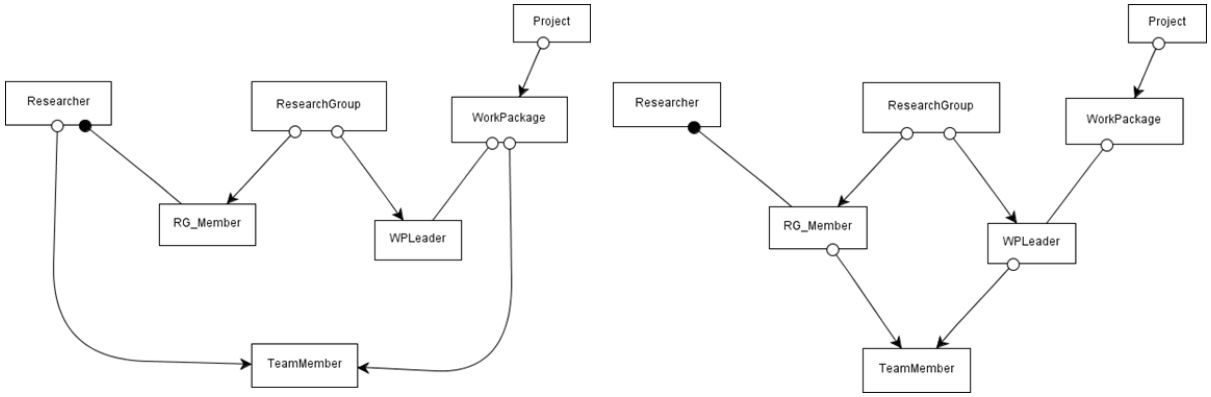
This is an exercise from the Merode course at KU Leuven.
The solution is given as an EDG (part of a Merode model)
Some of the mistakes from student solutions are specific to Merode, others are universal

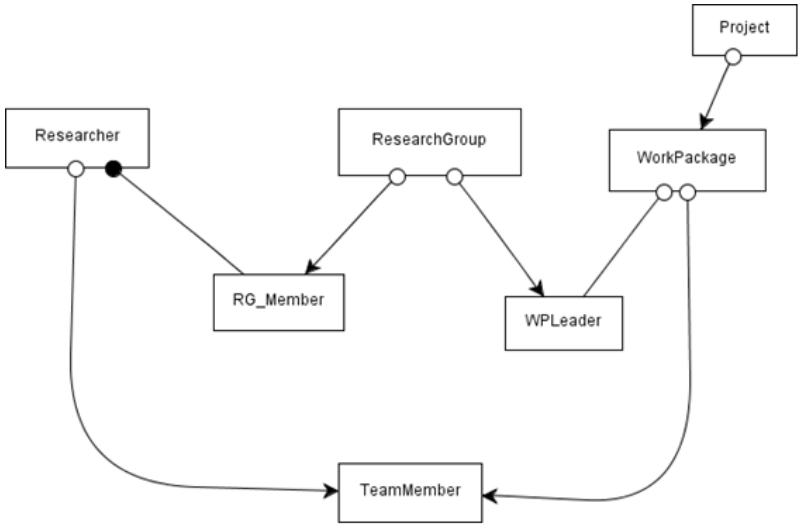
Case Description

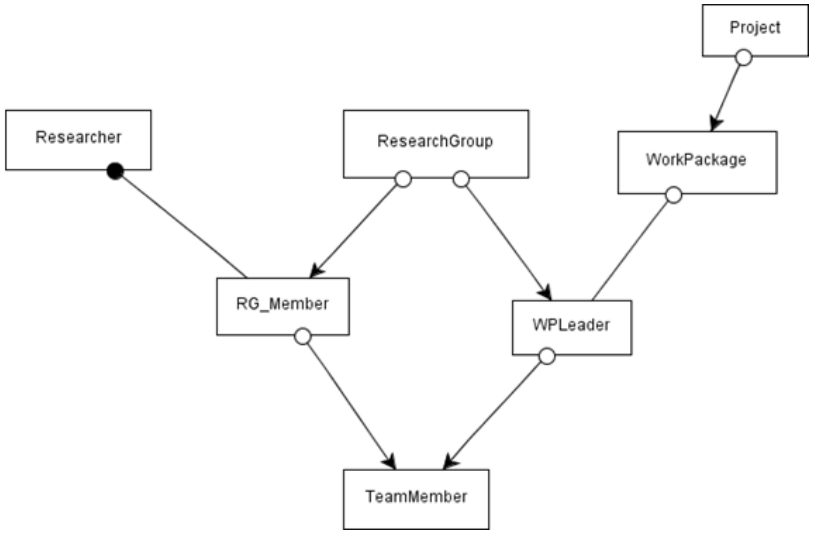
The following case describes a universe of research project management. The university has many research groups that belong to different departments. Researchers belong to exactly one research group. Projects are decomposed into Work Packages (WP), and each WP has exactly one research group assigned to it as WP leader. To execute a research project, researchers need to be assigned to Work Packages. Two EDGs are proposed:



Consider the two proposed EDGs and the following constraints. Investigate which combination of model and multiple propagation constraints enables you to capture the given constraints in the best way.

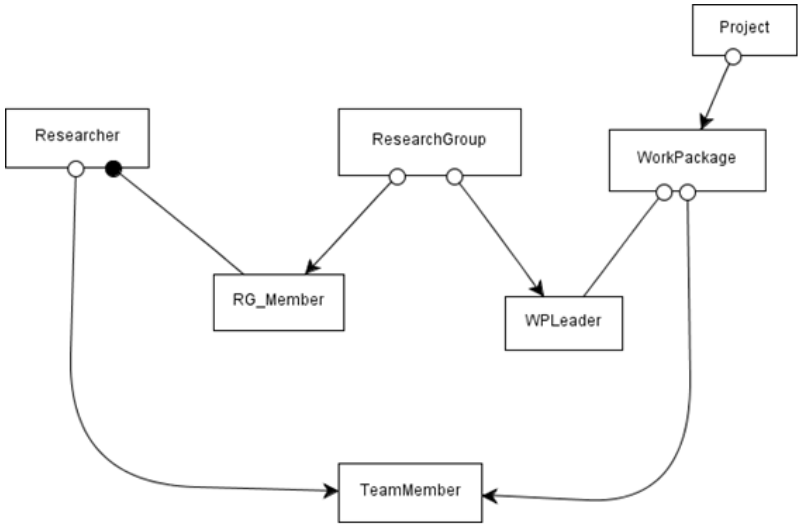
Situation 1

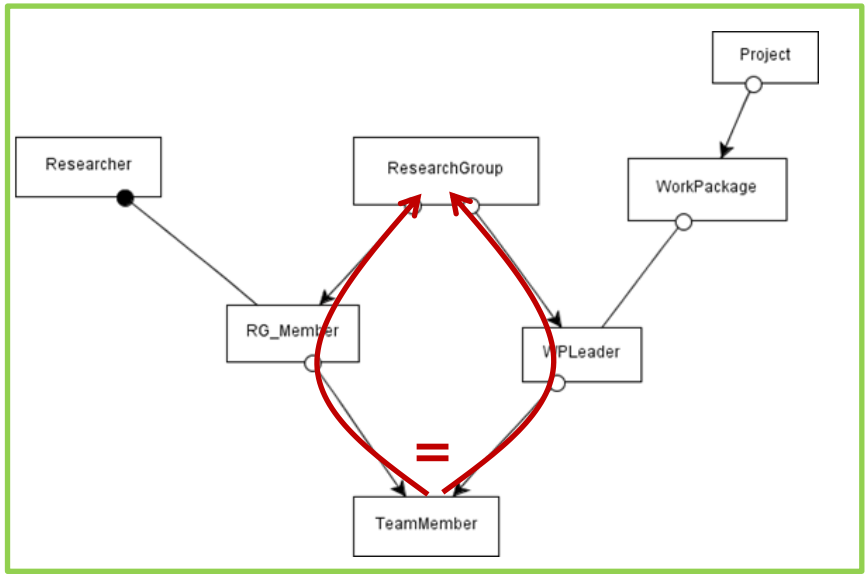




- A researcher can only be assigned to a WP his/her own research group is leading.

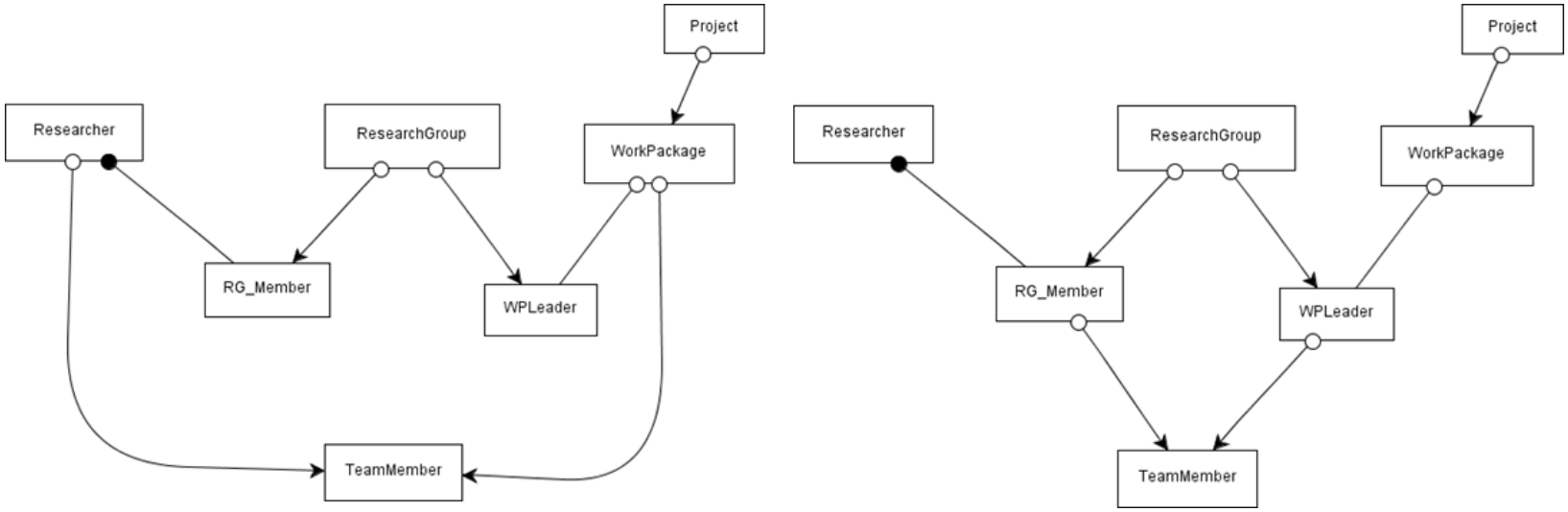
Situation 1





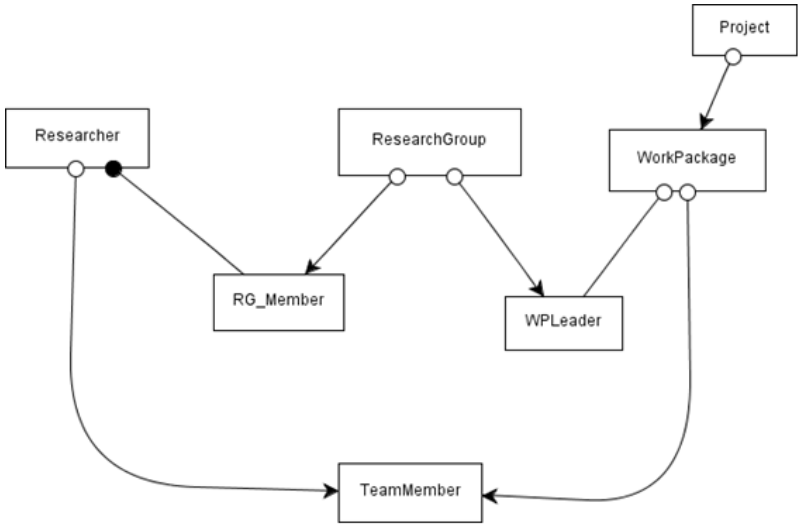
- A researcher can only be assigned to a WP his/her own research group is leading.
  - Model to the right
  - MPC teamMember.RGMember.RG = teamMember.WPLeader.RG)

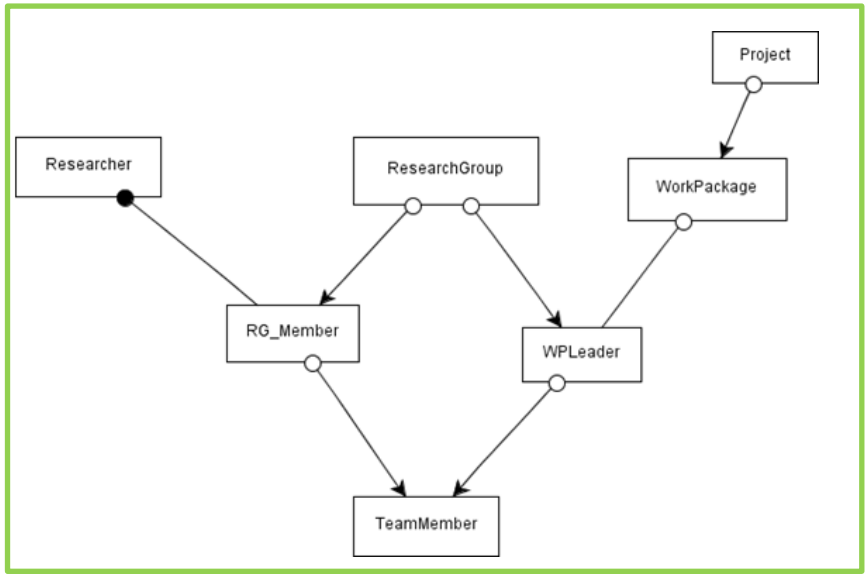
Situation 1



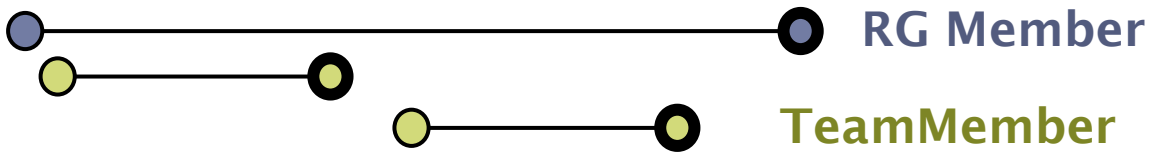
- Given the above constraint, when a researcher changes from research group (RG), the assignments s/he did while being in the first group need to be ended first before changing from group

Situation 1

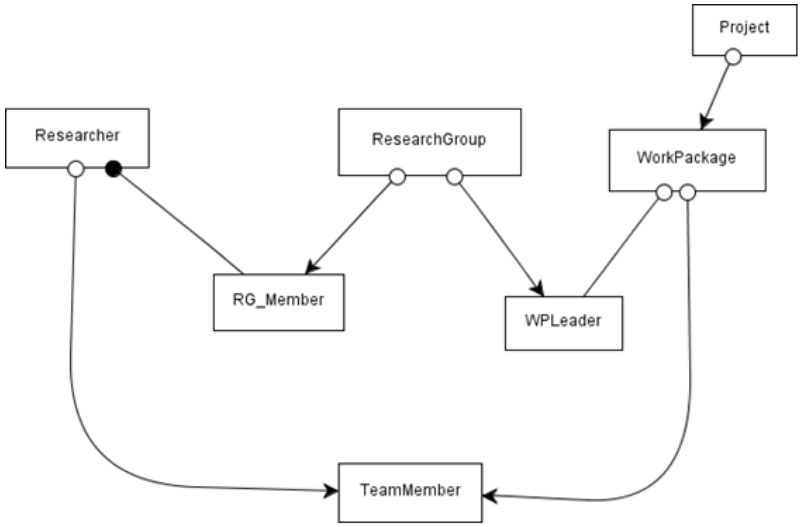


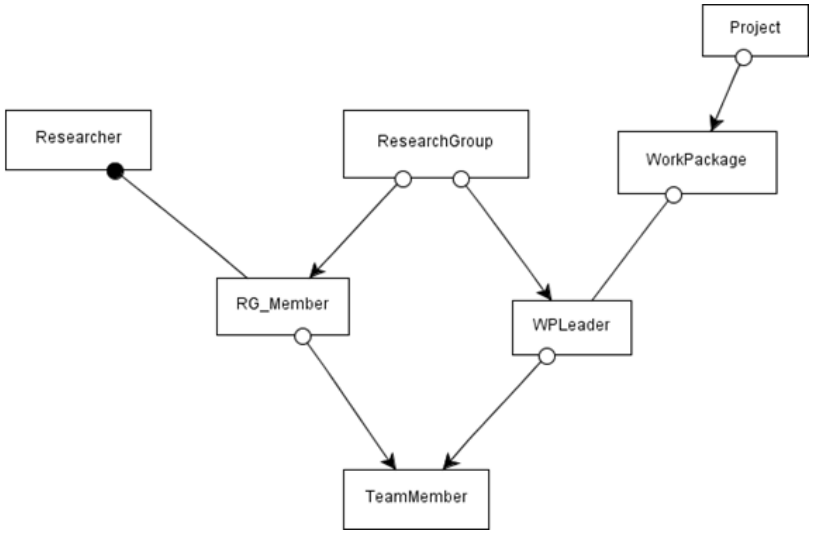


- Given the above constraint, when a researcher changes from research group (RG), the assignments s/he did while being in the first group need to be ended first before changing from group
  - Model to the right
  - Ending RG_Member requires ending the dependent obj. first = teamMember



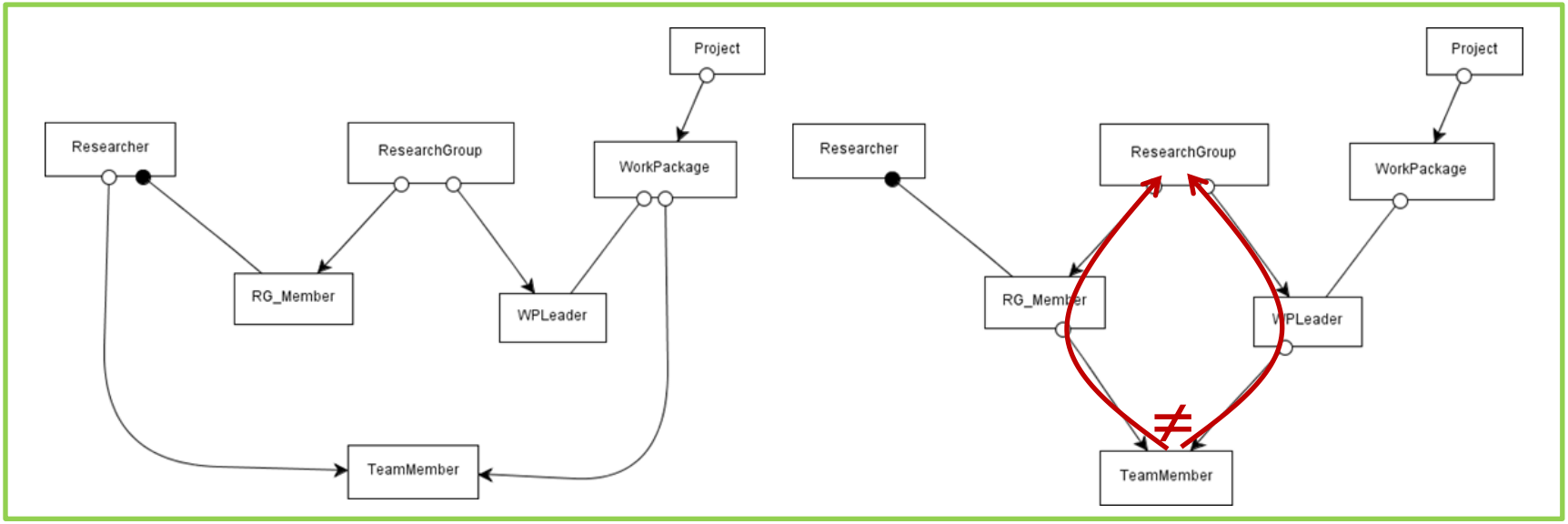
Situation 2





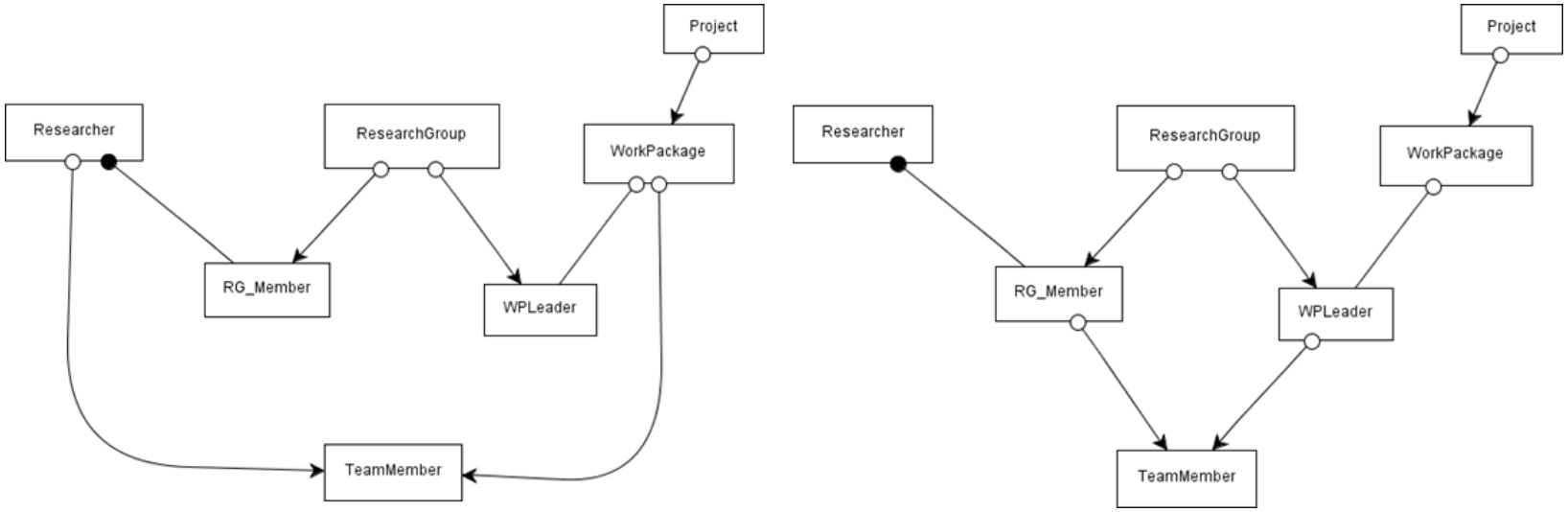
- A researcher can be assigned to any WP, independent of which RG is leading the WP

Situation 2



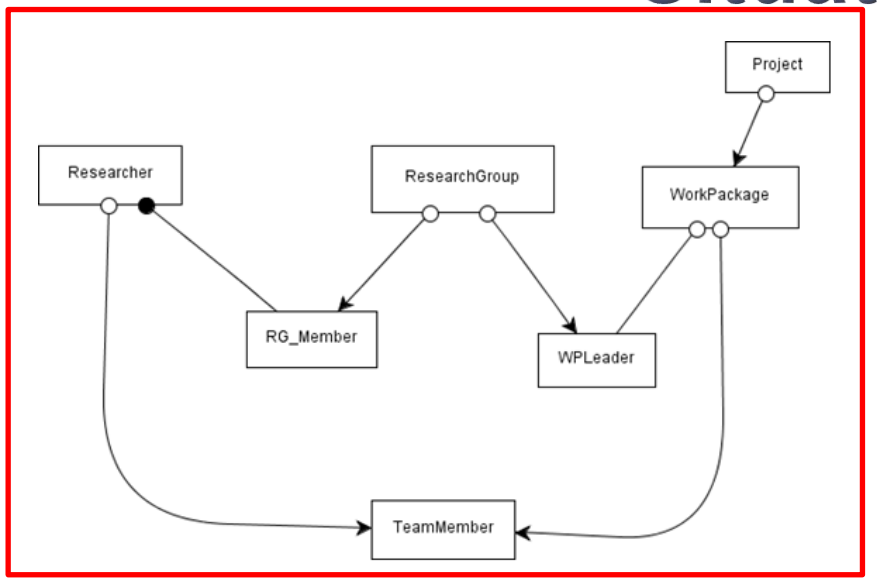
- A researcher can be assigned to any WP, independent of which RG is leading the WP
  - Model to the left + model to the right
  - No constraint

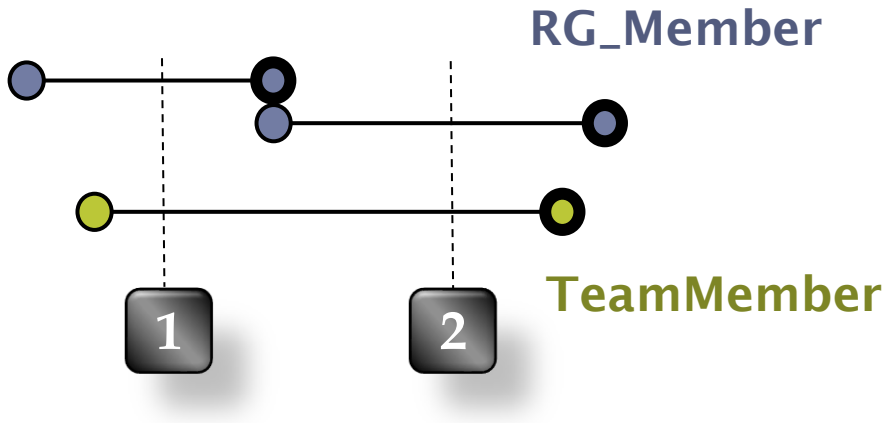
Situation 2



- However, “service delivery” from one group to another needs to be traceable. It should be possible to have an overview of researchers from other groups working on WPs led by a RG. This “service delivery” implies that whenever a researcher changes research group, the current assignments as team member of a WP are terminated and new ones are created to keep a correct record of the service delivery. Likewise, if WP leadership is re-assigned, team member assignments are re-recorded to ensure correct recording of the service delivery.

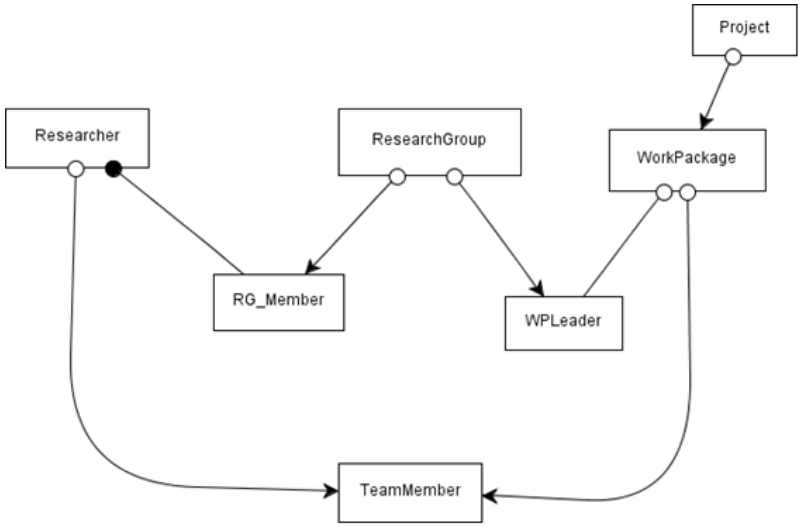
Situation 2

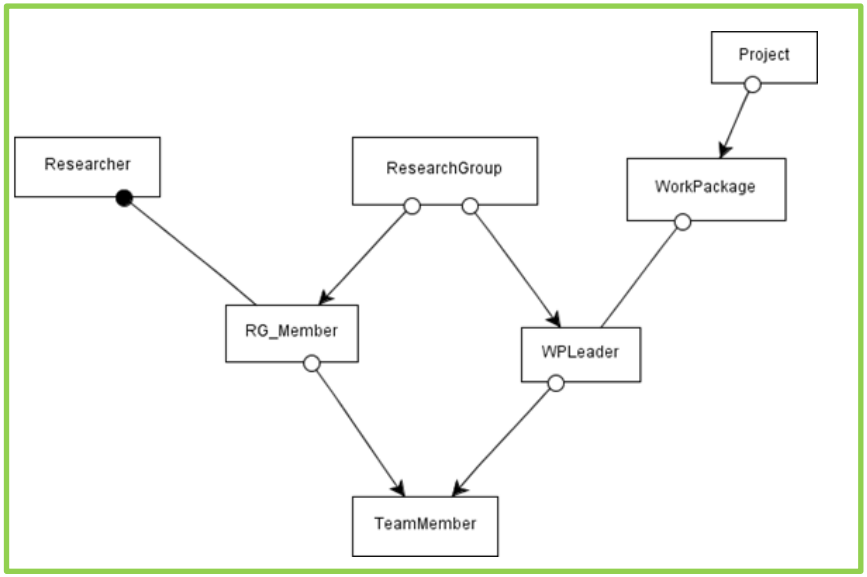




- However, “service delivery” from one group to another needs to be traceable. ... whenever a researcher changes research group ... keep a correct record of the (past) service delivery. ....
  - Model on the left:
    Life-cycles of RG_Member and TeamMember are unrelated
  - What happens when RG_Member is ended ?
    - At moment 1, team-membership seems in context of RG1 membership
    - At moment 2, same team-membership seems in context of RG2 membership

Situation 2





- However, “service delivery” from one group to another needs to be traceable. It should be possible to have an overview of researchers from other groups working on WPs led by a RG. This “service delivery” implies that whenever a researcher changes research group, the current assignments as team member of a WP are terminated and new ones are created to keep a correct record of the service delivery. Likewise, if WP leadership is re-assigned, team member assignments are re-recorded to ensure correct recording of the service delivery.
  - Model to the right
  - ending RG_Member and WPLeader requires ending the dependent obj. first = teamMember