**Global Exposure Database for Global Earthquake Modeling (GED4GEM)**

**Data Model Description**

This document describes the GED4GEM data model; its main structures, tables, attributes, and their relationships.

Global Grid

A global grid of 1 km resolution (30 Arc Seconds) is stored in the database and represented by the grid cell’s midpoint (grid point), using its latitude and longitude (as shape file geometry and numbers). Only land areas are included; oceans and uninhabited places such as Antarctica are excluded. Basic attributes of the grid cell such as the land area, categorization as urban/rural area, and quality information related to the urban/rural classification are also stored. The grid geometry and attribute information is generated using the “Land/Geographic Unit Area Grids” and “Urban Extents Grid” datasets from the SEDAC Global Rural-Urban Mapping Project, Version 1 (GRUMPv1), <http://sedac.ciesin.columbia.edu/gpw/global.jsp>. See tables *grid\_point*, and *grid\_point\_attribute*.

Additional information about the grid cell such as the country, the administrative units – down to three levels, and the CRESTA zone and sub-zones it belong to are also stored in the database. All the exposure data are either aggregated or disaggregated to the grid cell level depending on the resolution of original data.

National and Sub-National Administrative Units

## The geometry and attribute information of countries and their administrative units down to three levels - admin 1/, admin 2, and admin 3 (for example region, province, and municipality in Italy) – are stored in the database. The parent-child relationship between these administrative units is also stored in the database using foreign key references. These data are derived from the [GADM database of Global Administrative Areas](http://www.gadm.org/home), <http://www.gadm.org>. See tables *gadm\_country, gadm\_admin\_1, gadm\_admin\_2,* and *admin\_3*. These tables are time stamped (using the *date* column) and have the ability to store changes in boundaries and names over time.

## The tables *grid\_point\_country, grid\_point\_admin\_1, grid\_point\_admin\_2,* and *grid\_point\_admin\_3* store the information on what country and administrative units each grid point belongs to. In some cases, due to changes in the boundaries of countries and their administrative units over time, the units to which a grid point belongs to could change over time. So, these tables allow for tracking these changes over time by creating new associations between the grid point and a country and/or administrative units.

## CRESTA Zone and Sub-Zones

## The information about CRESTA countries and zones/sub-zones is stored in tables *cresta\_country* and *cresta\_zone*. The relationship between CRESTA countries and GADM countries are tracked using foreign key references between the *cresta\_country* and *gadm\_country* tables. The CRESTA zone and sub-zone of a grid point is stored using foreign key references between the *cresta\_zone* and *grid\_point\_attribute* tables.

## Population at Grid Cell Level

The population count (*pop\_value* column) of each grid cell is stored in the *population* table along with any available quality information (*pop\_quality* column) of the population measure. The data model has the ability to store population data for each grid cell from multiple data sources. The population source information is stored in the *population\_src* table. Each record (row) in the *population* table is associated with a grid cell and a population data source via foreign key references to the *grid\_point* and *population\_src* tables. Initially the database will have population data derived from the “Population Count Grids; Year 2000”, dataset from the SEDAC Global Rural-Urban Mapping Project, Version 1 (GRUMPv1), <http://sedac.ciesin.columbia.edu/gpw/global.jsp>. However, later on population data from other sources such as Landscan or a newer version of GRUMP could be added.

Population Ratios Allocation at Country Level

## The *pop\_allocation* table stores the day time, night time and transit population ratios, at the country level, for each combination of urban/rural categorization and occupancy classes. For example, in the beginning we have only two occupancy classes, residential and non-residential. So for each country, there will be four sets of day/night/transit population ratios; urban/residential, urban/non-residential, rural/residential, and rural/non-residential. These ratios for each country are obtained from the PAGER database.

## Mapping Scheme

Mapping scheme is a way to assigning attributes to building stock by using statistical inferences on known attributes. GED4GEM uses mapping schemes to create the global exposure.

Currently, PAGER taxonomy is used, until GEM taxonomy is finalized.

GED4GEM mapping schemes are structured as trees, with nodes containing the class names and corresponding weights. Each level represents one attribute mapping, and the nodes under same parent are the statistical distribution of attributes. The sum of the weights for those nodes are 1, making up for the entire distribution. Example: if you assign the mapping scheme given below to region A, then in region A, there is a “structure” level, with 3 nodes, “wood” (25%), “masonry” (50%), “steel”(5%) and “concrete” (20%). This mapping scheme therefore states that, in region A, there are 25% wood buildings, 50% masonry buildings, 5% steel buildings and 20% concrete buildings. Additional levels can be defined under each of the nodes, allowing more co-related attributes to be assigned.

Drawing 1: mapping scheme as tree

To store this tree structure in a relational database, we use a self-referencing table *mapping\_scheme (ms)* along with reference tables *mapping\_scheme\_class(msc)* and *mapping\_scheme\_type(mst)*. The *mapping\_scheme\_type* table stores information about the attributes, for example “Structure” or “Height” along with the taxonomy used (e.g. PAGER or GEM). The *mapping\_scheme\_class* table stores the attribute values, e.g. “Wood” or “High”. Each row in the *mapping\_scheme (ms)* table contains attribute value and weight, as well as ID of its parent node. A root node is created for any mapping scheme, and then all attribute mappings are attached to this top node.

Table 1: Mapping Scheme Tree represented as Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | ms.parent\_ms\_id | attribute  (mst.name) | value  (msc.name) | weight  (ms.ms\_value) |
| 1 | NULL | ROOT | ROOT | 1 |
| 2 | 1 | Structure | Wood | 0.25 |
| 3 | 1 | Structure | Masonry | 0.5 |
| 4 | 1 | Structure | Steel | 0.05 |
| 5 | 1 | Structure | Concrete | 0.2 |
| 6 | 2 | Height | Low | 1 |
| 7 | 3 | Height | Low | 0.8 |
| 8 | 3 | Height | Medium | 0.2 |
| 9 | 4 | Height | Medium | 0.4 |
| 10 | 4 | Height | High | 0.6 |
| 11 | 5 | Height | Medium | 0.8 |
| 12 | 5 | Height | High | 0.2 |

Table 2: mapping scheme as table

During the final attribute assignment, this table must join to self to construct the following mapping table.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Structure | Height | Weight |
| 1 | Wood | Low | 0.25 |
| 2 | Masonry | Low | 0.4 |
| 3 | Masonry | Medium | 0.1 |
| 4 | Steel | Medium | 0.02 |
| 5 | Steel | High | 0.03 |
| 6 | Concrete | Medium | 0.16 |
| 7 | Concrete | High | 0.04 |

Different mapping schemes can be stored in the database. Each mapping scheme can be unambiguously described using different mapping scheme sources. The *mapping\_scheme\_src* table stores information about the mapping scheme sources including who (*source* column) developed a particular mapping scheme (the tree) and the *taxonomy* used (e.g. PAGER or GEM).

Aggregated Building Infrastructure Data

A specific mapping scheme can be applied to a specific area within a study region to assign building type data to that area. That is the deepest attribute values of each attribute type in the path of a leaf node of the mapping scheme, for example the Structure/Height combinations in Table 2 above. Following paragraphs describe how this is accomplished.

The information about a specific area, including the geometry, the study region it belongs to, and the mapping scheme applied is stored in the *agg\_build\_infra\_src* table. If a specific area in that table exactly matches a country or an administrative unit, then the corresponding foreign key references (*gadm\_country\_id* etc.) will be set appropriately. In this case, since the geometry information of the area can be obtained from the corresponding *gadm\_country* or administrative unit tables (*gadm\_admin\_1* etc), the geometry information will not be repeated in the *agg\_build\_infra\_src* table.

Information about study regions are defined in the *study\_region* table. A study region can contain multiple areas. Within a study region, only one specific mapping scheme will be applied to a specific area. However, different mapping schemes can be applied to different areas within a study region. For example, let’s assume a study region A contains Italy, France, and Austria. You can apply mapping scheme 1 for Italy, 2 for France, and 3 for Austria. However, if you want to apply mapping scheme 1 for Italy and France, and 2 for Austria, then you have to create a different study region B containing Italy, France, and Austria. In other words, a specific study region can contain a specific area only once.

Finally, the building type data (that is the deepest attribute values of each attribute type in the path of a leaf node of the mapping scheme, for example the Structure/Height combinations in Table 2 above) for the specific area in a given study region and mapping scheme is stored in the table *agg\_build\_infra*. To avoid potentially creating hundreds of columns for each mapping scheme attribute type that could be added in future, for example with GEM taxonomy, the building type data are stored as rows instead of columns. Then specific combination of attribute values (only the deepest values for each attribute type) leading to a leaf node are identified by biding them together using a record number (*ms\_class\_group* column). The final weight of each of the record (building type) is the product of the weights of each node in the path of leaf node (see the Weight Column in Table 2 above).

Population Allocation by Building Type at Grid Cell Level

For each grid cell in a specific area within a study region, the day/night/transit population allocation ratios for each building type can be computed by multiplying the weight of that building type by the day/night/transit population ratios of that grid cell (which can be obtained from the appropriate set of population allocation ratios at the corresponding country level for the given urban/rural and occupancy category of the grid cell). The day/night/transit population allocation ratios for each building type for a grid cell (in a specific area within a study region) is stored in the *agg\_build\_infra\_pop\_ratio* table. Finally, using these ratios and the population data of the grid cell (see *population* table), the day/night/transit population counts for each building type within a grid cell is computed and stored in the *agg\_build\_infra\_pop* table along with number of buildings (*num\_buildings* column) and the area (*struct\_area* column). Obviously, different population sources can be used to estimate the population count for each building type; the database supports such computation and ability to store them (see foreign key reference *population\_id* in *agg\_build\_infra\_pop* table to the *population* table).

De-normalized Pager Exposure

Due to the highly abstract and complex nature of the relations database, a de-normalized flat table is created using materialized view to store pager exposure data. This will enable relatively easier querying and visualization of pager exposure data by client application tools. See table *pager\_exposure* for details. It basically stores the geometry of the grid point, land area and urban/rural information, GADM country and administrative unit information, CRESTA zone and sub zone information, building type data, and the day/night/transit population ratios and count by building type along with number of buildings and area. Note that a grid point will appear multiple times in this table for each unique study region and building type combination. So, specifying a specific study region and building type in a query will ensure that the result set will contain a specific grid point only once.