Lindholmen database schema

Figure 1 shows Lindholmen database schema. It contains 16 tables, which can be split into following two parts:

Part 1: UML Projects Meta-data contains meta-data of the projects that use UML models. Example of the meta-data are the commits where UML models were introduced/updated, the developers who committed and the co-changed files.

Part 2: Class Diagram Details contains contents extracted from class diagram images. Example of the class diagram components stored in this database are class attributes, methods, method parameters, associations between classes.
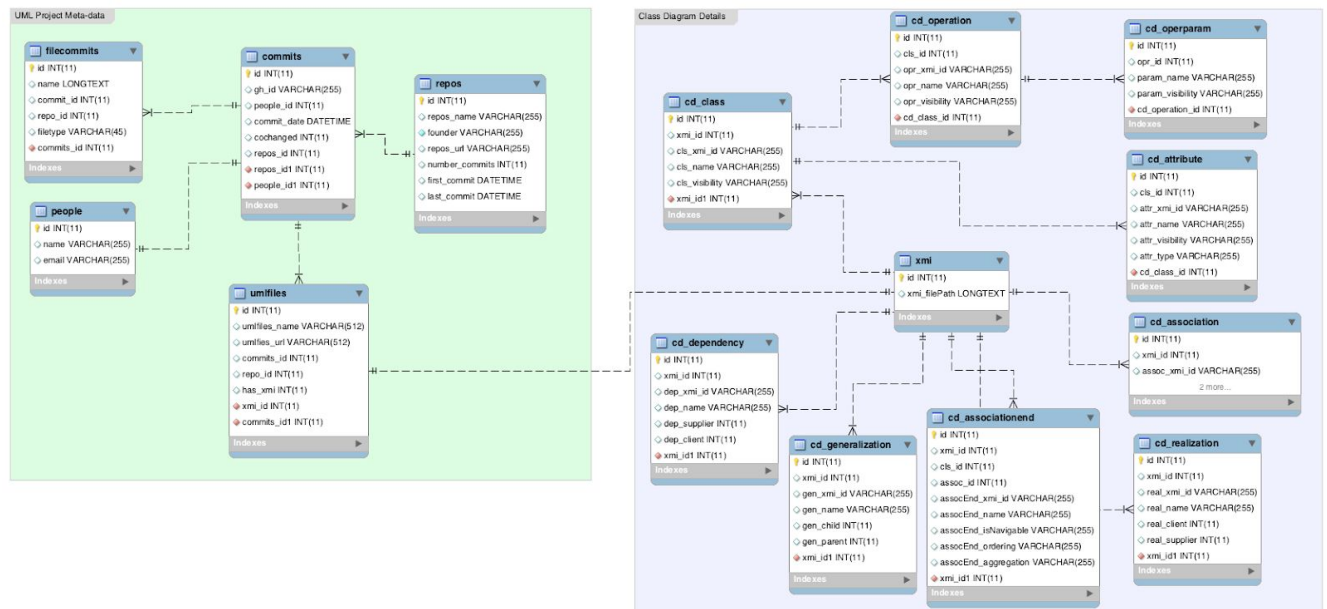

Figure 1. Database schema of the Lindholmen database
Downloadable at
http://oss.models-db.com/Downloads/SATToSE2018_Hackathon/lindholmen_dbschema.png

Details per table

1. Table repos: This table contains general information about repository. Each repository is represented by a record in the repos table.

Name	Data type	Description

id	INTEGER	Repository unique number (Primary Key)
repos_name	VARCHAR	Repository name
founder	VARCHAR	Github ID of the founder of the repository
repos_url	VARCHAR	URL to the Github page of the repository
number_commits	INT	Number of commits committed from (including both) first commits to last commits
first_commit	DATETIME	Timestamp of the first commits
last_commit	DATETIME	Timestamp of the last commits

2. Table commits: contains general information about the commits of repositories. Every commit in the repository is represented by a record in the commits table.

Name	Data type	Description
id	INTEGER	Commit unique number (Primary Key)
gh_id	VARCHAR	Corresponding Github commit ID
people_id	INTEGER	Committer identifier
commit_date	DATETIME	Timestamp of the commit
cochanged	INT	Binary value. True, if the file has been committed with other files at the same time. False, if committed alone
repo_id	DATETIME	Repository identifier (Foreign Key)s

3. Table filecommits: contains general information of files that was being committed together with its type.

Name	Data type	Description
id	INTEGER	File unique number (Primary Key)
name	LONGTEXT	File name including extension
commit_id	INTEGER	Commit identifier (Foreign Key)
filetype	VARCHAR	Type of file (Foreign Key)
repo_id	INTEGER	Repository identifier (Foreign Key)

4. Table umlfiles contains details of UML models. Each row represents one UML file.

Name	Data type	Description
id	INTEGER	UML file unique number (Primary Key)
umlfiles_name	LONGTEXT	UML file name including extension
umlfiles_url	VARCHAR	URL to the UML file. It is noted that some of the URLs are outdated or unreachable at the accessing time. That's because of several possible reasons such as: i) the file no longer exist; ii) file name has changed; iii) access right halted.
commits_id	INTEGER	Commit identifier (Foreign Key)
repo_id	INTEGER	Repository identifier (Foreign Key)
has_xmi	INTEGER	Each UML diagram should has a corresponding .xmi file that stores the content of the diagram in XMI format. This field contains the identifier of the XMI file that represent the UML diagram (links to table xmi)

5. Table people: contains general information about contributors of the repositories.

This table is the same as above!

Name	Data type	Description
id	INTEGER	UML file unique number
umlfiles_name	LONGTEXT	UML file name including extension
umlfiles_url	VARCHAR	URL to the UML file. It is noted that some of the URLs are outdated or unreachable at the accessing time. That's because of several possible reasons such as: i) the file no longer exist; ii) file name has changed; iii) access right halted.
commits_id	INTEGER	Commit identifier
repo_id	INTEGER	Repository identifier
has_xmi	INTEGER	Each UML diagram should has a corresponding .xmi file that stores the content of the diagram in XMI format. This field contains the identifier of the XMI file that represent the UML diagram (links to table xmi)

6. Table xmi contains information about .xmi presentation of uml diagrams. Each UML diagram should has a corresponding .xmi file that stores the content of the diagram in XMI format. However, there are only a few diagrams stored in .xmi format. Most of the .xmi file stored in this table were generated from our tool (IMG2UML - see [REF]) from class diagram images.

Name	Data type	Description
id	INTEGER	XMI file unique number (Primary Key)
xmi_filePath	LONGTEXT	This refers to the file path to the .xmi file stored in our local machine. The xmi files are provided on request to admins of the database.