach for using granularity in the subject domain of infectious diseases

C. Maria Keet

Affiliation	KRDB Research Centre, Faculty of Computer Science Free University of Bozen-Bolzano Piazza Domenicani 3, 39100 Bolzano, Italy
Corresponding author	Maria Keet keet@inf.unibz.it
Keywords	Granularity, infectious diseases, bio-ontologies
Number	KRDB06-2
Date	26 April 2006
URL	http://www.inf.unibz.it/krdb/pub/

A formal approach for using granularity in the subject domain of infectious diseases

C. Maria Keet

KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
keet@inf.unibz.it

Abstract. The aim of the experiment is to put the domain- and implementation-independent theory of granularity to the test with the subject domain of human infectious diseases. After determining the data sources, defining the model, and data manipulation operators, the granular perspectives and their levels were defined and contents added. Subsequently, granular information retrieval is tested for cholera and blood, both regarding querying and inferencing. Observations are that the limitations of data sources complicates applying a domain granularity framework, but developing a domain granularity framework is possible and does not violate the domain- and implementation-independent theory of granularity. Reasoning over the applied domain granularity framework does enable targeted searches and inferencing for advanced knowledge management and information retrieval.

1 Introduction

Granular perspectives and their levels of granularity can be a useful modelling approach to deal adequately with management of the huge amount of biological data and information, concerning the structure to store the information and support for reasoning to verify existing information, deduce new information automatically, and generate research hypotheses[1]. I use a formal approach of granularity and apply it to the subject domain of infectious diseases, extending and improving on the mainly informal bottom-up approach taken earlier (reported on in [9]). First, assumptions, model theoretic considerations and necessary extensions to the in [9] introduced formalisation of granularity are given. Second, the formal characterisation of granular perspectives and levels in the subject domain of infectious diseases is declared, which is followed by some example queries for data retrieval about blood and cholera. Last, I discuss strengths and weaknesses of hte experiment and outline further research.

2 Assumptions, methodology, and data sources

2.1 Assumptions and data sources

- ⋆ The entities taken from the data sources, $\mathcal{DS}$, have been defined in their respective sources.
- ⋆ A reasoner for checking consistency with the Theory Of Granularity (TOG)[2] is at our disposal (at present, this is done manually).

[1] More precisely, the latter is in fact a 'find useful errors that one can use for wet-lab experimentation or through which one has to re-analyse a (small part of) a theory'.

[2] i.e., consistent with the formalisation included in Chapter 3 of the TOG report.

- ⋆ Implementation occurs in a tool like a DL Database or Datalog (both support recursion and deductive reasoning).
- ⋆ The data sources involved to populate the granularity framework are:
  i. The Foundational Model of Anatomy [23] for the perspectives *SiteOfEntry* and *SiteOfEffect* and for other inferencing on anatomical structures.
  ii. SNOMED CT [30] and ICD10 [25] for *DiseaseClassification*.
  iii. The species taxonomy of the Tree of Life [12] or the NCBI taxonomy [27] for the *Phylogeny* of infectious organisms.
  iv. The remainder of granular perspectives, levels and data, such as the *ModeOfTransmission* and *PredisposingFactors*, are based on a compilation from various data sources: scientific literature ([2], [14], [19], [15], [21], [4], [20]), textbooks ([16], [17]), NCID [28], the Encyclopedic reference of parasitology [13] and Pathology Online [29].

  Data sources i-iii are characteristic of their taxonomic structure and can be used for semi-automated loading of the domain granularity framework, as will be illustrated below.

2.2 Methodology

The procedure to define and apply granularity to the infectious diseases data and information is as follows:

1. Demarcate subject domain – which aspects of infectious diseases to include and which not.
2. Define signature, containing the use of the elements of the granularity framework and the operations one can use to manipulate the data.
3. Identify granular perspectives for the chosen subject domain.
4. Identify granular levels and assign the levels to their appropriate perspective.
5. Load this domain granularity framework with data (or: assign a particular level to the entities).
6. Query and retrieve information.

While carrying out these steps, emphasis will be put on assessing:

- ⋆ The correspondence between the rigid characterisation of granularity and its applicability to an arbitrarily chosen subject domain. For instance, does the domain granularity framework violate the TOG, and if yes: where, why?
- ⋆ The feasibility of implementing granularity; hence, what is lacking at the implementation level to bring it to the level of automation of the procedure and usage of applied domain granularity?

3 Preliminaries and model

Recollecting the TOG framework components, we have points 1-3 and from Kumar et al's [11] approach one could add 4-6. However, because 5 and 6 were defined for the case of one perspective only, we need to replace this with two extensions, 7 and 8, and two additional functions (9 and 10) to enable effective manipulation of the system, prevent inconsistencies in information retrieval, and avoid patchwork.

1. Subject domain D^{sd} and framework D^{fw}, with instances d^{sd} and d^{fw} for the subject domain and domain granularity framework, respectively.
2. Granular perspective GP and its instances in the d^{fw}, denoted with subscripts $gp_i, ..., gp_n$.
3. Granular level GL with its instances unique for a perspective $gp_igl_i, ..., gp_igl_n$.
4. U denotes the set of (biological) universals. In DL terminology, the entities to be allocated in the granular levels are part of the TBox, whereas the granularity framework elements are, ontologically, instances[3].
5. GR as the ordered set of levels of granularity applicable to a "domain". Note that in [11] it it applicable to a domain, but they have only *one* 'domain,' that of anatomical granularity of the human body, which amounts to a *perspective* within the more complex subject domain of infectious diseases. The "ordered set of levels" then corresponds to the linked granular levels in one perspective, related with the $partOf$ relation.
6. $gran(x)$ is the function of U onto GR, that takes a universal as argument and returns the level it belongs to [11]. This function will be replaced with $grains(x)$, explained in point 8 below.
7. Kumar et al's [11] GR "domain" corresponds to a gp_i in the more comprehensive TOG setting, and the TOG d^{fw} to a more comprehensive GR' (both regarding formalisation and scope of the domain), such that GR' equals a d^{fw} that contains > 1 $GP(x)$ and ≥ 2 $GL(x)$ in each perspective.
8. Whereas $gran(x)$ can retrieve only one level at a time, which suffices in a domain where each entity is allocated to one level only, we need a function that can retrieve a set of levels, called $grains(x)$, that retrieves *all* levels the selected entity resides[4]. Let ls_i be the set of levels that $grains$ returns and Ls the set of all levels in the domain granularity framework d^{fw}, where $x \in U$, $ls_i \subset Ls$, and $gp_igl_i \in Ls$, then
$$grains(x) = \{gp_1gl_i, ..., gp_ngl_n\} = ls_i \tag{1}$$
$$grains(x) \rightarrow \exists^{\geq 1} y, z(GL(y) \wedge GP(z) \wedge contains(z, y) \wedge U(x) \wedge isOfLevel(x, y)) \tag{2}$$
where $isOfLevel(x)$ is defined as:
$$\exists x, y(isOfLevel(x, y) \triangleq (U(x) \wedge GL(y) \wedge grain(x) = y)) \tag{3}$$
Although it is strictly not necessary to have a function that returns only one level, because ls_i can be a set containing one element only, for both conceptual clarity and ease of implementation to enforce constraints, I use $grain(x)$ for retrieving a single level. This is useful for declaring that an entity belongs to a particular level and, if desired, can be used together with rules like "if granular perspective x, then retrieve the level of entity y", which only ever returns either one level or an empty set if y is in no level of perspective x. So as not to confuse the functions introduced here with the semantics of [11], this function is named $grain$ instead of $gran$.

[3] with nominalisation, i.e. declaring that each framework instance is a concept with that instance, the framework elements can be declared in the TBox as well.

[4] To clarify, an entity occurs ≤ 1 times in a (level of a) perspective, hence ls_i is a set in mathematical terms, thus never a multiset.

9. I also add the assignment of a level of granularity to an entity, as given for $assignGrainLevel(x, y)$ in (4, 5); it also accepts namespacing (not elaborated on here).
$$\forall x \exists y(assignGrainLevel(x, y)) \quad (4)$$
$$\exists x, y(assignGrainLevel(x, y) \rightarrow (GL(y) \wedge U(x) \wedge isOfLevel(x, y))) \quad (5)$$
To add whole subtrees to a level in one single operation to prevent having to assigning each entity in a taxonomy one at a time, I use $assignGrainLevelMulti(x, y)$ with x as the chosen root entity in the tree tha tis to be loaded into the level and y a particular granular level.
10. To retrieve all entities residing in a level, I use the function $getContent(x) = E$, where $x \in Ls$ and $E \subset U$ such that E is an (un)ordered set that has a more elaborate structure where appliccable (as described in [6]). Alternatively, one can use a DL query, where `level` is the chosen level one wants to retrieve its content, i.e. a particular $\text{gp}_i\text{gl}_i \in Gl$:
`intersection of (restriction (contentOf allValuesFrom(level)))`

Summarizing the previous points, the signature is $\langle \Delta, Gp, Gl, U, E, Ls, F \rangle$, where:

- ⋆ Δ has two instances, d^{fw} (the domain granularity framework) and d^{sd} (the subject domain), i.e. there is an interpretation function from Δ onto its interpretation $\mathcal{I}$ following standard DL conventions[5].
- ⋆ Gp is the set of granular perspectives defined, with $gp_i \in Gp$ and $i \geq 1$.
- ⋆ Gl is the ordered set of granular levels defined for each perspective, with $gp_igl_i \in Gl$ and $i \geq 2$.
- ⋆ U denotes the set of universals.
- ⋆ E denotes the collection (and any further structure with in the collection) of universals that reside in a single granular level, and $E \subset U$.
- ⋆ Ls denotes the set of all granular levels, and $Gl \subset Ls$. The set of granular levels returned upon querying the system is subset of Ls, denoted with ls_i.
- ⋆ F denotes the set of functions:
  - $grain(x) = y$, is the function to retrieve the level a particular entity, x, resides in, where $x \in U$ and $y = gp_igl_i \in Gl$.
  - $grains(x) = \{gp_igl_i, ..., gp_ngl_n\} = ls_i$, is the function to retrieve *all* levels a particular entity, x, resides in where $x \in U$ and $ls_i \subset Ls$.
  - $assignGrainLevel(x, y)$, is the function to allocate one entity x to y, where y is the granular level gp_igl_i.
  - $assignGrainLevelMulti(x, y)$, is the function to allocate the entity x *and all the entities it subsumes* to y, where y is the granular level gp_igl_i.
  - $getContent(x) = \{y_1, ..., y_n\} = E$, where $x = gp_igl_i \in Gl$ and and $y_1, ..., y_n \in U$.

[5] For clarity, I separate the interpretation into two, d^{fw} and d^{sd}, but if one takes the reality to be granular, then $d^{fw} \cup d^{sd} \equiv \Delta^{\mathcal{I}}$

4 Applying granularity to source data

4.1 Defining the perspectives and levels

The informal partitioning of the infectious disease subject domain into nine granular perspectives [9] is shown in Fig.1. In the following sections we formalise, correct, and extend this categorisation.

Dimensions		**Level 1**	**Level 2-3**	
Source	Mode of Transmission	Air-borne, Food-borne, Water-borne, Direct contact	Direct Contact: Person-to-person, Animal-to-person (zoonoses)	Person-to-person: Sexual intercourse, Skin, Blood
			Food-borne: Production, Preservation, Preparation	
Site	Site of entry	Respiratory system, Digestive system	Digestive system: Stomach, Duodenum, Colon	
	Site of effect	Respiratory system, Digestive system	Digestive system: Stomach, Duodenum, Colon	
Infectious organism	Common name	Multi-cellular animal Worms and flukes Arthropods Micro-organism Protozoa Fungi and moulds Bacteria	Worms and flukes: Roundworms, Hookworms, Tapeworms, Threadworms	
			Fungi and moulds: Amoebae, Fungi	
			Bacteria: Gram-negative, Gram-positive, Cocci, Rod, Flagellate	Cocci: Mono, Di, Strepto, Staphylo
	Phylogeny	*Prokaryote* *Eubacteria* *Eukaryote* *Mycota* *Protozoa* *Metazoa* *Trypanosomatidae* *Ancylostomatidae*	*Eubacteria*: *Salmonella* spp., *Aeromonas* spp.	*Salmonella* spp.: *S. enteritidis, S. typhi*
			Mycota: Myxomycetes, Phycomycetes, Eumycetes	
			Trypanosomatidae: Leishmania spp., *Trypanosoma* spp.	*Leishmania* spp.: *L. braziliensis, L. donovani, L. tropica*
			Ancylostomatidae: Ancylostoma duodenale, Necator americanus	
Disease classification	Infectious disease	Infectious disease: Dysentery, Pneumonia, Meningitis	Pneumonia: Lobar pneumonia, Segmental or lobular pneumonia, Bronchopneumonia, Interstitial pneumonia	
Pathology	Mode of action	Toxin-producer, Genetic interference	Toxin-producer: Stimulator, Inhibitor	Inhibitor: Covalent binding of the small subunit of the cholera toxin to the G-protein of the Second Messenger System, Covalent modification by pertussis toxin of inhibitory G_i protein that blocks inhibition of adenylate cyclase of the Second Messenger System
	Path. structure	Marbled parenchyma, White cicatricial tissue	White cicatricial tissue: Dense collagen connective tissue with reduced cell density	
	Path. process	Inflammatory process, Proliferative process	Inflammatory process: Congestion, Red hepatisation, Grey hepatisation, Resolution	Congestion: Serous exudation, Vascular engorgement, Rapid bacterial proliferation
Predisposing factors		Living habits, Hereditary, Environment, Age	Living habits: Diet, Smoking, Stress, Personal hygiene	

Fig. 1. Informal perspectives with some examples for each level (Source: [9], p1237).

Following the methodology given in §2.2 and with the obvious case that $\mathrm{d}^{sd} \equiv HumanInfectiousDiseases$, we construct the d^{fw} first by mapping the nine "partitions" listed in [9] into perspectives:

$\mathrm{gp}_1 = ModeOfTransmission$
$\mathrm{gp}_2 = SiteOfEntry$
$\mathrm{gp}_3 = SiteOfEffect$
$\mathrm{gp}_4 = Phylogeny$
$\mathrm{gp}_5 = DiseaseClassifcation$
$\mathrm{gp}_6 = ModeOfAction$
$\mathrm{gp}_7 = PathologicalStructure$
$\mathrm{gp}_8 = PathologicalProcess$
$\mathrm{gp}_9 = PredisposingFactors$

Then, for each perspective the levels can be defined in the following way:
$instantiate(\mathrm{gp}_1\mathrm{gl}_1, GL)$
$contains(\mathrm{gp}_1, \mathrm{gp}_1\mathrm{gl}_1)$
With the TBox statement for the *contains* relation as

$$GP \sqsubseteq \geq 2\ contains.GL \tag{6}$$

we can represent this in DL ABox statements like:
$\mathrm{gp}_1\mathrm{gl}_1 : GL$
$\langle \mathrm{gp}_1,\ \mathrm{gp}_1\mathrm{gl}_1 \rangle : contains$
For non-logicians, an informal representation of the second and third perspectives and their levels with some examples is shown in Table 1. The table name corresponds to gp_i and the values in each row of the first column the levels of gp_i. For usability, this can be mapped to a particular name for each level, included in the second column. Alternatively, everything can be entirely encoded for a DL System. Both options are shown for the *SiteOfEntry* below, for the others only the DL version is given. *SiteOfEffect* has the same level definitions as *SiteOfEntry*, therefore is omitted. To show additional possibilities, labels for the levels and some indicative examples of human anatomy entities are included in second and third column of the table. The levels for anatomical structure are not uncontroversial (cf. FMA with [5] [3]), but it is outside the scope to discuss this in detail; here, I use a condensed FMA [23] partonomy and *not* the 12 (inconsistent) levels of [11]; also the collapsing together of three levels into one ($\mathrm{gp}_2\mathrm{gl}_2$) is not uncontroversial, but used as a simplification[6]. The levels of the

[6] It is outside the scope of this technical report to go into details of ontological soundness of the levels and its contents, and of the notion of granularity in a granularity hierarchy. First, the ontological (un)soundness of the presented level definition, naming, and allocation of the entities in their respective levels. For instance, the FMA contains *Hormone* as structural entity, but it is a functional one (structurally, it can be a peptide like insulin). Second, the granularity in a granularity hierarchy to 'skip' levels where one may identify in reality more levels than used in the software system, like $\mathrm{gp}_2\mathrm{gl}_2$ collapses three levels into one. Others [18] [22] achieved better performance with software implementation using wider and shallower levels of granularity than with fewer perspectives that had more detailed levels; this is a topic of future research.

phylogeny perspective (gp_4) is condensed as well (see also §4.2).

$\langle gp_1,\ gp_1gl_1\rangle : contains$
$\langle gp_1,\ gp_1gl_2\rangle : contains$
$\langle gp_1,\ gp_1gl_3\rangle : contains$

Table 1. gp_2

`SiteOfEntryLevels`	`Name`	`Examples`
gp_2gl_1	Body	*MaleBody*
gp_2gl_2	Principal body part Subdivision of principal body part Organ system	*Head*, *Limb* *LimbGirdle*, *Face* *RespiratorySystem*
gp_2gl_3	Organ	*SalivaryGland*, *Pancreas*
gp_2gl_4	OrganPart	*Tendon*, *Cortex*, *LymphNode*
gp_2gl_5	Tissue	*Epithelium*, *SmoothMuscle*
gp_2gl_6	Tissue part	*HairFollicle*, *Nail*
gp_2gl_7	Cell	*MultiPotentStemCell*, *Melanocyte*
gp_2gl_8	Cell part	*Chromosome*, *Cytoskeleton*
gp_2gl_9	Molecule	*Hormone*, *Protein*, *Melanin*

$\langle gp_2,\ gp_2gl_1\rangle : contains$
$\langle gp_2,\ gp_2gl_2\rangle : contains$
$\langle gp_2,\ gp_2gl_3\rangle : contains$
$\langle gp_2,\ gp_2gl_4\rangle : contains$
$\langle gp_2,\ gp_2gl_5\rangle : contains$
$\langle gp_2,\ gp_2gl_6\rangle : contains$
$\langle gp_2,\ gp_2gl_7\rangle : contains$
$\langle gp_2,\ gp_2gl_8\rangle : contains$
$\langle gp_2,\ gp_2gl_9\rangle : contains$

$\langle gp_4,\ gp_4gl_1\rangle : contains$
$\langle gp_4,\ gp_4gl_2\rangle : contains$
$\langle gp_4,\ gp_4gl_3\rangle : contains$
$\langle gp_4,\ gp_4gl_4\rangle : contains$
$\langle gp_4,\ gp_4gl_5\rangle : contains$
$\langle gp_4,\ gp_4gl_6\rangle : contains$

$\langle gp_5,\ gp_5gl_1\rangle : contains$
$\langle gp_5,\ gp_5gl_2\rangle : contains$
$\langle gp_5,\ gp_5gl_3\rangle : contains$

$\langle gp_6,\ gp_6gl_1\rangle : contains$
$\langle gp_6,\ gp_6gl_2\rangle : contains$

$\langle \mathrm{gp}_6,\ \mathrm{gp}_6\mathrm{gl}_3 \rangle : contains$

$\langle \mathrm{gp}_7,\ \mathrm{gp}_7\mathrm{gl}_1 \rangle : contains$
$\langle \mathrm{gp}_7,\ \mathrm{gp}_7\mathrm{gl}_2 \rangle : contains$

$\langle \mathrm{gp}_8,\ \mathrm{gp}_8\mathrm{gl}_1 \rangle : contains$
$\langle \mathrm{gp}_8,\ \mathrm{gp}_8\mathrm{gl}_2 \rangle : contains$
$\langle \mathrm{gp}_8,\ \mathrm{gp}_8\mathrm{gl}_3 \rangle : contains$

$\langle \mathrm{gp}_9,\ \mathrm{gp}_9\mathrm{gl}_1 \rangle : contains$
$\langle \mathrm{gp}_9,\ \mathrm{gp}_9\mathrm{gl}_2 \rangle : contains$
$\langle \mathrm{gp}_9,\ \mathrm{gp}_9\mathrm{gl}_3 \rangle : contains$
$\langle \mathrm{gp}_9,\ \mathrm{gp}_9\mathrm{gl}_4 \rangle : contains$

4.2 Populating the framework with entities

The perspectives gp_2, gp_3, gp_4, and gp_5 can be done semi-automatically with $assignGrainLevelMulti(x, y)$ because each data source is (also) a taxonomic structure. For example $\mathrm{gp}_2\mathrm{gl}_2$: *all* types of body systems, as represented in the FMA, can be loaded into this level by selecting *BodySystem* in the FMA, set $x \leftarrow BodySystem$ and iterate through its subclasses. With the FMA stored in a database and queryable through the OQAFMA with StruQL, then loading, say, the *Organ*-level occurs, one uses:

```
WHERE
Organ->":NAME"->"Organ",
Organ->":DIRECT-SUBCLASSES"+->OrganSubclasses,
OrganSubclasses->":NAME"->OrganSubclassesName,
    CREATE
    OrganLevel(OrganSubclassesName)
```

For taxonomies, allocating the entities to the levels can occur for each level of depth in the tree as with gp_4 (phylogeny), which results in a content of $\mathrm{gp}_4\mathrm{gl}_2$ as

$$getContent(\mathrm{gp}_4\mathrm{gl}_2) = \{Mycota, Protozoa, Metazoa\}$$

That is, the 'condensed' contents of the first level of the phylogeny dimension in Fig.1 is separated into several levels of the species taxonomy. Ignoring the difficulty of classification, and classification of prokaryotes in particular, a relatively simple levelling is:

$\mathrm{gp}_4\mathrm{gl}_1 = Kingdom$
$\mathrm{gp}_4\mathrm{gl}_2 = Phylum$
$\mathrm{gp}_4\mathrm{gl}_2 = Order \cup Class$
$\mathrm{gp}_4\mathrm{gl}_4 = Family$
$\mathrm{gp}_4\mathrm{gl}_5 = Genus$
$\mathrm{gp}_4\mathrm{gl}_6 = Species$

There are issues with representing properly the separate branches in the tree. For illustrative purposes, the subsets of the contents of finer-grained levels for prokaryotes and eukaryotes are distinguished with a$'$ for the latter. This is an example of a generic problem-to-solve, which will be discussed in §5.

$$
\begin{aligned}
&getContent(\mathrm{gp_4gl_1}) = \{Prokaryote, Eukaryote\}\\
&getContent(\mathrm{gp_4gl_2}) = \{Mycota, Protozoa, Metazoa\}\\
&getContent(\mathrm{gp_4gl'_2}) = \{Eubacteria, Archae\}\\
&getContent(\mathrm{gp_4gl_3}) = \{Gracilicutes, Firmicutes, Tenericutes, Scotobacteria, ...\}\\
&getContent(\mathrm{gp_4gl_4}) = \{Enterobacteriacea, ...\}\\
&getContent(\mathrm{gp_4gl'_4}) = \{Trypanosomatidae, Ancylostomatidae, ...\}\\
&getContent(\mathrm{gp_4gl_5}) = \{Salmonella, Aeromonas, ...\}\\
&getContent(\mathrm{gp_4gl'_5}) = \{Leishmania, Tripanosoma, ...\}\\
&getContent(\mathrm{gp_4gl_6}) = \{Salmonella\ enteritidis, Salmonella\ typhi, ...\}\\
&getContent(\mathrm{gp_4gl'_6}) = \{Leishmania\ braziliensis, Leishmania\ donovani, ...\}
\end{aligned}
$$

The other levels require manual assignment for the time they are not structured in an ontology/taxonomy/partonomy. One example is shown here for the *ModeOfTransmission*, where the rest ($\mathrm{gp_6}$, $\mathrm{gp_7}$, $\mathrm{gp_8}$, and $\mathrm{gp_9}$) follows the same pattern.

$$
\begin{aligned}
&assignGrainLevel(AirBorne, \mathrm{gp_1gl_1})\\
&assignGrainLevel(FoodBorne, \mathrm{gp_1gl_1})\\
&assignGrainLevel(WaterBorne, \mathrm{gp_1gl_1})\\
&assignGrainLevel(DirectContact, \mathrm{gp_1gl_1})
\end{aligned}
$$

$$
\begin{aligned}
&assignGrainLevel(PersonToPerson, \mathrm{gp_1gl_2})\\
&assignGrainLevel(AnimalToPerson, \mathrm{gp_1gl_2})\\
&assignGrainLevel(FoodProduction, \mathrm{gp_1gl_2})\\
&assignGrainLevel(FoodPreservation, \mathrm{gp_1gl_2})\\
&assignGrainLevel(FoodPreparation, \mathrm{gp_1gl_2})
\end{aligned}
$$

$$
\begin{aligned}
&assignGrainLevel(SexualIntercourse, \mathrm{gp_1gl_3})\\
&assignGrainLevel(Skin, \mathrm{gp_1gl_3})\\
&assignGrainLevel(Blood, \mathrm{gp_1gl_3})
\end{aligned}
$$

Alternatively, one can opt for DL roles instead of using a function, like

$$
\begin{aligned}
&AirBorne \sqsubseteq \exists isOfLevel.\mathrm{gp_1gl_1}\\
&PersonToPerson \sqsubseteq \exists isOfLevel.\mathrm{gp_1gl_2}\\
&SexualIntercourse \sqsubseteq \exists isOfLevel.\mathrm{gp_1gl_3}
\end{aligned}
$$

and so forth. In addition to the list of examples in Fig.1 of [9], these entities were added:

$PersonToPerson \sqsubseteq \exists involvedIn.DirectContact$
$AnimalToPerson \sqsubseteq \exists involvedIn.DirectContact$

$SexualIntercourse \sqsubseteq \exists involvedIn.PersonToPerson$
$Skin \sqsubseteq \exists involvedIn.PersonToPerson$
$Blood \sqsubseteq \exists involvedIn.PersonToPerson$

$FoodProduction \sqsubseteq \exists involvedIn.FoodBorne$
$FoodPreservation \sqsubseteq \exists involvedIn.FoodBorne$
$FoodPreparation \sqsubseteq \exists involvedIn.FoodBorne$

As is obvious, such laborious manual work should be avoid where possible. In this case, examples and relations were provided in [9], such that *if first*, the *involvedIn* relations above are added, *then second*, the $assignGrainLevelMulti(x, y)$ can be used, thereby saving 5 operations of the 20 otherwise required for the *ModeOfAction* alone. Alternatively, implicit relations can be found with an automated reasoner such as FaCT or RACER: with the level specification of gp_1 and the relations between the entities as given above, one can deduce (7), i.e. because *Blood* is *involvedIn* the *PersonToPerson* mode of transmission and *PersonToPerson* is in the level $\mathrm{gp}_1\mathrm{gl}_2$, therefore *Blood* resides in the immediate finer-grained level $\mathrm{gp}_1\mathrm{gl}_3$.

$$\begin{aligned}((Blood \sqsubseteq \exists involvedIn.PersonToPerson) \sqcap \\ (PersonToPerson \sqsubseteq \exists isOfLevel.gp_1gl_2)) \rightarrow \\ Blood \sqsubseteq \exists isOfLevel.gp_1gl_3 \end{aligned} \tag{7}$$

Last, some more sample entities and additional levels are defined for the granular perspectives *PathologicalProcess* and *PredisposingFactors*.

$InflammatoryProcess \sqsubseteq \exists involves.Congestion$
$InflammatoryProcess \sqsubseteq \exists involves.RedHepatization$
$InflammatoryProcess \sqsubseteq \exists involves.GreyHepatization$
$InflammatoryProcess \sqsubseteq \exists involves.Resolution$
$Congestion \sqsubseteq \exists involvedIn.InflammatoryProcess$
$RedHepatization \sqsubseteq \exists involvedIn.InflammatoryProcess$
$GreyHepatization \sqsubseteq \exists involvedIn.InflammatoryProcess$
$Resolution \sqsubseteq \exists involvedIn.InflammatoryProcess$

$Congestion \sqsubseteq \exists involves.SerousExudation$
$Congestion \sqsubseteq \exists involves.VascularEngorgment$
$Congestion \sqsubseteq \exists involves.(BacterialProliferation \sqcap \exists hasValue.Rapid)$

$LivingHabit \sqsubseteq PredisposingFactor$
$Genome \sqsubseteq PredisposingFactor$
$Environment \sqsubseteq PredisposingFactor$
$Age \sqsubseteq PredisposingFactor$

Content of gp_9gl_2 on the left-hand side of the "$\sqsubseteq$":

$Diet \sqsubseteq LivingHabit$
$Smoking \sqsubseteq LivingHabit$
$Stress \sqsubseteq LivingHabit$
$PersonalHygiene \sqsubseteq LivingHabit$

$SocialEnvironment \sqsubseteq Environment$
$EconomicEnvironment \sqsubseteq Environment$
$PoliticalEnvironment \sqsubseteq Environment$
$BiologicalEnvironment \sqsubseteq Environment$

Content of gp_9gl_3 on the left-hand side of the "$\sqsubseteq$":

$PopulationSize \sqsubseteq \exists partOf.SocialEnvironment$
$PopulationDensity \sqsubseteq \exists partOf.SocialEnvironment$
$Urbanization \sqsubseteq \exists partOf.SocialEnvironment$
$ChangeableSocialNetwork \sqsubseteq \exists partOf.SocialEnvironment$

$PersistentPoverty \sqsubseteq \exists partOf.EconomicEnvironment$
$PublicHealthSystem \sqsubseteq \exists partOf.EconomicEnvironment$
$DrugRelatedTRIPS \sqsubseteq \exists partOf.EconomicEnvironment$

$PoliticalInstability \sqsubseteq \exists partOf.PoliticalEnvironment$
$ArmedConfict \sqsubseteq \exists partOf.PoliticalEnvironment$
$PoliticalIgnorance \sqsubseteq \exists partOf.PoliticalEnvironment$
$PoliticalDenial \sqsubseteq \exists partOf.PoliticalEnvironment$
$PoliticalObduracy \sqsubseteq \exists partOf.PoliticalEnvironment$

$MixedFarmingFarm \sqsubseteq \exists partOf.BiologicalEnvironment$
$ColonizationZone \sqsubseteq \exists partOf.BiologicalEnvironment$
$EcologicalSite \sqsubseteq \exists partOf.BiologicalEnvironment$

Content of gp_9gl_4 on the left-hand side of the "$\sqsubseteq$":

$ClimateChangeAffectedZone \sqsubseteq \exists partOf.EcologicalSite$
$HabitatDestructionZone \sqsubseteq \exists partOf.EcologicalSite$

It is important to notice that the first eight perspectives have their corresponding levels related through one type of relation between the levels within a perspective, but not gp_9. For the levels in the *PredisposingFactors* perspective, gp_9gl_1 and gp_9gl_2 are related through *isA*, but $gp_9gl_4 \prec gp_9gl_3 \prec gp_9gl_2$. Should one model this as two separate, orthogonally positioned perspectives with two levels each? This makes the d^{fw} consistent with the TOG constraints, ensures transitivity between the levels and thereby facilitate automated reasoning[7], but combining them for the user is more

[7] although the reasoner should be able to 'understand' that if partOf(A, B), and that when isA(B,C) and isA(C, D), that then partOf(A, D) – for gp_9, e.g. that *ArmedConfict* is a part of *Environment*. However, that this inference is semantically correct does not imply it holds always.For certain is,

user-friendly. Of course, in a software application the 'back-end' implementation can deviate from that what is shown in the user interface. An option is to teach the user to model ontologically correct granular levels, but this may not always be doable in practice. Thus, to adhere tot he meta-level constraint of granularity we would introduce a $gp_{10} = Environmental_Predisposing Factors$ and map the bottom three levels of gp_9 into those of gp_{10}, i.e. gp_9gl_2 becomes $gp_{10}gl_1$, gp_9gl_3 becomes $gp_{10}gl_2$, and gp_9gl_4 becomes $gp_{10}gl_3$.

Note that the $partOf$ and $involvedIn$ relations have their semantics defined in the domain-independent TOG formalisation, and its inverse does not hold in all axioms listed above. In a final model theory, this needs to be made explicit. In addition, (non-)disjointness has to be addressed.

4.3 Updating (the contents of) the framework

When new entities are added to an existing populated domain granularity framework, there are two possible situations. First, if it is part of a batch update of the source, then re-running $assignGrainLevelMulti$ or a database view ensures automatic assignment of the new entities to its appropriate level(s). Second, if not already structured in its $\mathcal{DS}$, individual assignment is a feasible strategy similar to a database SQL `UPDATE` to insert a tuple. Alternatively, one can write a trigger that fires upon adding a new entity. For instance, *ListeriaContamination* is related to an already included entity, then the high-level design code applied to this instance can be:

1. involvedIn(ListeriaContamination, FoodProduction)
2. gran(FoodProduction) = gp_1gl_2
3. **if** exists $\text{gp}_1\text{gl}_3 \prec \text{gp}_1\text{gl}_2$
3.1 **then** assignGrainLevel(ListeriaContamination, gp_1gl_3)
3.2 **else** instantiate(gp_1gl_3)
3.2.1 contains(gp_1, gp_1gl_3)
3.2.2 assignGrainLevel(ListeriaContamination, gp_1gl_3)
4. **end**

There are several possible strategies to implement adding a new level to a particular perspective or adding a perspective. The chosen strategy affects the correct functioning of the algorithm depicted with the high-level design code, but it is outside the current scope to go into detail of the options[8].

5 Retrieving information with the applied domain granularity framework

Recollecting the methodology, we have carried out in sequence: defined the domain granularity framework d^{fw} and the nine perspectives, created levels and assigned levels to each perspective. If not already provided by the $\mathcal{DS}$, then taxonomic or partonomic (including $involvedIn$) structure was added to the relevant entities and

that its *inverse* does not hold, as, for instance, it is not the case that *Environment* always has part *ArmedConflict*).

[8] See Chapter 4 of the TOG report for details.

each defined level was populated with entities. There are several options to visualise the knowledge represented in the previous paragraph to enable communication with the domain experts, which will be elaborated on elsewhere.

With all this in place, we can put it to work. This is illustrated for cholera in §5.1, which uses the internal *isA* structures of the relevant granular levels, and for blood's involvement in transmission of infectious agents (§5.2), which mainly utilises granularity with the FMA.

5.1 Cholera

Cholera is a feared disease to emerge after a natural disaster such as excessive flooding. To increase understanding of the what it is and what happens, we have to retrieve a variety of information about its causative agent, the *Vibrio cholerae*. Assuming an ignorant user, this is carried out in several operations, although the system allows 'shortcut' information retrieval of specific detailed aspects as well.

1. First, we retrieve information about *V. cholerae* looking for the levels that contain the *V. cholerae* (8), by using the *grains* function:
$$grains(Vibrio\ cholerae) = \{\mathrm{gp_4gl_6}, \mathrm{gp_2gl_2}, \mathrm{gp_3gl_4}, \mathrm{gp_6gl_1}\} \quad (8)$$
With the levels of granularity it returns, we can elucidate and derive several facts.
2. **derive**:
   - ⋆ The type of agent: $\mathrm{gp_4gl_6} = Species$, which means it is an organism because the phylogeny contains *V. cholerae* . Usning the hierarchy of the phylogeny, we find in the top-most level $\mathrm{gp_4gl_1}$ that *V. cholerae* is a prokaryote, or more precisely (in $\mathrm{gp_4gl_2}$) that it is a bacterium.
   - ⋆ Out of curiosity, its 'sister' causative agents at the *Species* level are causative organisms such as
$$getContent(\mathrm{gp_4gl_6}) = \{Salmonella\ typhi,\ Clostridium\ tetani,\ Vibrio\ cholera,\ ...\} \quad (9)$$
The first two cause typhus and tetanus, respectively. At present, there is no way to derive that it is the causative agent of the disease cholera ($\mathrm{gp_5gl_2}$) because granularity in causality is not (yet) included in the system.
   - ⋆ Then, the site of entry how the bacterium gets into the human body $\mathrm{gp_2}$, with the second level reveals body systems
$$getContent(\mathrm{gp_2gl_2}) = \{DigestiveSystem,\ RespiratorySystem,\ CirculatorySystem,...\} \quad (10)$$
more precisely, the digestive system it relates to.
   - ⋆ To get a scope of the site of effect, we retrieve the contents of $\mathrm{gp_3gl_4}$ with (11) and observe that the site of effect is the intestine.
$$getContent(\mathrm{gp_3gl_4}) = \{Duodenum,\ Colon,\ Liver,\ Kidney,\ ...\} \quad (11)$$
   - ⋆ last, we can retrieve information about $\mathrm{gp_6}$, the mode of action: what does it do?
$$getContent(\mathrm{gp_6gl_1}) = \{ToxinProducer,\ GeneticInterference,\ Vector,\ ...\} \quad (12)$$
To illustrate how one can find that *V. cholera* is a producer of toxins, we proceed to the next point.

3. Traversing the partonomy for *V. Cholera* down to its parts at the *Molecule*-level, we find (13).
$$Vibrio\ cholerae \sqcap \exists hasPart.CholeraToxin \tag{13}$$
4. Subsequently, retrieve all the levels the cholera toxin resides, we perform (14) to find out what the toxin does.
$$grains(CholeraToxin) = \{\mathrm{gp_6gl_2}, \mathrm{gl_2gl_9}, ...\} \tag{14}$$
Analogous to step 2, we like to know more about the (contents of) the levels and their perspectives.
5. Upon retrieving the contents of the levels for the cholera toxin (15, 16)
$$getContent(\mathrm{gp_6gl_2}) = \{Inhibitor,\ Stimulator, ...\} \tag{15}$$
$$getContent(\mathrm{gp_2gl_9}) = \{Vasopressin, PertussisToxin,\ CholeraToxin,\ ...\} \tag{16}$$
and traversing its *isA* hierarchy, then
6. **derive**:
   - ⋆ Knowing that $\mathrm{gp_2gl_9}$ is the *Molecule*-level, then *CholeraToxin* is a molecule from a structural perspective, and going up in the taxonomy,
   $$CholeraToxin \sqsubseteq Protein \tag{17}$$
   it is a protein
   - ⋆ Then, looking at the functional taxonomy as shown in Table 2, or its relevant part in DL (18-20), then the cholera toxin has the function of being an inhibitor.
   $$CholeraToxin \sqsubseteq AB5TypeToxin \tag{18}$$
   $$AB5TypeToxin \sqsubseteq ABTypeToxin \tag{19}$$
   $$ABTypeToxin \sqsubseteq Inhibitor \tag{20}$$
   - ⋆ To find out that it affects the (site of effect) epithelium cells (26) of the intestinal tract ((12) in point 2 above), we have to traverse the taxonomy and partonomy trees twice, visualised in Fig.2, and in DL:
   $$Intestine \sqsubseteq \exists hasPart.WallOfIntestine \tag{21}$$
   $$WallOfIntestine \sqsubseteq \exists hasPart.IntestinalMucosa \tag{22}$$
   $$IntestinalMucosa \sqsubseteq \exists hasPart.IntestinalEpithelium \tag{23}$$
   $$IntestinalEpithelium \sqsubseteq EpitheliumOfOrganPart \tag{24}$$
   $$EpitheliumOfOrganPart \sqsubseteq Epithelium \tag{25}$$
   $$Epithelium \sqsubseteq \exists hasPart.EpithelialCell \tag{26}$$
   $$EpithelialCell \sqsubseteq SomaticCell \tag{27}$$
   $$SomaticCell \sqsubseteq NucleatedCell \tag{28}$$
   $$NucleatedCell \sqsubseteq Cell \tag{29}$$
   How to find the shortest path in an ontology is a separate topic (but not impossible, see e.g. [7]).
7. Going to an even more fine-grained level (not elaborated on here as it follows the same procedure as outlined), then we find out that it affects a cell membrane-bound component of the Second Messenger System. Continuing this procedure further, we also retrieve that the inhibition occurs because of the "covalent binding of the small subunit of the cholera toxin to the G-protein of the Second Messenger System" and so forth. (see Fig.1).

Summarizing, after carrying out points 1-7, we now know about *V. cholera* that it is a bacterium that produces the cholera toxin (that is a type of protein), causes the disease cholera, is contracted through the digestive system and affects the epithelial cells of the intestinal tract and its model of action is to inhibit the Second Messenger System. In addition, other more or less relevant information could be retrieved, like bumping into the pertussis toxin, *S. typhi*, and that the cholera toxin is an AB5 type toxin, among other things.

Entity1	Relation	Entity2
AB type toxin	isA	Inhibitor
Pertussis toxin	isA	AB5 type toxin
AB5 type toxin	isA	AB type toxin
Cholera toxin	isA	AB5 type toxin
Diphtheria toxin	isA	AB5 type toxin
Botulinum toxin	isA	Inhibitor
...	isA	...

Table 2. Examples for the table Inhibitor_level

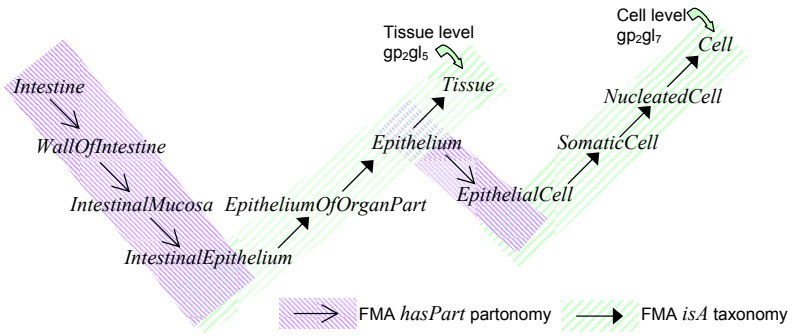

Fig. 2. Visualisation of (21-29) traversing the FMA partonomy and taxonomy.

5.2 Blood

For the sake of example, I use a different strategy here by not focusing on retrieving the contents of levels, but on the relations between the entities that result from residing in different levels: once the extra 'granularity layer' is added to the system, then we also may use the relations it has introduced, as well as exploit these to 'fill gaps' in the relations between entities in the FMA. The topic is to figure out the involvement of blood in the transmission of diseases.

1. Recollecting §4, the *ModeOfTransmission* perspective gp_1, the $\mathcal{KB}$ contains (30, 31) in the second and third granular level.

$$Blood \sqsubseteq \exists involvedIn.PersonToPerson \tag{30}$$
$$PersonToPerson \sqsubseteq \exists involvedIn.DirectContact \tag{31}$$

2. Because we use the partonomy of the FMA, we have (32, 33) to derive that blood is part of the hemolymphoid system.

$$Blood \sqsubseteq \exists partOf.HematopoieticSystem \tag{32}$$
$$HematopoieticSystem \sqsubseteq \exists partOf.HemolymphoidSystem \tag{33}$$

   By traversing each of the two granular perspectives used in point 1 and 2 upwards, we
3. **derive**: that the *HemolymphoidSystem* is involved in transmission of infectious agents via direct contact (*DirectContact* resides in $\mathrm{gp_1gl_1}$), because if a part is involved, then so is the whole.
4. In the other direction, we traverse the partonomy downwards: from the level of *Blood* (34) plus recursively downwards two levels (35) to the *Cell*-level of structural anatomy, the cells that are part of blood can be retrieved with the *getContent* function (36). The procedure and StruQL code to achieve this is described and explained in [7].

$$\text{if } \mathrm{gp_3} \text{ then } grain(Blood) = \mathrm{gp_3gl_5} \tag{34}$$
$$\mathrm{gp_3gl_7} \prec \mathrm{gp_3gl_6} \prec \mathrm{gp_3gl_5} \tag{35}$$
$$getContent(\mathrm{gp_3gl_7}) = \{Leukocyte, Erythrocyte, Basophil,\ ...\} \tag{36}$$

5. These cells are in the answer because of the underlying FMA taxonomy that contains (37-43):

$$Basophil \sqsubseteq GranularLeukocyte \tag{37}$$
$$GranularLeukocyte \sqsubseteq Leukocyte \tag{38}$$
$$Leukocyte \sqsubseteq DifferentiatedHemalCell \tag{39}$$
$$DifferentiatedHemalCell \sqsubseteq HemalCell \tag{40}$$
$$HemalCell \sqsubseteq SomaticCell \tag{41}$$
$$SomaticCell \sqsubseteq NucleatedCell \tag{42}$$
$$NucleatedCell \sqsubseteq Cell \tag{43}$$

   Because blood is involved in transmission of infectious diseases, then at least one of *Blood*'s 41 parts must be involved in transmission of infectious agents. This is already supported by scientific evidence, such as hepatitis C virus for erythrocytes [21] and West Nile Virus for blood plasma [4]. Note that this assumption implies a reductionist viewpoint, and philosophically encounters the problem of infinite regress. It is possible that when the implementation suggests 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: if the latter, the inferred fact requires justification (but see also [8]).
6. Continuing with point 3, that the *HemolymphoidSystem* has something to do with the transmission of infectious agents, does not imply all of its parts do. In fact, it is not only skin-associated lymphoid tissue (44-46), but also, for instance, *Thymus* and *LymphNode* that are part of branches in the partonomy of the *HemolymphoidSystem* (47-50) and all are part of the defence mechanisms of the human body to prevent or combat infection, i.e. part of the *ImmuneSystem* (but see also

point 7 below).

$$HemolymphoidSystem \sqsubseteq \exists hasPart.LymphoidSystem \quad (44)$$

$$LymphoidSystem \sqsubseteq \exists hasPart.NonLymphaticLymphoidSystem \quad (45)$$

$$NonLymphaticLymphoidSystem \sqsubseteq \exists hasPart.SkinAssociatedLymphoidTissue \quad (46)$$

$$HemolymphoidSystem \sqsubseteq \exists hasPart.HematopoieticSystem \quad (47)$$

$$HematopoieticSystem \sqsubseteq \exists hasPart.Thymus \quad (48)$$

$$Hemolymphoidsystem \sqsubseteq \exists hasPart.LymphaticTreeOrgan \quad (49)$$

$$LymphaticTreeOrgan \sqsubseteq \exists hasPart.LymphNode \quad (50)$$

Traversing levels downwards from *HemolymphoidSystem* through another route in the partonomy (such as (44-50)), one cannot conclude that *Skin-associated lymphoid tissue* (SALT) is involved in transmission through *DirectContact* like the *HemolymphoidSystem* is, but does pose hypotheses on involvement. In fact, SALT *prevents* infectious agents to enter the vascular system, hence prevents them from entering blood. Although the involvement is different, new combinations may be identified and suggest directions for new biomedicine research.

7. However, the FMA has an 'empty' and undefined *ImmuneSystem*, other than being subsumed by a "set of heterogeneous clusters". (51-53) are not in the FMA but my additions, and other partonomic relations could be added as parts, such as the immunoglobulins, macrophages etc. that are already defined in the FMA; having the granularity structure can simplify filling such gaps, in no small part because by *using* the ontology in a different way than it was designed for (i.e. querying for information instead of browsing), it reveals lacunas quicker.

$$SkinAssociatedLymphoidTissue \sqsubseteq \exists partOf.ImmuneSystem \quad (51)$$

$$Thymus \sqsubseteq \exists partOf.ImmuneSystem \quad (52)$$

$$LymphNode \sqsubseteq \exists partOf.ImmuneSystem \quad (53)$$

8. If we want to know the anatomical level of the defence mechanism of the thymus and lymph nodes, we can either retrieve this information using the *grain* function in conjunction with a rule (54, 55), execute a DL query (56, 57), or directly move up the *isA* hierarchy (58-63).

$$\text{if } \mathrm{gp}_3 \text{ then } grain(Thymus) = \mathrm{gp}_3\mathrm{gl}_3 \quad (54)$$

$$\text{if } \mathrm{gp}_3 \text{ then } grain(LymphNode) = \mathrm{gp}_3\mathrm{gl}_4 \quad (55)$$

$$\text{Level } x\ (\forall containedIn.\mathrm{gp}_3 \sqsubseteq \exists isGrain.Thymus) \quad (56)$$

$$\text{Level } x\ (\forall containedIn.\mathrm{gp}_3 \sqsubseteq \exists isGrain.LymphNode) \quad (57)$$

$$Thymus \sqsubseteq CorticomedullaryOrgan \quad (58)$$

$$CorticumedullaryOrgan \sqsubseteq ParenchymatousOrgan \quad (59)$$

$$ParenchymatousOrgan \sqsubseteq SolidOrgan \quad (60)$$

$$SolidOrgan \sqsubseteq Organ \quad (61)$$

$$LymphNode \sqsubseteq OrganComponent \quad (62)$$

$$OrganComponent \sqsubseteq OrganPart \quad (63)$$

9. We can continue this relatively random exploration in any direction that piques a user's interest, such as retrieving information about the viruses, function of blood, components and location of the thymus, etc.

6 Discussion

6.1 Omitting details

The relatively simple illustrations of data retrieval with cholera and blood do not fully address its underlying complexities: from a user perspective, this is as intended. However, it is important to address aspects that are far from unimportant for implementation of the system. Concerning the subject domain, some of the perspectives had readily loadable data (see §2), whereas for others a basic structure was already manually created [9], thereby speeding up the loading of entities into their respective levels with $assignGrainLevelMulti(x, y)$. However, at present, this function relies on recursive queries and has not been adequately implemented in systems tat store ontologies [7]. Put differently, many assignments were carried out manually, which is a laborious task. Domain experts eventually are the users who will have to carry out this task, of whom it cannot be expected to code it in DL (or any other logic representation), although a graphical language and an easy to use user interface will ease this task. In addition, implementing this task at least semi-automatically is a prerequisite for successful adoption of applied domain granularity.

A substantial amount of data of the subject domain was seemingly 'ignored' in the retrieval phase, while this is – or assumed to be – present in the system. For instance, we could easily zoom in into the desired information relevant to cholera, but what is *intentionally* ignored? In this respect it is difficult to show the added value of granularity in information management, because it is exactly the hidden complexities that a user otherwise has to put up with. We could have carried out a retrieval operation on the topics of *Bordetella pertussis* as the causative agent of whooping cough, or the effects of caffeine, or morphine, or the docking mechanism of the cholera toxin to the cell-bound receptor, or the ancient hunger signal of the cellular slime mould (social amoeba) *Dictyostelium discoideum* before it forms a pseudoplasmodium, etc. Each one affects the Second Messenger System distinctly in different types of cells in different organs, in different types of organisms, and involving different parts of the Second Messenger Systems: understanding the power of the approach of using granularity for data retrieval also requires an appreciation of information left out from the query answers. Taking Swiss-Prot [32] as (incomplete) reference repository for toxins that are proteins[9], the query needs to find the unique cholera toxin out of over 1428 sequenced annotated toxins within the 190225 annotated protein sequences, the roughly 72000 entities and 1,9 million relations of the FMA, let alone if used in conjunction with mining MEDLINE for articles[10].

[9] More precisely: the recently started Tox-Prot Project [31]

[10] i.e. finding the right information about cholera at the desired level of detail in 12 million articles. GOPubMed [24] [1] can be seen as a very simple way of granular information retrieval, but is in fact *sorting* the standard PubMed query result along the Gene Ontology taxonomies and not that the query itself takes into account granularity in query formulation.

6.2 Structure in the contents of a granular level

The basic functions suffice for carrying out specific, targeted, queries to retrieve desired information. A better way of retrieving the data, however, has not been addressed adequately. In particular, the query answer of the *getContent*() function ignored *how* a taxonomic structure present in the underlying data can be preserved in the query answer. For instance, selecting a particular level within perspective and retrieving the contents (64) glosses over the fact that the environmental factors $xEnv$ are in a different branch of the tree from the four living habit predisposing factors.

$$\begin{aligned} &getContent(gp_9gl_2) = \\ &\quad \{SocEnv, PolEnv, EcoEnv, BioEnv, Diet, Stress, Smoking, PersHyg\} \end{aligned} \tag{64}$$

This information is captured in the underlying data source (see §3). At this level there is not one top entity: the coarser-grained level $\mathrm{gp_9gl_1}$ contains the entity *Environment* that subsumes the entities of $\mathrm{gp_9gl_2}$ $\{SocEnv, PolEnv, EcoEnv, BioEnv\}$, whereas $\{Diet, Stress, Smoking, PersonalHygene\}$ are subsumed by $LivingHabits$ in $\mathrm{gp_9gl_2}$. Reason why we do not observe an internal structure is because the granularity with respect to the two levels has been defined according to the levels of detail using the isA relation and no other relations have (yet) been defined. The predisposing factors are of the granularity type **npG** (non-scale-dependent, levels related with a primitive relation, see [6] for details) and the high-level goal to add this underlying semantics is to answer the queries "given the predisposing factor *Environment* at level $\mathrm{gp_9gl_1}$, retrieve the contents at level $\mathrm{gp_9gl_2}$" and "given the predisposing factor *LivingHabits* at level $\mathrm{gp_9gl_1}$, retrieve the contents at level $\mathrm{gp_9gl_2}$". More formally,

$$\begin{aligned} &\text{if } grain(Environment) = gp_9gl_1 \text{and } isA(x, Environment) \text{ and} \\ &\quad grain(x) = gp_9gl_2 \\ &\quad \text{then } getContent(gp_9gl_2) = \{SocEnv, PolEnv, EcoEnv, BioEnv\} \end{aligned} \tag{65}$$

$$\begin{aligned} &\text{if } grain(LivingHabits) = gp_9gl_1 \text{and } isA(x, LivingHabits) \text{ and} \\ &\quad grain(x) = gp_9gl_2 \\ &\quad \text{then } getContent(gp_9gl_2) = \{Diet, Stress, Smoking, PersHyg\} \end{aligned} \tag{66}$$

If we do not know the supertype, then we need to prefix the aforementioned queries with "retrieve the parent type of *SocEnv*, ..., *PersHyg*, then for each parent type, do...". The same implementation-level solution can be applied to the branches in the species taxonomy as mentioned in §3.

A prerequisite for generating interesting visuals of this information, is that the query answer must contain some structure of itself, such as a database table like depicted in Table 2, and transformed into an explanatory figure of, eventually, the type of Fig.2 such that the user can backtrack if s/he wishes to do so. A first step is the sub-tree visualisation alike the trees in Protégé[11] with or without ezOWL[12]

[11] http://protege.stanford.edu/
[12] http://iweb.etri.re.kr/ezowl/plugin.html

or OWLviz[13], GrOWL[10], DAG-Edit[14] etc., or a variation on Cytoscape[15] graphical representation (like done with the SEWASIE project) or DogmaModeler.

6.3 Other issues

The signature in §3 is constrained by the full first order logic characterisation of the TOG, which is omitted from this report. To name one aspect, the *involvedIn* and *partOf* relations have their semantics defined in the domain-independent TOG formalisation. This experiment assumes a proper model theory (a 'mapping' from the characterisation in FOL to ease using constructors) exists, but this is yet to be developed. Experimenting with applied domain granularity enabled exploring which elements are important and (in)convenient; these are the functions to assign levels to entities and information retrieval, and the relations. Also, it may be useful to allow rules to simplify information retrieval.

We now return to the two questions in §2.2, being 1) the correspondence between the rigid characterisation of granularity and its applicability to an arbitrarily chosen subject domain, and 2) the feasibility of implementing granularity. Regarding the former, the domain granularity framework does not violate the TOG. The initial inconsistency in the perspective of predisposing factors was solved by recognising that the 'inconsistency' concerning the relations between the granular levels is in fact another perspective that should be positioned orthogonally. An ontological trade-off is more likely to be expected in 'skipping', or condensing, levels to a reasonable and workable subset of the theoretically correct amount of levels in a perspective. Regarding the second aspect, what is lacking at the implementation level to bring it to the level of automation of the procedure and usage of applied domain granularity, there are several outstanding tasks. For an integrative subject domain like infectious diseases, there are at present insufficient ontologies for all the perspectives, which means that a lot of information has to be declared manually, which is prohibitive for successful adoption of granularity as a knowledge management methodology. Furthermore, the few ontologies available are offered in a sub-optimal format [7], in particular where operations require support for some version of recursive queries to either load the data into a level or for retrieval of contents of levels. Integration, or at least the linking, of the disparate data sources is another aspect that needs to be addressed satisfactorily. One advantage of the granularity-approach is that full integration is not required, hence the chance of success is within sight. Last, graphical visualisation for both the development of a domain granularity framework as well as the retrieval of information can be advantageous for the understanding of the system, in particular when a user does not use the system on a daily basis. Another service to the user can be to offer a set of types of queries where only a particular level or entity has to be selected.

[13] http://www.graphviz.org/
[14] DAG-Edit. http://www.godatabase.org/dev/.
[15] http://www.cytoscape.org/.

7 Conclusions and further research

Limitations of the data sources complicates applying a domain granularity framework, but developing a domain granularity framework is possible and does not violate the domain- and implementation-independent theory of granularity. Reasoning over the applied domain granularity framework, demonstrated with cholera and blood, enables targeted searches and inferencing for advanced knowledge management.

Future work includes developing an easy-to-use model theory, manipulation of queries and their answers, and to experiment with another subject domain to ensure genericity of the approach.

References

1. Doms, A., Schroeder, M. GoPubMed: Exploring PubMed with the GeneOntology. *Nucleic Acids Research*, 2005, 33:W783-W786.
2. Editorial. Lessons from Listeria. *Nature Structural & Molecular Biology*, 2005 12: 1.
3. Grizzi, F., Chiriva-Internati, M. The complexity of anatomical systems. *Theoretical Biology and Medical Modelling*, 2005, 2:26.
4. Hollinger, F.B., Kleinman, S. Transfusion transmission of West Nile virus: a merging of historical and contemporary perspectives. *Transfusion*, 2003, 43(8): 992-997.
5. Hunter, P.J., Borg, T. Integration from Proteins to Organs: The Physiome Project. *Nature*, 2003, 4(3): 237-243.
6. Keet, C.M. A taxonomy of types of granularity. *IEEE Conference in Granular Computing (GrC2006)*, 10-12 May 2006, Atlanta, USA.
7. Keet, C.M. *Granular information retrieval from the Gene Ontology and from the Foundational Model of Anatomy with OQAFMA*. Technical Report KRDB06-1, Faculty of Computer Science, Free University of Bozen-Bolzano. 2006.
8. Keet, C.M. Granularity as a modelling approach to investigate hypothesized emergence in biology. (*submitted*).
9. Keet, C.M., Kumar, A. 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.
10. Krivov, S., Williams, R., Villa, F. *A Visualization Model for the Languages of Semantic Web*. Technical Report, University of Vermont. 9-6-2005. http://www.uvm.edu/∼skrivov/growl.pdf. (Date accessed: 3-10-2005).
11. Kumar, A., Smith, B., Novotny, D.D. Biomedical Informatics and Granularity. *Comparative and Functional Genomics*, 2005, 5(6-7): 501-508.
12. Maddison, D.R., Schulz, K.-S. (ed.). The Tree of Life Web Project. 2004. http://tolweb.org.
13. Melhorn, H. (ed.). *Encyclopedic reference of parasitology*. 2nd ed. Springer-Verlag Heidelberg, 2004.
14. Rodal, A.A., Sokolova, O., Robins, D.B., Daugherty, K.M., Hippenmeyer, S., Riezman, H., Grigorieff, N, Goode, B.L. Conformational changes in the Arp2/3 complex leading to actin nucleation. *Nature Structural & Molecular Biology*, 2005 12: 26-31.
15. Rossetto, O., de Bernard, M., Pellizzari, R., Vitale, G., Caccin, P., Schiavo, G., and Montecucco, C. Bacterial toxins with intracellular protease activity. *Clinica Chimica Acta* 2000, 291 (2): 189-199.
16. Schlegel, H.G. *General Microbiology*. 7th ed. Cambridge: Cambridge University Press, 1995.
17. Stryer, L. 1988. *Biochemistry*. New York: WH Freeman and Co, 3rd ed. 1089p.
18. Tange, H.J., Schouten, H.C, Kester, A.D.M. and Hasman, A. The Granularity of Medical Narratives and Its Effect on the Speed and Completeness of Information Retrieval. *Journal of the American Medical Informatics Association*, 1998, 5(6): 571-582.
19. Wang, G., Van Dam, A.P., Schwartz, I., Dankert, J. Molecular Typing of Borrelia burgdorferi Sensu Lato: Taxonomic, Epidemiological, and Clinical Implications. *Clinical Microbiology Reviews*, 1999, 12(4): 633-653.

20. Weiss, R.A., McMichael, A.J. Social and environmental risk factors in the emergence of infectious diseases. *Nature Medicine*, 2004, 10, S70-S76.
21. Widell, A., Elmud, H., Persson, M.H., Jonsson, M. Transmission of hepatitis C via both erythrocyte and platelet transfusions from a single donor in serological window-phase of hepatitis C. *Vox Sang*, 1996 71(1): 55-7.
22. Zukerman, I., Albrecht, D., Nicholson, A., Doktor, K. Trading off granularity against complexity in predictive models for complex domains. In: *Proceedings of the 6th International Pacific Rim Conference on Artificial Intelligence (PRCAI 2000)*.
23. Foundational Model of Anatomy (FMA).
2003. http://fme.biostr.washington.edu:8089/FME/index.html.
24. GoPubMed. http://www.gopubmed.org.
25. International Classification of Diseases (ICD-10). 2003.
http://www.who.int/classifications/icd/en/.
26. Merck. Pneumonia. http://www.merck.com/mrkshared/mmanual/section6/chapter73/73a.jsp.
27. National Center for Biotechnology Information. Organism classification.
http://www.ncbi.nlm.nih.gov/Taxonomy/.
28. National Center for Infectious Diseases. http://www.cdc.gov/ncidod/.
29. Pathologie Online. http://www.pathologie-online.de/.
30. Snomed CT. http://www.snomed.org/snomedct/.
31. ToxProt: The Swiss-Prot toxin annotation project. http://www.expasy.org/sprot/tox-prot/.
32. UniProt. http://www.expasy.org/sprot/.