Graph-based Analysis of Computer Science Curricula for Primary Education

Stefan Pasterk
Institute for Informatics Didactics
Alpen-Adria-Universität Klagenfurt
9020 Klagenfurt, Austria
Email: stefan.pasterk@aau.at

Andreas Bollin
Institute for Informatics Didactics
Alpen-Adria-Universität Klagenfurt
9020 Klagenfurt, Austria
Email: andreas.bollin@aau.at

***Abstract*—Because of the present deep impact of information technology on society, school subjects that deal with topics of computer science or digital literacy gain importance nowadays. Some countries start to teach related topics in primary education and even in kindergarten. The underlying curricula, educational standards and/or competency models have already been developed and established and differ in a lot of points. Because of these differences, a comparison is a complex task. In this paper a graph-based approach is applied to introduce a framework for comprehensibly evaluating the different curricula, standards and competency models and to demonstrate its use by analyzing and comparing six existing curricula, standards and competency models. Our approach maps the content of curricula and standards to a directed graph by connecting their knowledge items with each other via dependency relations. This method enables a formalized comparison using graph theoretical metrics like the highest degrees, numbers of sources and sinks, or the connectivity. The representation is mapped to a graph database, allowing for further analysis of the content and preparing the ground for teachers and curriculum-developers to individually form a computer science curriculum in primary schools.**

I. Introduction

The deep impact of information technology (and with it informatics/computer science) not only affects our everydays' life, it also highly affects the education sector. Thus, computer science can be found in different curricula around the world, up to now mainly in secondary education. In addition to that, more and more countries start to teach simple or simplified topics of computer science in kindergarten and primary schools, too. For this purpose, curricula, educational standards and/or competency models were developed and in some countries like England [1], Switzerland [2] or Australia [3] already established. Some of these approaches are very extensive, while others focus on particular concepts of computer science only. Further they differ in aspects like structure, formulation, and detailing. Because of the obvious differences, a comparison of the approaches is a complex task and the main objective of this contribution is (a) to introduce a technique and framework for comprehensibly evaluating the different curricula, standards and competency models, and (b) to prepare the ground for individually forming a computer science curriculum in primary schools.

Although a lot of different curricula, standards, and competency models for computer science in primary education have already been analyzed and compared, there exist no analysis of structure, complexity, or central knowledge items. A promising model, which is applied in the analysis of curricula for higher education, represents curricula as directed graphs and uses graph theory to compare them [4], [5].

To introduce our approach, this paper gives answers to the following main research questions:

1) Can a graph-based representation form be used to show the main focus of a curriculum?
2) Can a graph-based representation form be used to determine the complexity or central knowledge items of a curriculum?
3) Does a graph-based representation form reveal structural problems?

In order to answer these questions, in our approach basic competencies and knowledge items are annotated by labels (e.g. age group, ACM knowledge area, and so on) and then mapped to a directed graph by connecting them with each other via dependency relations, as it is briefly described by Pasterk and Bollin [6]. The elements of the curricula or standards are displayed as the vertices. Dependencies between these elements are then mapped to labeled edges, resulting in a directed graph. At the end, the representation is then mapped to a graph database, allowing for further analysis (comparison, learning path selection etc.) of the content. Based on the resulting graph database a platform will be developed, that allows experts to evaluate elements of a curriculum e.g. by their importance and complexity, and users like teachers to select knowledge items defined in existing curricula for their own learning paths.

This paper is structured as follows. After a motivation and an overview of related work in the first two sections, the selected curricula, educational standards, and competency models are described. Section 4 introduces the notion of a graph-based and generic model of competencies. In section 5, the comparison process is summarized and first results are presented. The paper concludes with a section for discussion and future work.

II. Related Work

During the last years the research interest for computer science in primary education rose. This fact is reflected by the

978-1-5090-5920-1/17/$31.00

number of published literature concerning this field of study. A lot of these articles or books focus on one specific new developed curriculum and explain it in detail like for instance Berry does it with the curriculum for the subject "Computing" in England [7]. But there exist only a few analysis and comparisons of different curricula and standard-models for computer science in primary education. One of them is from Barendsen et al. [8]. Their article focuses on computer science concepts in K-9 education and considers curricula from England, Italy and the United States. For the analysis the content of these documents is grouped into knowledge categories and matched to knowledge areas of the ACM/IEEE Computer Science Curriculum [9]. Further tasks from the Bebras international competition are analyzed, to identify the assessed concepts and the assessment methods [8]. To design a primary school curriculum for computer science and programming, Duncan and Bell [10] also compare different curricula. There, the main English-language curricula for the primary school level [10], the CSTA K-12 Copmuter Science Standards [11], the UK computing curriculum [12], and the Australian "Digital Technologies" curriculum [13] are considered. To identify key ideas they define following six categories of themes, based on themes of the chosen curricula: algorithms, programming, data representation, digital devices and infrastructure, digital applications, and humans and computers. The allocation of the curriculum-elements to these themes shows differences and also similarities of the chosen curricula [10].

Different approaches to analyze and compare curricula are to be found by extending the area of interest. Most of these approaches focus on academic courses and undergraduate degree programs. In the field of undergraduate degree programs Pedroni, Oriol, and Meyer [14] present a framework to describe and compare curricula. It is based on relations between knowledge units of two levels of granularity: the so called "Trucs (Testable, Reusable Unit of Cognition)" represent a set of concepts and skills connected to an idea, and the "notions" represent single concepts [14]. There are also three types of relations defined: notions can be related by "is a" and "requires", whereas Trucs can have a "dependency" to other Trucs. In another approach Sekiya, Matsuda, and Yamaguchi [15] generate a map of the syllabi of a curriculum to analyze the structure and to get an holistic understanding. Lightfoot [4] describes a graph-theoretic approach with a focus on the improvement of the structure and the placement of assessment within a curriculum. For that purpose, the courses of a curriculum correspond to vectors, and prerequisite requirements display the edges of a simple acyclic directed graph. Basic graph metrics like structural centrality, clustering density, and degree are used to analyze a curriculum and to determine a good placement of topic introduction and assessment [4]. Another graph-theoretic approach to compare core aspects of curricula is presented by Marshall [5] by comparing three undergraduate degree computer science curricula including the ACM/IEEE Computer Science Curriculum [9]. It represents topics, knowledge areas, knowledge units or modules as vertices and the dependencies between these vertices as the edges of a graph. The comparison of the resulting directed graphs (digraphs) of the different curricula is conducted visually and algorithmically.

The graph-based approach of this contribution can be compared to those of Lightfoot [4] and Marshall [5], but it focuses on curricula, educational standards, and competency models for computer science in primary education. Further the resulting graphs are mapped to a graph-database, to offer the possibility of gaining information of a curriculum by simple queries. The graph-database also enables the integration of all selected curricula into one big graph collecting all competencies and knowledge items.

In the next section the curricula, educational standards, and competency models which were selected to be analyzed during this project, will be described.

III. Standards and Curricula

There exist a lot of different curricula, educational standards and competency models for computer science in secondary education. Related to organizational circumstances, learning goals, topics, and teaching methods they differ in several points [16].

Computer science in primary education as part of K-12 education was already considered in the first ACM model curriculum for K-12 computer science from 2003 [17], [8]. It differed between the four levels (1) Foundations of Computer Science, (2) Computers Science in the Modern World, (3) Computers Science as Analysis and Design, and (4) Topics in Computer Science. The first level lasted from kindergarten to grade eight, which includes primary education and contains, besides the use of technologies for learning, topics like binary numbers, algorithms, and fundamental logic [17]. Since 2003, this model curriculum has been revised and new curricula or educational standards for computer science education in primary and secondary schools have been developed.

Following a report from European Schoolnet [18], several European countries already introduced computer science in their national curriculum for primary education, including Spain, France, and the United Kingdom (UK). Also, in Australia the national curriculum contains a subject called "Digital Technologies" [13].

This contribution focuses on the analysis and comparison of the following six national and important international curricula, educational standards, and competency models:

- the CSTA K-12 Computer Science Standards [11],
- the UK computing curriculum following the detailed description of Berry from CAS (Computing at School) [7],
- the Australian Digital Technologies curriculum [3],
- the Austrian competency model for digital competencies and computer science education called digi.komp4 [19],
- the new curriculum in Switzerland called "Lehrplan 21" [2],
- the educational standards for computer science in primary education by the GI (the German Informatics Society) [20] in Germany, which is still under development.

The authors perceive the detailed description of each of the selected curricula, educational standards, and competency models as an important part of the analysis process, which is therefore provided in the following part of the contribution. Some of the quoted example elements had to be translated from German into English language by the authors.

a) CSTA K12 computer science standards: In 2011 the CSTA presented the "CSTA K-12 Computer Science Standards" [11] which are often referenced in relevant literature [8], [10], [16]. The CSTA K-12 Computer Science Standards are divided into the three levels (1) Computer Science and Me, (2) Computer Science and Community, and (3) Computer Science in the Modern World, Computer Science Concepts and Practices, Topics in Computer Science. Like the ACM model curriculum they start at the kindergarten and last till the twelfth grade. The standards in the levels K to 6 correspond to the age range from five to twelve years. Those levels are again divided into Level K-3 and 3-6 standards. Overall, the levels K to 6 include 45 standards, 16 for levels K-3 and 29 for levels 3-6. Further the standards are categorized into the following five strands, containing the corresponding number of standards for the levels K to 6:

- Collaboration: 5 standards
- Computational Thinking: 11 standards
- Computing Practice and Programming: 16 standards
- Computers and Communications Devices: 7 standards
- Community, Global, and Ethical Impacts: 6 standards [11]

The level K-3 represents the age range from five to seven years what will be displayed by level 1 in the further discussion of this contribution. Level 2 will cover the ages from eight to twelve years what corresponds to level 3-6 of the CSTA K-12 Computer Science Standards.

b) Curriculum in England - Computing At School (CAS): The "Computing At School" curriculum was introduced in 2013 and established in 2014 in the National Curriculum. It provides content for the new subject "Computing" replacing "Information and Communication Technology (ICT)", which was often related to the use of technology instead of learning the concepts of computer science or the creation of software [12]. Since this new curriculum computer science is in England taught compulsory from age five to sixteen. The curriculum is divided into four key stages which correspond to levels of the CSTA standards. The first two key stages cover the age range from five to eleven years. Overall, the program of study for this curriculum contains 16 statements, six in key stage 1 and ten in key stage 2. What has to be mentioned is, that some of these statements contain more than one knowledge item. This can be shown by the following example statement from key stage 1:

- *"Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions."* [7]

The whole statement can, for example, be divided into smaller knowledge items by using the given semicolons like

- *"Understand what algorithms are."*
- *"Understand how algorithms are implemented as programs on digital devices."*
- *"Understand that programs execute by following precise and unambiguous instructions."*

If all of the statements would be divided into smaller knowledge items, the curriculum would of course have far more items than the 16 original statements. We are aware of this fact, but in this contribution we wanted to consider the statements in their original versions and therefore a knowledge item refers to a whole statement.

The original statements can be categorized into the following three sub-sections, containing the corresponding number of statements for the key stages one and two:

- Computer Science (CS): 8 statements
- Information Technology (IT): 3 statements
- Digital Literacy (DL): 5 statements [7]

The key stage 1 statements from this curriculum cover the age range from five to seven years and will correspond in further discussion to level 1. Key stage 2 represents statements for students that are eight to eleven years old, and will correspond to level 2.

c) Australian Curriculum (AC): In Australia, the curriculum for the new learning area "Technologies" was presented in 2013 and represented a combination of the two "distinct but related" subjects "Design and Technologies" and "Digital Technologies" [13]. Both subjects start with the first school year called Foundation (F) and end at the tenth grade as an elective subject. They are divided into levels that represent two school grades. So the levels F-2, 3-4, and 5-6 cover primary school what means the age range from five to twelve years. For "Design and Technologies" the main topics are the impact of technologies on society and related design topics, whereas "Digital Technologies" focuses on the background and the use of information technology. Therefore only the curriculum for "Digital Technologies" will be discussed in this contribution. Overall, the levels F-6 contain 22 learning objectives, six of them in the F-2 level, seven in level 3-4, and nine in level 5-6. It is divided into the two strands "Knowledge and understanding" and "Processes and production skills". Both strands are further divided into the following sub-strands, containing the corresponding number of learning objectives:

- Digital systems: 3 learning objectives
- Representation of data: 3 learning objectives
- Collecting, managing and analyzing data: 3 learning objectives
- Creating digital solutions by investigating and defining: 3 learning objectives
- Creating digital solutions by generating and designing: 2 learning objectives
- Creating digital solutions by producing and implementing: 2 learning objectives

- Creating digital solutions by evaluating: 3 learning objectives
- Creating digital solutions by collaborating and managing: 3 learning objectives

The level F-2 covers the ages five to seven and will correspond to level 1 in this discussion. The years 3-4 and 5-6 represent the age range from eight to eleven and will be merged to level 2.

d) Competency model in Austria (digikomp4): In Austria computer science or informatics is not a subject in primary education on its own. For this reason there exists no national curriculum, but suggestions for interested teachers in form of a competency model presented in 2013 by Mulley and Zuliani [21]. The model focuses on digital literacy but also considers some basic concepts of computer science. Overall, it contains 49 competencies, but is not divided into different levels. It covers the four years of primary school with the age range from six to ten years. The following four major topics or strands are included in the model, containing the corresponding number of competencies for each strand:

- Informationtechnology, humans and society: 16 competencies
- Informaticssystems - Usage of digital devices and networks: 13 competencies
- Applications - Digital tools in everyday life: 15 competencies
- Informaticsconcepts - First steps in informatics: 5 competencies

In the tables I, II, and III the digikomp4 competency model will be shortened to ”dk”.

e) Standards from German Informatics Society (GI): In Germany the ”Gesellschaft für Informatik (GI)” (the German Informatics Society) already developed and published educational standards for informatics in lower secondary eduction in 2008 [22] and also for higher secondary education in 2016 [23]. A group of experts is actually working on standards for informatics in primary education called ”Bildungsstandards Informatik für den Primarbereich” [20]. Since these standards are still under development an early, not-final version from August 2016 had to be analyzed for this contribution. It covers the primary education what in most areas of Germany means the first four years in school. That starts with an age of six years and ends with ten years. The standards are divided into the two levels what students should be able to do after the second year and after the fourth year. Overall, this version included 49 competencies, 27 for the first two years and 22 for the third and fourth year. The standards define five content areas that stayed the same also in newer released versions. What changed were the competencies themselves as well as the number of competencies for each area. The following numbers of competencies for the content areas only conform to the early version of the standards.

- Information and data: 13 competencies
- Algorithms: 10 competencies
- Language and automata: 12 competencies
- Informaticssystems: 6 competencies
- Informatics, humans and society: 8 competencies

It has to be mentioned, that in this version it is indicated, that some of the content areas are not complete and will include more competencies in later versions. Because the age range from grade one and two is from six to eight the first level of the competencies will correspond to level 1 in further discussion. The competencies of grades three and four cover the ages from eight to ten and will correspond to level 2.

f) Curriculum in Switzerland (Lehrplan 21): In Switzerland, the curriculum for primary and lower secondary education (called Lehrplan 21) was presented in 2014. Because the cantons have the authority over their school systems, they were free to accept it or not. Additionally the cantons were allowed to adapt the original version of this curriculum and therefore 21 of overall 26 cantons accepted the new curriculum. It contains the subject ”Media and informatics (Medien und Informatik)”, starting in the first year of primary school. Following the documents of the curriculum [2] the part called ”media” focuses on the understanding and responsible use of media, whereas ”informatics” includes basic concepts of computer science and problem solving. All the other subjects are responsible to foster the application competency needed in their content areas. The curriculum defines three cycles, which can be compared to levels. The first cycle contains kindergarten and the school grades one and two, cycle two grades three to six, and cycle three grades seven to nine. That means the first two cycles cover the primary education which can last in Switzerland four to six years. In the curriculum seven major competencies and content areas are described, four for the ”media” part and three for ”informatics”. For each of this major competencies competency levels, which differ in complexity and focus and represent steps in reaching the major competency, are defined and assigned to the cycle they should be learned in. Overall, 44 competency levels for cycles one and two exist, 14 for cycle one and 30 for cycle two. The following content areas are part of the curriculum, containing the corresponding number of competency levels for each area:

- Media: Life in media society: 3 competency levels
- Media: Understand media and media products: 7 competency levels
- Media: Produce media and media products: 7 competency levels
- Media: Communicate and cooperate with media: 3 competency levels
- Informatics: Data structures: 7 competency levels
- Informatics: Algorithms: 6 competency levels
- Informatics: Informatics systems: 11 competency levels

Cycle one covers a age range from five to eight years and will therefore correspond to level 1 in this discussion. The second cycle starts with the third year of primary education at an age of nine and ends after four years with an age of twelve and will correspond to level 2.

A summary of the facts described in the section above can be found in table I. Because of the different terms like

TABLE I
SUMMARY OF THE DESCRIPTIONS

		CSTA	CAS	AC	dk	GI	21
1.	**Number knowledge items**	45	16	22	49	49	44
2.	**Number level 1 knowledge items**	16	6	6	/	27	14
3.	**Number level 2 knowledge items**	29	10	16	/	22	30
4.	**Age range**	5-12	5-11	5-11	6-10	6-10	5-12

standards, competencies, statements, or learning objectives used in the curricula, educational standards, and competency models, we define the term ”knowledge items” as the general term for all of the mentioned concepts.

Table I does not explain differences in structure or complexity. So, where do the curricula, educational standards, and competency models differ? Do they cover the same content areas? Does a structural difference exist? Are there some problems concerning the connections of the components?

To answer these questions the chosen curricula, educational standards and competency models will be analyzed and compared, applying the method described in section IV. The results will be presented in section V.

IV. A Graph-based Approach

This paper aims at comparing curricula, educational standards and competency models by making use of a graph-based approach. Whereas the idea of using a graph is not new, our approach goes one step further. It uses typed edges and vertices (so both of them can be and are enriched by different types of annotations), and maps them to a neo4j graph database. We call the resulting model a *Generic Standards Model* (or GSM), and this section briefly describes its properties and possibilities.

A. Graph-based Model

As a first step comparable key elements of the curricula, educational standards and competency models had to be identified. In a discussion of three experts (one teacher and two researchers active in the field of computer science education) the intended learning outcomes, which also describe the content to be taught, were detected as possible comparable items. Each formulated learning outcome refers to a single knowledge item and is displayed as a vertex in the graph of the corresponding curriculum, standards or competency model. These vertices are annotated by the related curriculum, the age-group, keywords (describing the content), a reference to the classification scheme of the ACM/IEEE Computer Science Curriculum, experts’ ratings about their relevance for digital literacy or computer science, and a unique identification number. To finish this first phase of the process the results were revised two times by the experts. In a second step, the vertices were analyzed again and related to each other. For this, we classified the relations as either being a *”required by”* relation or an *”expanded by”* relation, leading to labeled (and directed) edges in the graph. The ”required by” relation indicates that one vertex is required by another one, and the ”expanded by” relation represents either a generalization or a specialization relationship. Mapped the relations to the graph database, a generic standard competency model for a curriculum or standard is created. These models with added relations were also revised twice by the experts, at first by the computer science teacher and then by the two researchers.

B. Basic Graph-theoretic Metrics

The representation of curricula, educational standards, and competency models in form of graphs offers new possibilities to analyze and compare. With the help of graph-theory and selected metrics, central vertices, structural problem, and complexity can be determined. In the article ”Mathematics of networks” [24] Newman defines different ways to locate ”important” vertices of a network or a graph, depending on what is meant with ”important”. Besides others he describes different measures of centrality in a graph, like *degree centrality*, *eigenvector centrality*, or *betweenness centrality*. Lightfoot points out, that these three measures are of interest for a curricula analysis and adds a clustering measure [4]. In this contribution we focus on the *degree centrality*, which gives a lot of information about central elements in a curriculum and is described as follows. The *degree* of a vertex represents the number of connected relations. The *degree centrality* is a simple method to measure the influence or importance of a vertex within a graph [24]. In directed graphs two types of degree can be determined: indegree and outdegree. The indegree of a vertex v counts the number of incoming relations to v, the outdegree of v counts the number of the outgoing relations of v. Vertices with a outdegree of 0 are called *sources* and vertices with an indegree of 0 are called *sinks*. These special vertices can help to find origins (*sources*) or destinations (*sinks*) in networks [4].

To determine information about the complexity of a graph, simple metrics are the *size*, the *number of edges*, and the *density*. Size and number of edges don’t need a description, but the *density* has to be defined. The *density* of a graph reflects the ratio of numbers of existing edges and possible edges. For directed graphs it is calculated by dividing the number of existing edges by the number of vertices multiplied with the number of vertices minus one [25].

The the method to detect of structural problems used in this paper bases on different aspects. One indication for problems in structure can be, that there exist vertices without edges to other vertices. These vertices have a degree of zero and can easily be determined. Further the *connected components* of a graph give more information about the structure. *Connected components* of a graph are subgraphs with the restriction, that every vertex of this subgraph is reachable from every other vertex of the same subgraph. For directed graphs the *weakly connected components* are connected components without considering the directions of the edges, whereas *strongly connected components* also consider the edge direction.

V. Results and Comparison

A. Using a Graph-database for Calculations

As mentioned this approach uses the graph database *neo4j* [26] to store the data, model the graph, and calculate the needed graph-theoretic metrics. Graph databases belong to the NoSQL databases and differ in some points from well known relational databases. They store the data and their connections in form of vertices and edges of a graph and don't rely on indexing. That means, connected vertices directly "point" at each other, what can lead to performance improvements [27].

The following query is written in *cipher*, the query language of *neo4j*, and shows one example of the many queries, that were used for this research. It determines the density for the graph of the CSTA standards.

```
START n=node(*), n2=node(*)
MATCH (n organization: "CSTA")
OPTIONAL MATCH (n)-[r]-(n2)
    WHERE n2.organization = "CSTA"
RETURN count(DISTINCT n) as nrNodes,
   count(DISTINCT r) as nrEdges,
   count(DISTINCT r)/((count(DISTINCT n)) *
   (count(DISTINCT n) - 1.0)) AS graphDensity
```

The graph database offers an interesting platform to represent curricula in form of graphs, and to analyze and compare different approaches. In further steps it will be used to build an online service to develop own curricula in the field of computer science education.

B. Analysis of Focus

The graph-based representation form offers the possibility to display the focus of a curriculum, educational standards, or a competency model in a very clear way, by using different colors. In this contribution this method is used to show two aspects of focus: which level and which category of the two categories "computer science" and "digital literacy" is represented more frequently.

Figure 1 shows one example graph of the the Australian curriculum for "Digital Technologies". The different colors of the vertices show the two levels: blue vertices correspond to level 1, green vertices correspond to level 2. The numbers in table I and the colored vertices in the graph show, that the Australian curriculum focuses on level 2, with 16 standards in level 2 and 6 in level 1.

A second aspect of focus is the categorization of the knowledge items into the areas "computer science" and "digital literacy". With regard to this, seven experts evaluated the curricula, standards, and competency models and first results from this study are presented by Micheuz, Pasterk and Bollin [28]. The results for the curriculum of Switzerland are shown in figure 2. In this case the red vertices represent knowledge items concerning "digital literacy", and the blue vertices correspond to "computer science". It shows a focus on "digital literacy" with 33 knowledge items from overall 44. The graph-based representation form gives a very useful overview of the situation.

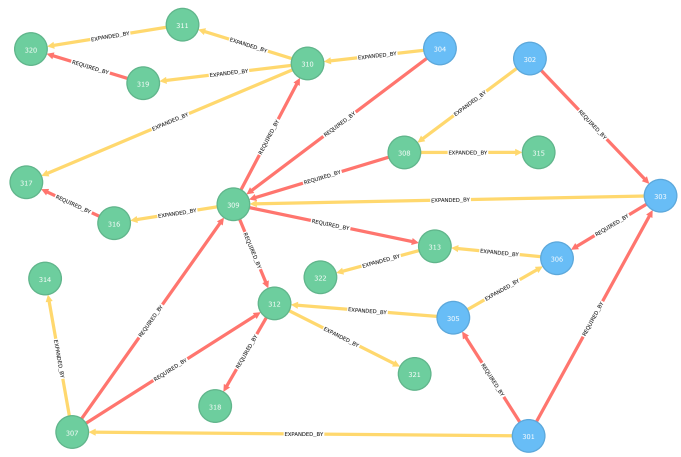

Fig. 1. The Australian Curriculum for "Digital Technologies" colored by knowledge items from level 1 (blue) and level 2 (green).

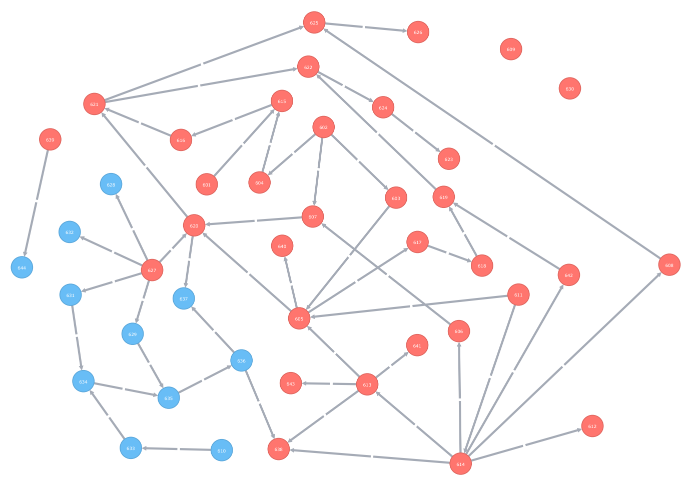

Fig. 2. The curriculum from Switzerland colored by categorization into computer science (blue) and digital literacy (red).

The two examples of possible aspects for the focus of curricula, educational standards, and competency models show, that a graph-based representation has advantages compared to other existing approaches. However, even more information can be extracted when looking at graph measures.

C. Analysis of Central Knowledge Items

The degree centrality, as it is described in section 4.B, is used to determine central and interesting knowledge items. In case of curricula analysis those vertices with lowest and highest degrees are of special interest. A high degree indicates a very important knowledge item, that can either be a prerequisite or a merging link, connecting knowledge from a lot of different knowledge items together, for a lot of other items. The first row of table II shows the highest degrees of the six analyzed curricula, standards, and competency models.

For example the vertex with the highest degree in the CSTA standards has a degree of 11. None of the other curricula contains a vertex with so many relations. Having a look at the standards gives an answer, why this standard has such a high degree:

- *"Use technology resources (e.g., puzzles, logical thinking programs) to solve age-appropriate problems."* [11]

In this case it is a very basic and general standard, that has impact on many other standards. Therefore its high degree is obvious. The GI standards show the lowest degree of the vertex with most relations, with a degree of 4.

Because the analysis is based on directed graphs, two different types of degree, indegree and outdegree, can be determined. The indegree shows the amount of incoming relations, the outdegree the number of outgoing relations. Also in this case the maximum and minimum values are of interest. Depending on the direction of the relation it can be a highly required or an often expanded knowledge item, or an item, that depends of a lot of other knowledge items. Concerning the overall outdegree again the same standard from the CSTA standards with the maximum degree stands out, having an overall outdegree of 10.

The highest indegree value refers to the knowledge item, that requires and expands most other knowledge items. Here the highest values of the six curricula are very balanced. An example from the Australian curriculum (AC) with an overall indegree of 4, representing the following learning objective:

- *"Collect, access and present different types of data using simple software to create information and solve problems"* [3]

With the help of the outdegree and the indegree special vertices can be identified, *sources* and *sinks*. Sources represent the vertices with no incoming relations, sinks are vertices without outgoing relations. In curricula sources can be interpreted as good knowledge items to start with, because they require no prior knowledge within this subject. Sinks in curricula refer to vertices, that mark the end of an topic and are not continued, or represent the knowledge level, that should be reached till the end of a given school grade, and are therefore meant to be continued. As it can be seen in row four of table II the curriculum from Switzerland contains with 8 sources the highest number of sources from all analyzed curricula. Examples from these sources would be the following competency levels:

- *"Students can compare notes from their direct environment concerning experiences with media or virtual living environments and can talk about their use of media."*
- *"Students can use different representation forms for data."* [2]

The numbers of sinks of all six curricula are very diverse. Most of them can be found in the CSTA standards with 17 sinks. In the curriculum from England there only occur 2 sinks. Two examples for sinks in the CSTA standards are following standards:

TABLE II
RESULTS OF THE ANALYSIS OF CENTRAL KNOWLEDGE ITEMS

		CSTA	CAS	AC	dk	GI	21
1.	**max. degree**	11	7	8	7	4	7
2.	**max. outdegree**	10	4	6	5	3	6
3.	**max. indegree**	4	4	4	4	2	3
4.	**Number of sources (indegree = 0)**	6	3	3	5	7	8
5.	**Number of sinks (outdegree = 0)**	17	2	7	13	10	13

- *"Develop a simple understanding of an algorithm using computer-free exercises."*
- *"Recognize that computers model intelligent behavior."* [11]

Reading these two standards shows, that they don't complete a topic and therefore it seems obvious that both will be continued in higher school grades. The two only sinks in the curriculum from England represent the following statements:

- *"Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs."*
- *"Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact."* [7]

The two statements show that both of them can complete a topic or also be continued. It seems to be not that clear as the two presented standards from the CSTA standards.

With the help of graph-theoretical metrics central and important knowledge items of the analyzed curricula, standards, and competency models could be determined and described. Therefore some possibilities using a graph-based approach could be shown.

D. Analysis of Complexity and Structure

Besides the analysis of the focus and central knowledge items also the complexity and the structure are of interest.

To evaluate the complexity of the curricula, standards, and competency models this contribution focuses on the two basic metrics size and density. To calculate the size the number of vertices and relations can be considered. As the overall numbers of knowledge items for each curriculum are already shown in table I, row one of table III gives some information about the number of relations. The highest number of knowledge items have the Austrian competency model "digikomp4" and the educational standards from the GI, with 49 items, and the highest number of relations can again be found in the "digikomp4" competency model, with 64 relations. It has to be considered, that the numbers of relations highly depend on the numbers of knowledge items. Therefore the density of each graph has been calculated. A comparison of the density values shows, that the graph of the CAS curriculum from England has with 0.079 the highest density of all analyzed curricula, educational standards, and competency model. So although this curriculum contains least knowledge items and relations, the density indicates a high complexity.

In the third row of table III, the numbers of vertices with a degree of 0 are presented. These vertices have no relations to any other vertex of the graph. This indicates, that they cover knowledge, that is more or less independent from other knowledge items, what can signify structural problems. In case of the ”digikomp4” competency model from Austria there can be found one vertex without relations to other vertices:

- *”I can encrypt and decrypt some information from every-day life.”* [21]

This competency represents the start of a new topic that is not continued by other competencies in this model for primary education. Of course that can be a indication, that this competency will be required in parts of secondary education. Comparing all curricula the number of vertices with a degree of 0 is very low, with a maximum of 2. The curricula from England (CAS) and Australia (AC) have no vertices without relations.

As it can be seen in row four and five of table III the standards from the GI and the curriculum 21 from Switzerland contain knowledge items in level 1, which have no relations to knowledge items in level 2. This can indicate structural problems. In case of the curriculum 21 the following competency level is affected:

- *”Students can arrange things considering self chosen attributes, to find objects with certain attributes faster.”* [2]

The number of connected components can also give some information about the structure of curricula, educational standards, and competency models. In case of this contribution the weakly connected components are considered. The higher the number of these components is, the more vertex cluster are not connected to each other. This can indicate more independent content areas. Taking a look at the numbers of connected components, the standards from GI and the curriculum 21 from Switzerland show the highest numbers. Here has to be considered, that vertices without any connection are also counted as single connected components. That means for both curricula two connected components are single vertices. In case of the curriculum 21 the other two connected components can be interpreted as the two major topics of this curriculum ”media” and ”informatics”.

With the help of basic graph-theoretic metrics the complexity of analyzed curricula, educational standards, and competency models could be determined and some conclusions about the structure and possible structural problems could be drawn.

TABLE III
RESULTS OF THE ANALYSIS OF COMPLEXITY AND STRUCTURE

		CSTA	CAS	AC	dk	GI	21
1.	**Overall relations**	61	19	30	64	50	51
2.	**Density**	0.034	0.079	0.065	0.027	0.021	0.027
3.	**Vertices with degree = 0**	2	0	0	1	2	2
4.	**Vertices with degree = 0 in level 1**	0	0	0	/	2	1
5.	**Vertices with degree = 0 in level 2**	2	0	0	/	0	1
6.	**Connected components**	3	1	1	3	4	4

VI. CONCLUSION AND FUTURE WORK

In this contribution we wanted to show, that a graph-based approach to analyze and compare curricula, educational standards, and competency models offers new possibilities. Therefore six different models were selected and, in a first step, described. Already within this first phase some differences could be found, but the questions regarding focus, complexity, central knowledge items, and structure could not be answered by this method. In order to answer these questions, the knowledge items of the selected curricula, standards, and competency models were connected via dependency relation, resulting in individual directed graphs. With the help of this representation form and a graph database, the required information could be assembled. Concerning the focus of a specific curriculum a graph-based representation form can give a clear overview of the situation. It is possible to differ content areas or levels by color and simultaneously show the content or level crossing connections. This paper also shows, that by determining some basic graph-theoretic values, some conclusion about central knowledge items, complexity, and structure of the selected curricula, standards, and competency models can be drawn. The degree was used to find central knowledge items and identify possible structural problems with not connected knowledge items. The complexity was determined with the help of the size and the density of the graphs. The structure was further investigated by finding the connected components of the graphs. Therefore all three research questions can be answered with ”yes” and the results show, that a graph-based representation and graph-theoretic calculations offer new possibilities for analysis and comparison of curricula, educational standards, and competency models.

In future steps the knowledge items of the curricula will be categorized into the knowledge areas of the ACM/IEEE curriculum and broken down to their smallest knowledge items. Similar knowledge items will be merged together and a general graph, containing all the knowledge items of the existing curricula, standards, and competency models, will be generated. To enable collaborative work, an online platform will be developed, based on the graph database neo4j. Experts will have the possibility to evaluate existing graphs from curricula, and users like teachers will be able to develop an own curriculum from knowledge items defined in existing curricula.

REFERENCES

[1] National Curriculum UK, https://www.gov.uk/government/collections/national-curriculum, 2014, retrieved on 30.04. 2017.

[2] Lehrplan 21, http://v-ef.lehrplan.ch, 2014, retrieved on 30.04. 2017.

[3] Australien Curriculum, http://www.australiancurriculum.edu.au/technologies, 2013, retrieved on 30.04. 2017.

[4] J. M. Lightfoot, "A Graph-Theoretic Approach to Improved Curriculum Structure and Assessment Placement," *Communications of the IIMA*, vol. 10, no. 2, pp. 59–73, 2010.

[5] L. Marshall, "A comparison of the core aspects of the acm/ieee computer science curriculum 2013 strawman report with the specified core of cc2001 and cs2008 review," in *Proceedings of Second Computer Science Education Research Conference*, ser. CSERC '12. New York, NY, USA: ACM, 2012, pp. 29–34. [Online]. Available: http://doi.acm.org/10.1145/2421277.2421281

[6] S. Pasterk and A. Bollin, "A graph-based approach to analyze and compare computer science curricula for primary and lower secondary education." in *Proceedings of the 22nd Annual Conference on Innovation and Technology in Computer Science Education*, ser. ITiCSE '17. ACM, in press.

[7] M. Berry, *Computing in the national curriculum - A guide for primary teachers*. Computing at School, 2013.

[8] E. Barendsen, L. Mannila, B. Demo, N. Grgurina, C. Izu, C. Mirolo, S. Sentance, A. Settle, and G. Stupurienė, "Concepts in k-9 computer science education," in *Proceedings of the 2015 ITiCSE on Working Group Reports*, ser. ITICSE-WGR '15. New York, NY, USA: ACM, 2015, pp. 85–116. [Online]. Available: http://doi.acm.org/10.1145/2858796.2858800

[9] A. f. C. M. A. Joint Task Force on Computing Curricula and I. C. Society, *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. New York, NY, USA: ACM, 2013.

[10] C. Duncan and T. Bell, "A pilot computer science and programming course for primary school students," in *Proceedings of the Workshop in Primary and Secondary Computing Education*, ser. WiPSCE '15. New York, NY, USA: ACM, 2015, pp. 39–48. [Online]. Available: http://doi.acm.org/10.1145/2818314.2818328

[11] D. Seehorn, S. Carey, B. Fuschetto, I. Lee, D. Moix, D. O'Grady-Cunniff, B. B. Owens, C. Stephenson, and A. Verno, "Csta k–12 computer science standards: Revised 2011," New York, NY, USA, Tech. Rep., 2011, 104111.

[12] N. C. C. Brown, S. Sentance, T. Crick, and S. Humphreys, "Restart: The resurgence of computer science in uk schools," *Trans. Comput. Educ.*, vol. 14, no. 2, pp. 9:1–9:22, Jun. 2014. [Online]. Available: http://doi.acm.org/10.1145/2602484

[13] K. Falkner, R. Vivian, and N. Falkner, "The australian digital technologies curriculum: Challenge and opportunity," in *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148*, ser. ACE '14. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2014, pp. 3–12. [Online]. Available: http://dl.acm.org/citation.cfm?id=2667490.2667491

[14] M. Pedroni, M. Oriol, and B. Meyer, "A framework for describing and comparing courses and curricula," in *Proceedings of the 12th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, ser. ITiCSE '07. New York, NY, USA: ACM, 2007, pp. 131–135. [Online]. Available: http://doi.acm.org/10.1145/1268784.1268824

[15] T. Sekiya, Y. Matsuda, and K. Yamaguchi, "Analysis of computer science related curriculum on lda and isomap," in *Proceedings of the Fifteenth Annual Conference on Innovation and Technology in Computer Science Education*, ser. ITiCSE '10. New York, NY, USA: ACM, 2010, pp. 48–52. [Online]. Available: http://doi.acm.org/10.1145/1822090.1822106

[16] P. Hubwieser, M. Armoni, T. Brinda, V. Dagiene, I. Diethelm, M. N. Giannakos, M. Knobelsdorf, J. Magenheim, R. Mittermeir, and S. Schubert, "Computer science/informatics in secondary education," in *Proceedings of the 16th Annual Conference Reports on Innovation and Technology in Computer Science Education - Working Group Reports*, ser. ITiCSE-WGR '11. New York, NY, USA: ACM, 2011, pp. 19–38. [Online]. Available: http://doi.acm.org/10.1145/2078856.2078859

[17] A. Tucker, "A model curriculum for k–12 computer science: Final report of the acm k–12 task force curriculum committee," New York, NY, USA, Tech. Rep., 2003, aCM Order No.: 104043.

[18] E. Schoolnet, "Computing our future: Computer programming and coding ? priorities, school curricula and initiatives across europe (update 2015," Tech. Rep., 2015, update 2015.

[19] Digikomp4 competency model, http://digikomp.at/praxis/portale/digitale-kompetenzen/digikomp4-volksschule/kompetenzmodell.html, 2013, retrieved on 30.04. 2017.

[20] Gesellschaft für Informatik (GI), "Bildungsstandards Informatik fr den Primarbereich," http://bscw.ham.nw.schule.de/pub/bscw.cgi/d5797526/LP_IF.pdf, 2016, retrieved on 30.08. 2016.

[21] U. Mulley and B. Zuliani, "Digitales Kompetenzmodell für die Volksschule," *Digitale Schule Österreich, OCG Schriftenreihe*, 2013.

[22] Gesellschaft für Informatik (GI), "Bildungsstandards Informatik für die Sekundarstufe I," https://www.gi.de/fileadmin/redaktion/empfehlungen/Bildungsstandards_2008.pdf, 2008, retrieved on 30.04. 2017.

[23] ——, "Bildungsstandards Informatik für die Sekundarstufe II," https://www.gi.de/fileadmin/redaktion/empfehlungen/Bildungsstandards-Informatik-SekII.pdf, 2016, retrieved on 30.04. 2017.

[24] M. E. J. Newman, "Mathematics of networks." in *The New Palgrave Dictionary of Economics*, 2nd ed., S. N. Durlauf and L. E. Blume, Eds. Palgrave Macmillan, 2007.

[25] C. Prell, *Social network analysis: history, theory and methodology*. London, UK: SAGE, 2012.

[26] Neo4j Graph Database, https://neo4j.com/, retrieved on 30.04. 2017.

[27] I. Robinson, J. Webber, and E. Eifrem, *Graph Databases*, 2nd ed. Sebastopol, CA, USA: O'Reilly, 2015.

[28] P. Micheuz, S. Pasterk, and A. Bollin, "Basic digital education in austria. one step further." in *Proceedings of the World Conference on Computers in Education*, ser. WCCE '17. IFIP, in press.