

Hi Karen,

About your question:

"why the flexible_mark and rubric_mark can't be collapsed into one table".

This is my answer:

(You can see that text on page 6 of the analysis document)

If the Instructor decides to switch to another marking scheme, he can do it at any moment. If grading is started, a warning message will be displayed, and the modification will be applied if he really wants to continue. The grading done with the other marking scheme will remain in the database, but will not be accessible by the application.

This is the schema of what could be scenario 4

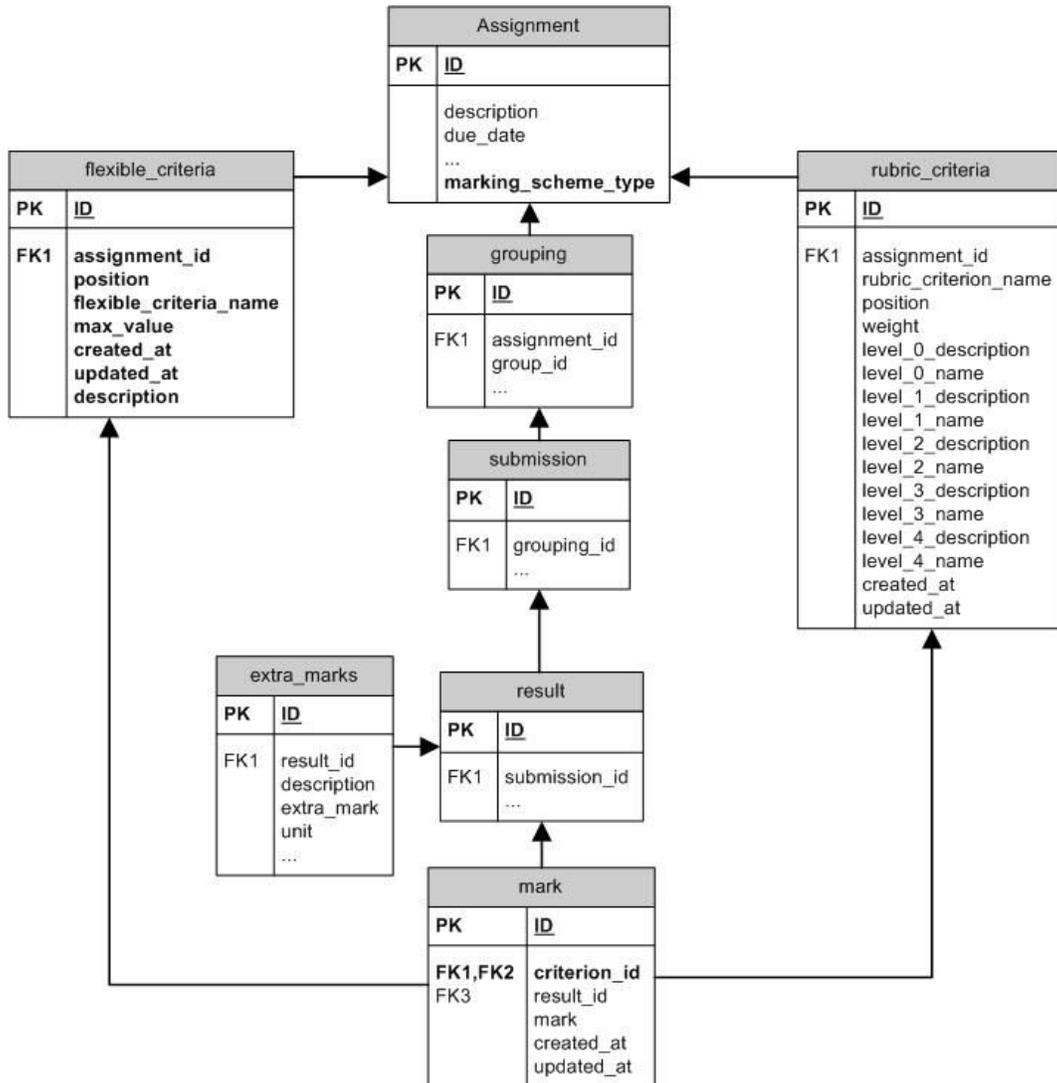


Figure 1: Scenario 4

If we really want that a grading, started with another marking scheme remains in the DB, this scenario would not be good. Let me explain with some data simulation:

Imagine that you have started grading with rubric marking scheme this is an example of a records that could have been created in the table Mark:

Table	ID	Result_id	criterion_id	Mark
Mark	1	1	1	4

Imagine now that you decide, for any reason, to change of marking scheme, and enter a flexible criterion. You than get as flexible_criterion_id, the id number 1... That is something possible, since the flexible_criteria table and rubric_criteria table are 2 tables totally different.

Table	ID	Result_id	criterion_id	Mark
Mark	2	1	1	4.5

When the application will go through all the flexible marking scheme's marks related to criterion of id 1, you will see all the marks associated to criterion_id 1. This means that you will see the marks that are related to rubric marking scheme also.

This is some way to solve it:

- 1- Create a join table, like scenario 1;
- 2- Add a new field in the table mark that will keep the flexible_criterion_id, but we usually don't like this kind of table in our DB, because if it is the rubric marking scheme, we need to assign "?" to flexible_criterion_id, and if the current marking scheme is flexible's, rubric_criterion_id has to be "?"... We don't like having "?" in our indexes...

Mark	
PK	ID
FK2	rubric_criterion_id
FK1	flexible_criterion_id
FK3	result_id
	mark
	created_at
	updated_at

- 3- We could probably create a function that could find the last criterion_id used and use it, but we don't want to go there;
- 4- Or we make sure that if we change of marking scheme, nothing remains in the DB that was related with the previous marking scheme. This can be done by giving a warning to the user that all the data related with previous marking scheme will be lost. If the user accepts, the application do the cleaning.
- 5- We could also extract the mark from table flexible_criteria and from rubric_criteria like presented in scenario 5... One more link one to one and we add the field mark_id in tables rubric_criterion and in flexible_criterion.

On the next page there is scenario 5, which is another version of scenario 4, but with this version, we can keep the grading done with another marking scheme in the DB, even though we change of marking scheme.

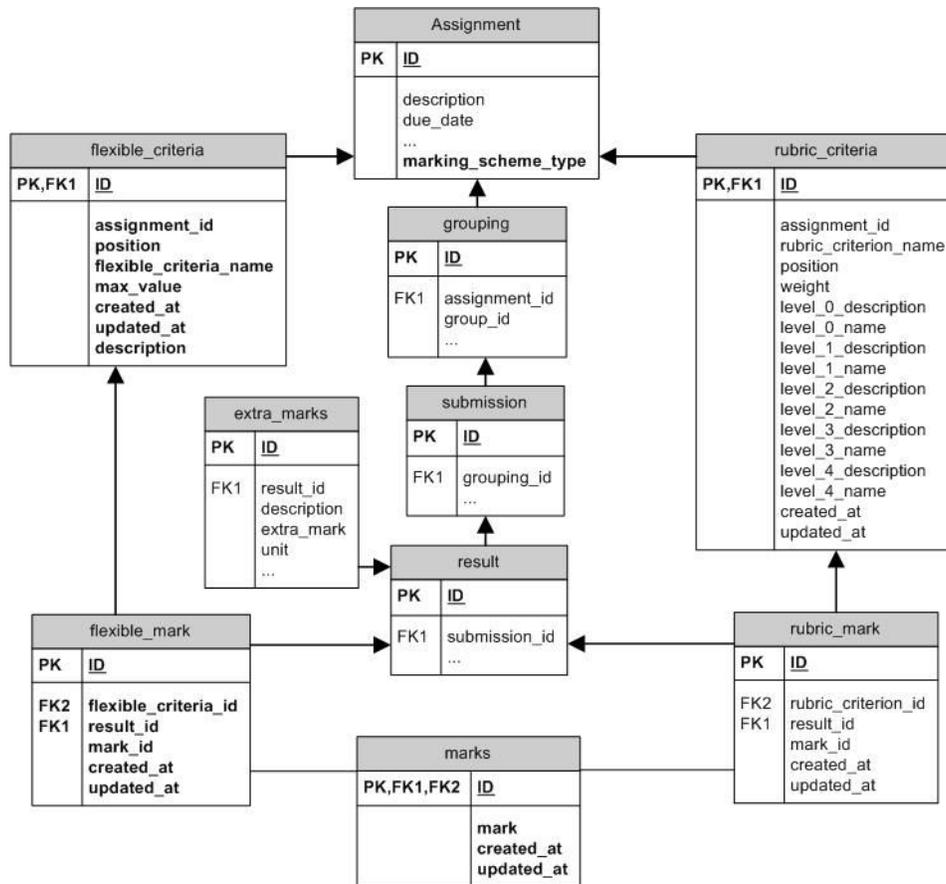


Figure 2: Scenario 5