Cleaning up the GLIF model

1) Many concepts are duplicated in the model. This applies for both slots and classes. For example, the slots “concept_code”, “concept_id”, and “term_id” mean the same thing. Similarly, the classes Concept, Concept_Code and Vocabulary_Code all refer to the same thing and have the same slots (the slots are still named differently, but this should also be changed).
2) The USAM ontology defines classes such as Time_Interval, Time_Stamp, and Duration_Interval, which are also defined in the GLIf ontology. However, in USAM the attributes that I put are text strings (and not the complex nested structures that Gunther developed), whereas in the GLIF ontology, the attributes are more structured. We can define a Time&Duration ontology that will be included both by USAM and GLIF (it will be part of GLIF, but in an included Protégé project. This way, any RIM can refer to its concepts).
3) We added a class called Data_Item_Or_Primitive_Item. We did this so that we can uniformly treat data items whose structure is defined by:
   a) USAM RIM
   b) User-defined data model class
   c) A simple attribute (e.g., integer, string)

   However, a data_item refers to a concept and a data model class, and a simple attribute thus has these fields as blanks always, which doesn’t make sense. I suggest that we take away the possibility to include a simple attribute as the value of a data item. Simple data items can then be defined separately.

   The classes that used the Data_Item_Or_Primitive_Item objects were: Get_Data_Action_Specification (for specifying the Data_Item from which data is to be retrieved) and Data_Item (for storing its value). I think that in both cases we should allow just Data_Item_Value (a super-class of a USAM patient *Data and a user defined Data*Model_Instance). There is no need to “get” the data of a simple attribute because we can directly refer to it in expressions and its current value will be used, just like a simple local variable in computer programs. For Data_Item, as I explained before, we shouldn’t allow a value that does not refer to a medical concept any way.
   We do need to define the scope of variables of simple type that are used by guidelines and subguidelines.