Competence-Oriented Task Catalog For Graphical Modeling

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Model understanding and interpreting				
Remember / Understand				
Theoretical questions	This task class includes task types that involve asking theoretical questions to test knowledge and understanding related to general facts, concepts, and procedures to graphical modeling.			
Knowledge questions	This type of task tests knowledge of familiar content from the teaching context (e.g. terms, basic concepts and methods) without embedding it in a more complex task. It is assessed to which extent knowledge can be remembered and reproduced. These are text-based questions, i.e. the questions do not refer to a given model. Typical task formats are short free-text answers, forced choice tasks (correct/wrong), single/multiple choice tasks, matching tasks or cloze tests. This includes reproducing a concept definition (e.g., defining a particular concept of modeling, such as inheritance in UML, or naming a particular type of model element based on the notation.)	"List five different reasons for business process modeling." "What is UML and what is this language used for." "The EPC a) is a semiformal modeling language (true o / false o) b) is a DIN standard (true o / false o) c) is used to describe business processes (true o / false o)"	All competence facets at the process level "remember" could be addressed (MU0, MB0, VAB0, MC0, SC0).	Knowledge questions serve to address competence facets at the cognitive process level "remember", i.e., basic knowledge has to be recalled and reproduced. The CMGM does not specify any competence facet at this level as its focus is on higher cognitive process levels. Nevertheless, the reproduction of knowledge is considered an important prerequisite for competence facets at higher process levels, since this knowledge is used in more complex tasks and in problem solving.
Comprehension tasks	This type of task tests to which extent the course content has been understood in depth and includes questions and tasks on specific terms and concepts of modeling as well as modeling techniques. These tasks go beyond the mere reproduction of knowledge. These are text-based questions, i.e. the questions do not refer to a given model. Typical task formats include matching tasks, comparison tasks, free text tasks in which learners have to explain, justify or compare something (using their own words or examples).	"What is a model? Explain the term and give two examples including a characterization (formalization, dynamic/static, textual/visual)." "Explain the similarities and differences between the modeling languages BPMN and Petri nets." "For the following scenarios, decide what type of relationship each is, just simple association or special aggregation or even composition. In each case, indicate the strongest relationship present and give detailed reasons for your decision. (a) A color is in a painting. (b) The human body is composed of organs (...)"	MU 1.01 Learners are able to explain basic key terms and concepts of modeling (such as model, principles of model building, definition and modification of models, metamodels). MU 1.02 Learners are able to explain and compare concepts of modeling structures or behavior (i.e., static and dynamic aspects). MU 1.03 Learners are able to explain the strengths and weaknesses of specific modeling languages or model types for a specific purpose. MU 1.04 Learners are able to distinguish between different areas of modeling (e.g., software engineering, database design, business process modeling) and describe them in a differentiated manner. MU 1.05 Learners are able to explain possible modeling purposes in terms of the intended use or implementation of models (e.g., for code generation, simulation, or organizational design). MU 1.06 Learners are able to explain the principle of abstraction (generalization/specialization) and levels of abstraction. MU 1.07 Learners are able to explain criteria for evaluating model quality (e.g., syntax, semantics, pragmatics according to relevant frameworks such as SEQUAL). MU 1.08 Learners are able to explain syntactical rules of the modeling language(s) under consideration. MU 1.09 Learners are able to explain the meaning (semantics) of existing modeling elements and their relationships/connections in respect to the modeling language(s) under consideration. MU 1.11 Learners are able to explain the possibilities and limitations of graphical modeling. MV 1.12 Learners are able to explain formal properties of models (e.g., properties of Petri nets). MB 1.01 Learners are able to distinguish between descriptive and prescriptive modeling. MB 1.02 Learners are able to explain different approaches to modeling in terms of level of abstraction and formalization (formal, semi-formal). MB 1.03 Learners are able to explain modeling techniques in different areas of modeling (e.g., model-based requirements engineering, business process modeling). MB 1.04 Learners are able to explain the relevance of the modeling purpose for model building (especially for the selection and use of different modeling techniques). MB 1.05 Learners are able to explain the effects of certain modifications to a model. VAB 1.01 Learners are able to explain the objectives and relevance of graphical modeling for the respective area of modeling in computer science. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. VAB 1.03 Learners are convinced that modeling tasks can be solved through adequate procedures and the use of appropriate modeling techniques. VAB 1.04 Learners are convinced that planned action and a systematic procedure are necessary when solving complex modeling tasks. MC 1.01 Learners understand that methods in the field of graphical modeling (e.g., modeling tools or languages) continue to evolve and therefore recognize the need for lifelong learning.	In general, most competence facets at the process level "understand" could be addressed by a specific comprehension task or question. Competence facets which refer to understanding given models are not included here.

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Model explaining				
Identifying model elements	In this type of task, learners are asked to identify and describe single types of model elements of a given model.	"Define and describe all entity types and relationship types that are in the model."	MU 1.09 Learners are able to explain the meaning (semantics) of existing modeling elements and their relationships/connections in respect to the modeling language(s) under consideration. MU 1.13 Learners are able to identify individual model elements in an existing model on the basis of the notation. MU 1.10 Learners are able to interpret and explain the content of a given model.	This type of task is recommended especially at the beginning of the learning process so that students learn to identify different types of model elements, understand their meanings, and distinguish between them.
Model translating	In this type of task, learners are asked to translate the content statements of a given graphical model into another, formal, non-graphical notation (e.g., mathematical notation, path expression; but expect for natural language)	"Please translate the following Petri net into mathematical notation."	MU 1.10 Learners are able to interpret and explain the content of a given model.	This is a more formal and abstract task type in which students must understand a graphical model in order to translate it into another formal notation.
Interpreting model content Variant I: without context	This type of task requires learners to understand the content and semantics of a given model and select appropriate statements about the model or answer specific questions about the model. The answers can be read from the model (i.e., no analyzing activities are required). **Variant I: Context-free (Multiple Choice)** In this variant, a model, where the individual model elements are only labeled with numbers or letters as well as statement about the model content are given. Students must check (and explain) which statement applies to the given model.	"Consider the UML state chart shown above and check whether the following statements are true or false: - The system shown terminates when state H has been exited. - The represented system can be in state G and H at the same time. - When state A is exited, the system is simultaneously in states C and F. - The system can remain in state B indefinitely. - The represented system terminates only when state I has been exited."	MU 1.10 Learners are able to interpret and explain the content of a given model. MU 1.09 Learners are able to explain the meaning (semantics) of existing model elements and their relationships/connections in respect to the modeling language(s) under consideration. MU 1.02 Learners are able to explain and compare concepts of modeling structures or behavior (i.e., static and dynamic aspects).	MU1.09 is considered "optional", as it could be addressed by tasks with constructed response format, in which students have to explain their answer and in doing so they have to explain the meaning of existing model elements. (If not addressed optionally, this competence facet is addressed marginally, because students have to understand the meaning of existing model elements but do not overtly explain it, e.g. in written form). MU1.02 is addressed marginally, as the tasks indirectly and implicitly requires that students to understand the various concepts of modeling structures or behavior and have to distinguish between them.
Interpreting model content Variant II: with context	This type of task requires learners to understand the content and semantics of a given model and select appropriate statements about the model or answer specific questions about the model. The answers can be read from the model (i.e., no analyzing activities are required). **Variant II: Embedded into a real-world scenario** In this variant, the model elements have textual identifiers that refer to a specific context. **a) Multiple-choice questions about the model** In this variant, a graphical model as well as statement about the model content are given. Students must check (and explain) which statement applies to the given model. **b) Explaining the model in natural language** In this case, learners are asked to describe the content of a given model in natural language.	**Variant II a:** "Based on this S/T net, judge whether the following five statements are true or false and indicate your answer with a cross in the corresponding column. - The barriers can be open at the same time - The barriers can be closed at the same time - The net models the states of two barriers - The delivery truck can drive from plant A to plant B as well as back. - If barrier BA is open, then only the journey from plant A to plant B is possible." **Variant II b:** "Describe the following business process modeled in BPMN in natural language."	MU 1.10 Learners are able to interpret and explain the content of a given model. Variant II b) SC 2.01 Learners are able to present models understandable and in a way that is appropriate to the audience (e.g., in relation to the audience's modeling or domain knowledge). SC 2.02 Learners are able to communicate and share knowledge about relevant modeling and domain knowledge and the content of a model. MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario. MU 1.02 Learners are able to explain and compare concepts of modeling structures or behavior (i.e., static and dynamic aspects).	SC2.01 and SC2.02 are addressed optionally, if students are required to explain the model statements in natural language and understandable for a certain audience group (or at least for the lecturer) and in doing so they share their knowledge about model content. MU3.03 is addressed marginally, because it is assumed that when students try to understand to content of the model they implicitly consider to correctness and completeness of the model in relation to the scenario or in relation to their domain knowledge. Additionally, they check if their solution represents the model correctly and completely.
Apply & Transfer				
Problem-solving based on a given model	In this type of task, learners are asked to answer problem-oriented questions based on a given model with reference to the corresponding application domain.	"Problem-solving questions asked for the university-research domain and based on a given model: - A research project that was supposed to be completed last month has not been completed. What reasons can you provide for the delay in completion? Write down as many reasons as you can think of. - A research project is half complete, and all research funds allocated to the project have been used. What should the project leader do? Write as many alternatives as you can think of. (...)" (Bodart et al., 2001: p. 393 figure 3; p. 401 table 8)	MU 2.01 Learners use the information contained in a given model to solve a problem or situation in the corresponding application domain. MU 1.10 Learners are able to interpret and explain the content of a given model. MC 2.04 Learners are able to acquire relevant domain-specific knowledge.	MU1.10 is addressed marginally as students are not explicitly required to explain the model content when solving the task, but they have to understand the model first, before solving problems based on the model. Additionally, to solve a problem in the respective domain students need respective domain knowledge or otherwise need to acquire it and gather relevant information (MC2.04). But normally this is not explicitly required in the task.

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Analyze & Evaluate				
Check formal model properties	In this type of task, learners are asked to check a given model with respect to certain formal properties or conditions. This task type goes beyond "interpreting model content" as it additionally requires to analyze the model based on given criteria. These include checking for model properties (e.g., freedom from cycles, deadlocks, liveness, boundedness), checking the reachability of markings/states under certain conditions, and identifying/stating markings/states or sequences of actions that have certain properties. The given model is not embedded into a real-world scenario (context-free).	"Check which of the following statements are true for the given Petri net: (a) The net is weak and strongly connected. (b) The net satisfies the free-choice property." "Check if place and transition invariants exist and specify them." "Given the following three Petri nets: [...] For each of them, give a reachable marking that has no successor marking. In each case, indicate which transitions have to be fired in order to reach the marking you have specified." "For each of the following properties, specify a complete sequence of actions (i.e., from start node to activity end) of the activity diagram shown that satisfies that property. (a) An action sequence in which [Y] never holds. (b) An action sequence that is as short as possible and contains F but not D. (c) An action sequence with length ≥ 7 in which [Y] holds until E has been executed."	MU 3.01 Learners are able to check a given model in terms of formal properties. MU 1.10 Learners are able to interpret and explain the content of a given model.	MU1.10 is addressed marginally as students are not explicitly required to explain the model content when solving the task, but they have to understand the model to be able to analyze it.
Error finding based on a model	This task class includes task types that require the identification of errors in a given model. The errors should usually be marked in the model and also briefly described.			
Syntactical error finding	In this type of task, a model is given in which learners are to identify syntax errors (i.e., violations of syntactical rules of the modeling language used) and describe or name them. The model elements are usually labeled without context (e.g., without identifiers or labeled only with numbers, letters, and expressions such as "Event 1"/ "Task 1"). The consistency to a model or described scenario is not considered here.	"The following extended event-driven process chain contains modeling errors. Highlight the errors in the diagram." "The EPC shown is not created methodologically correctly. Review the structure of the EPC and explain the modeling errors." "For each of the following diagrams, indicate whether they can be valid class diagrams or not (attributes and methods have been ignored here). In each case, justify your decision (even in positive cases)." "The Coffee Company would like to have flyers printed for their next barista seminar from an online mail order company. The order process of a customer at the online flyer mail order was modeled with BPMN. Mark the modeling errors in the BPMN diagram and name them.	MU 3.03 Learners are able to check the correctness of a given model in terms of the syntax of the modeling language used. MU 1.08 Learners are able to explain syntactical rules of the modeling language(s) under consideration. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use.	MU1.08 is addressed by this task type, if students are required to explain or justify the error, they have identified. We assume that by solving error-finding tasks, students become aware of the relevance of high model quality and, in this case, syntax error-free modeling (VAB1.02).
Error finding based on a model and a corresponding text	In this type of task, a model and a text that describes the scenario that the model is supposed to represent are given. The learners are required to find inconsistencies between the description of the scenario and the model, and usually mark the errors in the model and explain them. This type of task is often combined with the task "model adjusting", in which the incorrect model is to be corrected afterwards.	"The following Class Diagram should conform to the given textual specification. Unfortunately, the diagram contains some errors. Find 5 errors and correct them in the diagram." "During the last meeting with your superior, you were assigned the task to help new employees from another requirements engineering team in their work. Mathew Stark asked you to review a UML class diagram in which he modelled the university's new library system, which is developed in parallel in another project, based on a short description of the data to be stored this system. He provided you on the one hand the description and on the other hand with his diagram. Compare the given class diagram to the original brief system description. You should provide Mathew a detailed description of your observations and a corrected class diagram."	MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario. MU 3.02 Learners are able to check and evaluate the suitability of a given model for the description of a specific scenario and in relation to a specific modeling purpose. MU 1.08 Learners are able to explain syntactical rules of the modeling language(s) under consideration. MU 1.09 Learners are able to explain the meaning (semantics) of existing model elements and their relationships/connections in respect to the modeling language(s) under consideration. MU 1.10 Learners are able to interpret and explain the content of a given model. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. VAB 1.04 Learners are convinced that planned action and a systematic procedures are necessary when solving complex modeling tasks.	While checking the semantic correctness and completeness of the model students implicitly evaluate if the model is suitable for the described scenario (or in how far it is not suitable) (MU 3.02). When describing the inconsistencies students might refer to syntactical rules or the meaning of certain model elements and explain the content of the model (MU1.08, MU 1.09). We assume that by solving error-finding tasks, students become aware of the relevance of high model quality and, in this case, syntax error-free modeling (VAB1.02). In addition, we assume that students realize in more complex tasks of this type that it is useful to proceed systematically when checking the model against the scenario (VAB1.04).
Check pragmatic quality of a model	In this type of task, learners are asked to analyze and evaluate a given model in terms of pragmatic quality. This includes, for example, assessing the unambiguousness of the model as well as the adherence to conventions or guidelines (e.g., in terms of a violation of internal company guidelines). The assessment can either refer to criteria known from the teaching or the criteria are explicitly stated in the task.	"Analyze the following model for an *application for leave* (next page) for the Human Resources department of the Steel Company for deviations from the company's internal convention for process modeling. The corresponding convention guidelines can be found on the intranet. The model is now to be improved according to these guidelines. Conventions (these largely relate to pragmatic quality aspects): 1. Use as few elements as possible in the model 2. Minimize the arcs per element 3. Use one start and one end element 4. Model as structured as possible 5. Avoid mega-models (>50 elements), use refinement and abstraction 6. Label transitions after activity descriptions (preferably verb-object style e.g. "check invoice") 7. Label places after condition describing current state of node 8. Model from left to right 9. Use company design-guidelines described below 10. Avoid crossing arcs"	MU 3.05 Learners are able to check the pragmatic quality (comprehensibility, unambiguity) of a given model. MU 3.09 Learners are able to evaluate a given model in terms of model quality referring to quality criteria. MU 1.07 Learners are able to explain criteria for evaluating model quality (e.g., syntax, semantics, pragmatics according to relevant frameworks such as SEQUAL). VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. MU 1.10 Learners are able to interpret and explain the content of a given model.	MU1.07 is addressed optionally, if students are required to evaluate the "pragmatic quality" of a model and justify their solution, because they have to explain which pragmatic quality aspects should be considered in the analysis. We assume that by solving error-finding tasks, students become aware of the relevance of high model quality and, in this case, the importance of syntax error-free modeling (VAB1.02).

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Error finding based on two corresponding models	In this type of task, learners are asked to identify inconsistencies between two graphical models that represent the same scenario at different levels of abstraction or from different views. Thus, two models are given and errors are built into one of them.	"You are given the following UML 2 Class Diagram. A modeler tried to create a corresponding Object Diagram. Unfortunately, the modeler made some mistakes. Scratch out as many associations and objects of the Object Diagram as possible to make the Object Diagram conform to the Class Diagram." (Brandsteidl et al. 2009) "Can the following petri net B have originated from petri net A by coarsening, restriction, or folding? Justify your answer." "A colleague of yours was given the task to model valid exemplary instantiations of entity and relationship types, according to the multiplicity constraints specified in the class diagram. Since he is a new team member, Mr. Allen assigned you the task to check his results and identify possible mistakes. Furthermore, the project manager advised you to document your observations."	MU 3.08 Learners are able to check models representing different views of the same scenario for consistency with each other. MU 3.01 Learners are able to check a given model in terms of formal properties. MU 1.08 Learners are able to explain syntactical rules of the modeling language(s) under consideration. MU 1.09 Learners are able to explain the meaning (semantics) of existing modeling elements and their relationships/ connections in respect to the modeling language(s) under consideration. MU 1.10 Learners are able to interpret and explain the content of a given model. MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario.	MU3.01 is addressed by tasks in which two models are given which have different formal properties but which are related to each other by a specific transformation. When describing the inconsistencies students might refer to syntactical rules or the meaning of certain model elements and explain the content of the model (MU1.08, MU 1.09, MU 1.10) or at least they need an understanding of these aspects. While checking the consistency between the models, students might also relate and compare the models with a described scenario and thus, check the correctness and completeness of a model in relation to it (MU 3.04).
Check suitability of a model	In this type of task, a model and a described situation or modeling purpose are given. The learners are asked to discuss to what extent the given model is suitable to represent the described scenario or modeling purpose and thus to fulfill the given requirements. Optionally, the model could also be discussed concerning ethical or social aspects, if applicable.		MU 3.02 Learners are able to check and evaluate the suitability of a given model for the description of a specific scenario and in relation to a specific modeling purpose. VAB 3.01 Learners are able to judge a model from an ethical or social point of view. MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. MU 1.10 Learners are able to interpret and explain the content of a given model.	
Compare models	In this type of task, two models are given and learners have to analyze which model better represents a given issue or modeling purpose.		MU 3.06 Learners are able to differentiate given models with regard to their purpose-specific advantages and disadvantages and to judge which model better represents the considered scenario. MU 3.02 Learners are able to check and evaluate the suitability of a given model for the description of a specific scenario and in relation to a specific modeling purpose. MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario. MU 3.09 Learners are able to evaluate a given model in terms of model quality referring to quality criteria. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. MU 1.10 Learners are able to interpret and explain the content of a given model.	MU3.09 is addressed by this task type, if students are required to refer to specific quality criteria while comparing the models.
Peer feedback	Learners should examine and evaluate each other's models and discuss them with each other. This type of task is actually for formative purposes in the context of exercise courses or seminars.		SC 2.07 Learners are able to check and constructively critique models or model parts of others and accept constructive criticism from others. SC 2.02 Learners are able to communicate and share knowledge about relevant modeling and domain knowledge and the content of a model. MU 3.02 Learners are able to check and evaluate the suitability of a given model for the description of a specific scenario and in relation to a specific modeling purpose. MU 3.03 Learners are able to check the correctness of a given model in terms of the syntax of the modeling language used. MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario. MU 3.05 Learners are able to check the pragmatic quality (comprehensibility, unambiguity) of a given model. MU 3.06 Learners are able to differentiate given models with regard to their purpose-specific advantages and disadvantages and to judge which model better represents the considered scenario. MU 3.01 Learners are able to check a given model in terms of formal properties. MU 3.09 Learners are able to evaluate a given model in terms of model quality referring to quality criteria. MB 3.04 Learners are able to reflect on and judge the suitability of a model they have created to represent a specific scenario. MB 3.05 Learners are able to evaluate and justify their design decisions for a model they have created themselves. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. MU 1.10 Learners are able to interpret and explain the content of a given model.	

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Model building and modifying				
Apply & Transfer				
Exemplary model building	These task classes comprise task types in which learners have to create small, exemplary models independent of a specific scenario or context. This is intended in particular to illustrate or represent properties, rules, or concepts.			
Applying a modeling language	Learners are asked to create a small graphical model to illustrate the modeling language or create plausible, exemplary model elements or model parts based on domain knowledge (i.e., without a given scenario description).	"Which UML diagram type describes the sequence of actions? Draw a small but meaningful example?" "Your new colleague asks you about the modeling languages BPMN and Petri nets. Explain to him the similarities and differences using an exemplary process for each modeling language." "Given the following class diagram (as part of modeling a collaborative graphics program): [...] For each of the classes shown, create a corresponding object as an instance of that very class."	MB 2.04 Learners are able to correctly apply syntactical rules of the modeling language/s. MB 2.03 Learners are able to apply techniques of modeling structures and/or behaviors. MB 2.09 Learners are able to apply conceptual knowledge of modeling using the respective modeling language (e.g., UML, ER, Petri nets, EPC) to build models. MU 1.03 Learners are able to explain the strengths and weaknesses of specific modeling languages or types of models for a specific purpose. MB 2.01 Learners are able to apply and use modeling tools. MU 1.02 Learners are able to explain and compare concepts of modeling structures or behavior (i.e., static and dynamic aspects). SC 2.01 Learners are able to present models understandable and in a way that is appropriate to the audience (e.g., in relation to the audience's modeling or domain knowledge). MB 3.04 Learners are able to reflect on and judge the suitability of a model they have created to represent a specific scenario. MB 2.06 Learners are able to use a modeling language in a way that serves a certain purpose or solves a problem at hand.	
Model building based on (formal) properties or criteria	In this task type learners are required to create a graphical model that has certain formal properties or meets certain criteria and conditions. These include formal model properties (e.g., freedom from cycles, liveness) or formal notations (e.g., mathematical notation of a Petri net, path expression of an activity diagram). Usually, the model does not refer to a context and model building is not based on a given graphical model.	"Give an example of a Petri net in graphical form with the following properties and justify: An unbounded Petri net that contains a deadlock." "ER modeling: for each relationship type (1:1, 1:n, n:m), consider an example with a corresponding relationship attribute." "Translate the following Petri net given in mathematical notation into graphical representation." "Create a UML activity diagram that describes exactly the same set of sequences as the path expression given above."	MB 2.03 Learners are able to apply techniques of modeling structures and/or behaviors. MB 2.04 Learners are able to correctly apply syntactical rules of the modeling language/s. MB 2.01 Learners are able to apply and use modeling tools. MU 1.12 Learners are able to explain formal properties of models (e.g., properties of petri nets). MB 3.04 Learners are able to reflect on and judge the suitability of a model they have created to represent a specific scenario.	MU1.12 is addressed optionally, if the task requires an explanation. If not, the competence facet is addressed marginally as students need to know and understand the respective formal properties but do not need to explain them in written form/ overtly.

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Model modifying	This task class comprises task types in which a model or individual model elements are given, which must be adapted or supplemented, e.g., in order to represent additional information from a described scenario, to fulfill certain properties or to correct errors. Often, only the extension, i.e. the addition of model elements to the given model is required, but the removal or adaptation of existing model elements may also be necessary.			
Model completing	In this type of task, a scenario and some model elements or a partial model are already given. The learners have to add the missing model elements either on the basis of a scenario description or on the basis of their own plausible assumptions. In contrast to the task type "model adjusting" only single model elements are added here. For instance the relations/connections between model elements (e.g. by adding edges/arrows/arrowheads, cardinalities, multiplicities) or identifiers have to be completed or specified. These tasks are more like a cloze.	"Complete the following class diagram by completing the lines to inheritance, association, aggregation, or composition."	MB 2.08 Learners are able to adapt or further develop a given model due to errors, inconsistencies, incompleteness, or new requirements and add, modify, or remove model elements accordingly. MB 2.03 Learners are able to apply techniques of modeling structures and/or behaviors. MB 2.04 Learners are able to correctly apply syntactical rules of the modeling language/s. MB 2.05 Learners are able to derive identifiers for model elements from a given problem (task and scenario) and use them consistently. MB 2.01 Learners are able to apply and use modeling tools. MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario. MB 2.07 Learners are able to select and consistently create/complement corresponding views of a system or scenario with matching models. MB 1.05 Learners are able to explain the effects of certain modifications to a model. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. MU 3.09 Learners are able to evaluate a given model in terms of model quality referring to quality criteria. MU 3.01 Learners are able to check a given model in terms of formal properties. MU 1.10 Learners are able to interpret and explain the content of a given model.	MU3.04 refers to tasks in which the model has to be completed based on a described scenario. That means, before adding missing elements, students have to check the completeness of the given model parts to identify which elements are missing. MB2.07 refers to tasks in which a model has to be completed based on another graphical model. MB1.05 is considered as marginally addressed, because students usually do not need to overtly explain the effects of the modifications, but they have to understand and anticipate them when completing the model. We assume that by solving error-finding and modifying tasks, students become aware of the relevance of high model quality (VAB1.02).
Model adjusting	In this task type, learners are required to modify a given (or previously created) model, e.g., to correct errors, meet new requirements, or other formal properties. This task type may thus require the addition or removal of model elements/parts. This task type is more complex than "model completing", because here not only single missing elements are added, but also whole model parts have to be modeled correctly and integrated into the given model or the whole model has to be transformed to fulfill the formal properties.	New requirements: "Your project manager Mr. Allen had a second discussion with the university's teaching representative Mr. Wayne to get a better understanding of the situation. In this meeting Mr. Allen elicited new requirements, which can be found in the meeting minutes below. He assigned you the task to expand your class diagram from task 1A to include all the additional information given by Mr. Wayne, including appropriate generalization/specialization relationships." Formal properties: "Please modify the given Petri net as slightly as possible so that it is bounded and has no deadlock." "Model the following PR/T net as a B/E net with the same information content."; "Convert the following Petri net with capacities into a conventional Petri net without capacities."; "Given the following UML state diagram: [...] Specify an associated flat state diagram (that is, a state diagram that no longer contains hierarchical states or region states)." Correct errors: "Compare the given class diagramm to the original brief system description. You should provide Matthew (a detailed description of your observations) and a corrected class diagramm."	MB 2.08 Learners are able to adapt or further develop a given model due to errors, inconsistencies, incompleteness, or new requirements and add, modify, or remove model elements accordingly. MB 2.07 Learners are able to select and consistently create/complement corresponding views of a system or scenario with matching models. MB 2.03 Learners are able to apply techniques of modeling structures and/or behaviors. MB 2.04 Learners are able to correctly apply syntactical rules of the modeling language/s. MB 2.05 Learners are able to derive identifiers for model elements from a given problem (task and scenario) and use them consistently. MU 3.04 Learners are able to check the semantic correctness and completeness of a given model in relation to the considered scenario. MB 2.01 Learners are able to apply and use modeling tools. MU 3.03 Learners are able to check the correctness of a given model in terms of the syntax of the modeling language used. MU 3.01 Learners are able to check a given model in terms of formal properties. MU 3.09 Learners are able to evaluate a given model in terms of model quality referring to quality criteria. MB 1.05 Learners are able to explain the effects of certain modifications to a model. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. MU 1.10 Learners are able to interpret and explain the content of a given model.	Before starting to adjust the given model students have to thoroughly analyze the given model primarily in terms of sematic correctness and completeness (MU3.04) but also indirectly in terms of syntax (MU3.03), formal properties (MU3.01), and other quality criteria (MU3.09). MB1.05 is considered as marginally addressed, because students usually do not need to overtly explain the effects of the modifications, but they have to understand and anticipate them when adjusting the model. We assume that by solving error-finding and modifying tasks, students become aware of the relevance of high model quality (VAB1.02).
		Analyze & Evaluate		
Evaluate model building	In this task type, learners are asked to evaluate a self-created model with respect to certain criteria and/or justify design decisions. This task type is an optional addition to a "model building" task.	"Please evaluate the model you have created with respect to the following quality criteria: (...)" "Please evaluate to what extent the model you have created represents the described scenario [completely, understandably...]."	MB 3.04 Learners are able to reflect on and judge the suitability of a model they have created to represent a specific scenario. MB 3.05 Learners are able to evaluate and justify their design decisions for a model they have created themselves. VAB 2.01 Learners anticipate and are able to describe the possible consequences of using the models they create (impact assessment). VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use.	

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
		Create		
Model building	This task class includes all task types in which learners are expected to create graphical models on their own from scratch.			
Model building based on a text describing a scenario	The learners are required to create a graphical model based on a scenario described in natural language. The degree of difficulty of this task type depends on the specific design of the task: - Complexity of the scenario, i.e. the size of the model to be created (in terms of the number of model elements to be created) - Number and type of cues included in the task: e.g. hints on types of model elements to be considered, cues on specific model elements to be considered (i.e. Support in identifying types of model elements in the scenario), guidance for modeling by structuring the scenario in a certain way (bullet points, paragraphs) or hints and advice on procedural knowledge by structuring the task into subtasks (e.g., first identify types of model elements in the text, then model them; first model classes, then attributes). The tasks of this task type also differ in the form of representation of the described scenario. Mainly this is described in a pure prose text form. Possible variants are e.g. text in interview form, in document form (delivery bill, production order), as database structure or requirement specifications in natural language. The context or domain of the described scenario can be fictitious, from the learners' everyday life or from a practice-relevant application domain. In addition, the difficulty of the task can be increased, for instance, by requiring learners: - to choose an appropriate modeling language themselves - to describe and explain their design decisions - to research further information about the domain in question that is helpful for modeling	Hints on types of model elements / Procedural cues: "Create an ER diagram for the following scenario. [...] Note: Keep the number of entities used low, as far as this makes sense. Also, add attributes that are not explicitly mentioned but seem necessary to you." "Please model the situation described below as an EPC. In particular, consider organizational units, documents, and IT systems." "Please compile a graphical process model with only functions and events using the EPC." "Please model the classes and their relationships including their methods, attributes, and cardinalities in a UML class diagram." "The underlined words should be the classes in your diagram. Please identify attributes and methods yourself. Use appropriate associations, aggregations, and compositions when possible. In particular, consider when it makes sense to use an association, aggregation, or composition instead of an attribute. Also specify the respective multiplicities for corresponding relationships." "a) Identify the components mentioned in the description. b) Identify the interfaces of the components. c) Create a component diagram from their identified components and interfaces." Modeling tool: Please model the descibed scenario as a petri net using the Horus Business Modeler. Selection of a suitable modeling language: "Please model the relevant aspects of the business process using appropriate modeling languages." Justification of design decisions: "Justify your diagram in detail. In particular, be specific about why certain classes are or are not subclasses of other classes."; "For each generalization group, specify whether it is complete or incomplete, and whether it is disjoint or overlapping. Justify your decision in each case."	MB 4.01 Learners are able to create graphical models (such as UML diagrams, ER models, and Petri nets) themselves to represent a scenario. MB 4.02 Learners are able to create a model that is semantically correct and complete with respect to a scenario, and limit themselves to relevant model content (conciseness). MB 4.04 Learners are able to develop appropriate and consistent identifiers for model elements. MB 4.05 Learners are able to select an appropriate level of abstraction in relation to the modeling purpose when creating a model and maintain it consistently within the model. MB 2.04 Learners are able to correctly apply syntactical rules of the modeling language/s. MB 2.01 Learners are able to apply and use modeling tools. MB 3.02 Learners are able to check, evaluate, and select modeling languages or model types for their suitability for a specific application domain and modeling purpose. MB 3.05 Learners are able to evaluate and justify their design decisions for a model they have created themselves. MB 3.04 Learners are able to reflect on and judge the suitability of a model they have created to represent a specific scenario. MB 4.03 Learners are able to create understandable and readable models based on known guidelines or conventions. MB 4.06 Learners are able to create a model in a way that is target group-specific, i.e., understandable to a specific group of people. MC 2.04 Learners are able to acquire relevant domain-specific knowledge. MB 3.01 Learners are able to derive relevant information and requirements (e.g., modeling elements, relationships, etc.) from a problem (task and scenario) and thus structure the problem.	MB3.04, MB4.03, MB4.06, MC2.04 were classified as marginal because these aspects are not normally expected of students directly in the task and are often not included in the grading. Nevertheless, the competence facets are implicitly addressed since students have learned that they should verify the suitability of their model when making the model and make the model readable and understandable at least for the lecturer. In addition in formative settings students have to opportunity to acquire domain knowledge if need for a better understanding of the scenario. However, these competence facets could be addressed by a variation of the task and by explicitly requiring students to do so. MB3.01 is considered an important competence facets when building a model, but it does not become directly evident in the task solution and is difficult to measure or grade it.
	Variant II: Development of analysis questions This task variant requires students to develop formal analysis questions from the task text that can be answered using the model or that define the correctness of the model prior to model building.	"Specify the board game "Checkers". A sufficiently detailed description of the game can be found e.g. at Wikipedia. It is sufficient to consider one of the rule variants described there. To do this, first model or implement the game in a suitable way for model checking. Check for observance of the rules as well as the following property: is it possible to end the game without hitting a single stone?" "Consider the behavior of an elevator in a house with 4 floors. The elevator can be requested each floor and must serve all requests. It makes no difference if the elevator is requested from inside or outside for a floor. Specify the elevator and the floors with two templates in UPPAAL and verify the following properties: - The system never enters a deadlock. - The doors can only open on the floor where the elevator is currently located. - If the elevator is requested on a floor, it will reach it. "Specify the traffic light circuit from Task 1.2 in UPPAAL. Specify properties that prevent accidents at the intersection."	MB 2.11 Learners are able to translate general, abstract problems and objectives into concrete specifications and analysis questions. MU 1.12 Learners are able to explain formal properties of models (e.g., properties of petri nets).	
Model building based on another graphical model	Learners are required to create a graphical model based on a given model in another modeling language (e.g. UML activity diagram to petri net) or convert it to another diagram type (e.g. UML class diagramm to object diagram or sequence diagram) for instance to depict a different view of the scenario. Variant I: context-free Variant II: embedded into a context, i.e. an additional scenario is described that has to be taken into account.	Variant I: "Convert the following UML activity diagram into a Petri net. In doing so, label corresponding places and transitions of the Petri net with the names of the object nodes and actions they represent." Variant II: "Consider the following class diagram […]. Create a corresponding object diagram that describes the following scenario: [...]." "Create a UML object diagram that matches your class diagram of subtask (a) and models the above situation."	MB 4.01 Learners are able to create graphical models (such as UML diagrams, ER models, and Petri nets) themselves to represent a scenario. MB 2.07 Learners are able to select and consistently create/complement corresponding views of a system or scenario with matching models. MB 2.03 Learners are able to apply techniques of modeling structures and/or behaviors. MB 2.04 Learners are able to correctly apply syntactical rules of the modeling language/s. MB 2.06 Learners are able to use a modeling language in a way that serves a certain purpose or solves a problem at hand. MB 2.09 Learners are able to apply conceptual knowledge of modeling using the respective modeling language (e.g., UML, ER, Petri nets, EPC) to build models.. MB 4.02 Learners are able to create a model that is semantically correct and complete with respect to a scenario, and limit themselves to relevant model content (conciseness). MB 2.01 Learners are able to apply and use modeling tools. MU 3.08 Learners are able to check models representing different views of the same scenario for consistency with each other. MB 3.04 Learners are able to reflect on and judge the suitability of a model they have created to represent a specific scenario. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use.	

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Case study	A case study is a very complex task of the type "model building", which is embedded in an extensive, described real-world scenario. Usually, students have to create more than one model (e.g., to represent different views or aspects of an scenario). In addition to the pure model building, further tasks have to be completed, depending on the task design. Normally students should use a modeling tool to create the models. Depending on the complexity of the case study, it may take up to one semester to complete. Possible design options: - Working alone or in small groups (SC) - Communication and coordination with "externals" (teams, organization) required (SC) --> interviewing, background documentation, or negotiations might be required - Integration of different use cases of different disciplines of graphical modeling (MB 2.10) - Performing activities from requirements elicitation to model implementation (e.g., for process improvement, code generation) (MC) - Answering theoretical questions Different forms of submission may be required: - Submission of the created models - plus written documentation - plus (oral) presentation of the results / the models		MB 4.01 Learners are able to create graphical models (such as UML diagrams, ER models, and Petri nets) themselves to represent a scenario. MB 4.02 Learners are able to create a model that is semantically correct and complete with respect to a scenario, and limit themselves to relevant model content (conciseness). MB 4.04 Learners are able to develop appropriate and consistent identifiers for model elements. MB 4.05 Learners are able to select an appropriate level of abstraction in relation to the modeling purpose when creating a model and maintain it consistently within the model. MB 2.04 Learners are able to correctly apply syntactical rules of the modeling language/s. MB 2.01 Learners are able to apply and use modeling tools. MB 3.02 Learners are able to check, evaluate, and select modeling languages or model types for their suitability for a specific application domain and modeling purpose. MB 3.05 Learners are able to evaluate and justify their design decisions for a model they have created themselves. MB 2.07 Learners are able to select and consistently create/complement corresponding views of a system or scenario with matching models. MU 2.01 Learners use the information contained in a given model to solve a problem or situation in the corresponding application domain. SC 2.01 Learners are able to present models understandable and in a way that is appropriate to the audience (e.g., in relation to the audience's modeling or domain knowledge). SC 2.03 Learners are able to make and keep agreements (e.g., regarding task distribution, chosen level of abstraction) in a team when working on complex modeling tasks. SC 2.05 Learners are able to accept and take up ideas from other team members when building models. SC 2.07 Learners are able to check and constructively critique models or model parts of others and accept constructive criticism from others. SC 2.04 Learners are able to inquire about requirements for the model and the relevant domain knowledge from clients or other stakeholders (possibly also from persons not familiar with computer science) about requirements for the model and the domain knowledge needed, and involve them in the modeling process. MB 4.06 Learners are able to create a model in a way that is target group-specific, i.e., understandable to a specific group of people. MC 4.01 The learners are willing to take new - previously unknown - approaches when working on complex modeling tasks and thus to distance themselves from the familiar. MC 4.02 Learners are able to find strategic and creative answers when searching for solutions to well-defined, concrete and abstract problems. VAB 3.01 Learners are able to judge a model from an ethical or social point of view. VAB 2.01 Learners anticipate and are able to describe the possible consequences of using the models they create (impact assessment). MU 1.05 Learners are able to explain possible modeling purposes in terms of the intended use or implementation of models (e.g., for code generation, simulation, or organizational design). MB 1.04 Learners are able to explain the relevance of the modeling purpose for model building (especially for the selection and use of different modeling techniques). MU 1.01 Learners are able to explain basic key terms and concepts of modeling (such as model, principles of model building, definition and modification of models, metamodels). MU 1.02 Learners are able to explain and compare concepts of modeling structures or behavior (i.e., static and dynamic aspects). MU 1.11 Learners are able to explain the possibilities and limitations of graphical modeling. VAB 1.01 Learners are able to explain the objectives and relevance of graphical modeling for the respective area of modeling in computer science. MB 1.05 Learners are able to explain the effects of certain modifications to a model. SC 2.08 Learners are able to divide complex modeling tasks into subtasks and structure them as well as organize and coordinate the completion of subtasks by different team members or teams. SC 2.06 Learners are able to put themselves in the role of others (e.g., users, software developers, clients) and change their own perspective. MC 3.02 Learners reflect on their problem solutions and are able to learn independently from their mistakes. MC 2.01 Learners are able to adapt and extend their own skills and knowledge in the field of graphical modeling according to changing situational requirements through independent learning. MC 2.04 Learners are able to acquire relevant domain-specific knowledge. VAB 2.02 Learners develop high intrinsic motivation for modeling and interest in its technical innovations and development. VAB 2.03 Learners are willing to take on demanding modeling challenges. MB 3.01 Learners are able to derive relevant information and requirements (e.g., modeling elements, relationships, etc.) from a problem (task and scenario) and thus structure the problem. MB 3.04 Learners are able to reflect on and judge the suitability of a model they have created to represent a specific scenario. MB 4.03 Learners are able to create understandable and readable models based on known guidelines or conventions. MC 2.03 Learners are able to exert themselves and persevere when working on complex modeling tasks. MC 3.01 Learners are able to analyze and consciously select problem-solving strategies according to the respective context with regard to their appropriateness and efficiency when working on modeling tasks. VAB 1.02 Learners understand the relevance of high model quality (in terms of syntax, semantics, and pragmatics) for model understandability and subsequent model use. VAB 1.04 Learners are convinced that planned action and a systematic procedures are necessary when solving complex modeling tasks.	

Task class and type	Description	Examples	Competence facets central (product-related) / optional / marginal (process-realted)	Further remarks
Transversal knowledge and skills				
Apply & Transfer				
Reflection task	The learners are required to reflect on their problem solving process when working on more complex modeling tasks or on their learning process. This type of task can be used e.g. as a short reflection task after a more complex task or in the form of a more extensive portfolio / learning journal, e.g. accompanying a case study work. The tasks should also include questions directed towards the future.	"Please reflect on how you went about accomplishing this task or learning this knowledge area." "What did you do well and what would you do differently next time?" "What helped you accomplish the task and what was more of a hindrance?"" "What steps do you plan to take next to achieve your learning goals?"	MC 3.02 Learners reflect on their problem solutions and are able to learn independently from their mistakes. MC 2.02 Learners are able to control and organize their own learning process and development in the field of graphical modeling. MC 3.03 Learners reflect on and evaluate their own level of knowledge and skills related to graphical modeling. MC 3.01 Learners are able to analyze and consciously select problem-solving strategies according to the respective context with regard to their appropriateness and efficiency when working on modeling tasks.	
Role play	This task type could serve to simulate typical conversations that may arise in practice when working on modeling tasks (e. g. interviews in the context of requirements engineering). The learners should take on the roles of the various stakeholders involved (including people from outside IT). This type of task is actually for formative purposes in the context of exercise courses or seminars.		SC 2.06 Learners are able to put themselves in the role of others (e.g., users, software developers, clients) and change their own perspective. SC 2.02 Learners are able to communicate and share knowledge about relevant modeling and domain knowledge and the content of a model.	

* Red color indicates task types and variants added after the gap analysis.