Developing the CC2020 Knowledge Table and Tool to Globally Benchmark Computing Degrees

Alison Clear
Department of Computing
Eastern Insitute of Technology
Auckland, New Zealand
aclear@eit.ac.nz

Tony Clear
Department of Computer Science and Software Engineering
Auckland University of Technology
Auckland, New Zealand
tony.clear@aut.ac.nz

***Abstract*— The ACM/IEEE-CS Computing Curricula 2020 report was published in March 2021. The report is global in its attention and covers all aspects of the computing disciplines that had ACM/IEEE approved curricula: Computer Engineering, Computer Science, Software Engineering, Information Systems, Information Technology and Cyber Security.**

The CC2020 Steering Committee analysed all the approved curricula in the six discipline areas and looked for any overlaps. For example, all the areas have some aspect of programming but not all at the same level. Using a thematic analysis methodology, a "Landscape of Computing Knowledge" table was then developed with six main topic areas which included 34 subsections. Each computing discipline specified a minimum and maximum value for each subsection for that discipline, suggesting an importance range within which most degree programs are likely to fall.

The CC2020 project then developed online visualisations of the knowledge areas, where stakeholders can assess a degree program with the values required and then match it against any of the current approved curricula. This tool enables users to compare individual degrees with the ACM/IEEE curricula recommendations, to collect the structure and focus of computing degrees globally and compare computing degrees across institutions globally. It also enables industry to understand the different areas and contribute to curriculum development. This tool has now been used to gather structure and variation of degree programs globally.

This paper will describe and reflect on the development of the Landscape of Computing Knowledge table, and the development of the visualization tool. Insights from the first workshops that have investigated how to use the tool and its utility will be reviewed.

Keywords—Computing Curriculum, Computing Knowledge, Online tool

I. INTRODUCTION

The Association of Computing Machinery (ACM) Education Advisory Committee (ACM-EAC) and the Institute of Electrical and Electronic Engineers (IEEE) Computer Society (IEEE-CS) a project to update the Computing Curriculum 2005: the Overview Report (CC2005) [1]. CC2005 was commissioned to provide an overview of the several computing disciplines represented in each of the approved curricula at the time, showing the scope of the field and how each discipline was related to the others. The update was to be named Computing Curriculum 2020 (CC2020). The purpose of the project was to:

> "*The purpose of the CC2020 project, as a modern extension of the CC2005 report, is to provide global guidance in an evolving computing environment as it affects baccalaureate degree programs in computing worldwide.*" [2]

As this project was to be global in nature and across all the computing disciplines, the mission statement was:

> "*..... to produce a globally accepted framework for specifying and comparing computing baccalaureate degree programs that meet the growing demands of a changing technological world and is useful for students, industry, and academia*". [2]

The leading sponsors of the project were ACM and IEEE-CS and also ACM China, ACM India, Association for Information Systems (AIS), Education SIG of the Association for Information Systems and Computing Academic Professionals (ED-SIG), Informatics Europe, Italian Association of Computer Scientists (GRIN) and the Special Interest Group on Computer Human Interaction of (SIGCHI).

A Steering Committee of 13 members was established, later in the project this was to increase to 15 members. A Task Force of 50 members came from all the different computing disciplines, and as this project was global, from 20 countries.

A clear difference exists between the computing disciplines, the CC2005 report identified these, compared them, addressed the differences and identified the overlaps in generic topic areas. They all have distinguishing characteristics that are essential for their individual identities. It was important to include all disciplines and the ACM and IEEE-CS approved curricula at the time were used. However, all the disciplines had evolved in the following 15 years. The report analysed the six discipline areas of computing that currently had ACM and IEEE-CS approved curricula:

- Computer Engineering
- Computer Science
- Software Engineering
- Cyber Security
- Information Systems
- Information Technology

Data Science was not included in CC2020, it was still being developed and was eventually approved in Feb 2021.

The CC2020 report adopted the word "Computing" as a unifying term and adopted "competency" to represent the foundation of all future computing curriculum reports. All the six disciplines are constantly updating their curriculum to encompass the rapidly changing technology in their own disciplines. The project was also charged with generating modern visualizations to represent the disciplines.

To further identify the different topics in each of the six approved curricula and also identify the overlapping topic areas it was decided to develop a "Landscape of Computing Knowledge" table (LoCKt). [2, p65] This would also feed into the visualization tool specifications. Conceptual visualization work has been undertaken by members of the CC2020 Task Force [3] which has helped inform the research.

Consequently, the overarching research question driving this work was "How best to visualize and compare computing curricula for varying stakeholders".

This paper describes the development of the table, the visualization tool and populating the database of the tool with actual program data sourced from universities around the world. These developments and their implications are reviewed based on the work to date.

II. METHODOLOGY

Developing the Landscape of Computing Knowledge Table and supporting visualization tool adopted a two phased approach of theory building and system development. The first phase used thematic analysis, with the system building phase drawing upon a multimethodological approach from Nunmaker et al [4].

A. *The Landscape of Computing Table*

To identify and frame the areas of computing a thematic analysis (See Fig 1) of the content drawn from all the six approved curriculum documents, was undertaken.

> *"Thematic analysis is an approach that is often used for identifying, analyzing, and reporting patterns (themes) within data in primary qualitative research."* [5]

This was conducted by a subset of the CC2020 Steering Committee led by Professor Heikki Topi of Bentley College. The challenge was to create a conceptual framework providing a taxonomy and an overall view of all the topics in all the curricula. See Fig 2.

The curricula content was interpreted and aggregated to create 39 distinct knowledge areas (codes or sub sections).

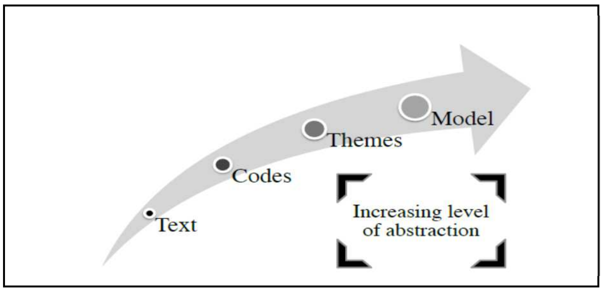

Fig 1: Levels of interpretation in thematic synthesis

Further follow-up analysis enabled the knowledge areas to be consolidated into the final 34 knowledge areas (codes or sub sections).

The expert team then considered the 34 knowledge areas and identified recurring themes or topics. These knowledge areas were then organized based on the deductive framing provided by the semiotic ladder [6,7] which resulted in a six-layer categorization of the themes (topics) as outlined below:

- users and organizations
- systems modeling
- systems architecture and infrastructure
- software development
- software fundamentals
- hardware

For each topic area (theme) 34 sub sections constituted their underlying codes eg *2.3 Requirements Analysis and Specification* was placed under the theme "*2 Systems Modelling*" See Table 1. The themes and codes were then mapped to the six disciplines of the approved computing curricula. See Table 1.

The next stage was to identify the relative importance of each sub section to each of the six disciplines. As the CC2020 Task Force was made up of experts in all six disciplines they were asked to rate each sub section on a scale of 0, being not important to that discipline, to 5 being most important to that discipline. A consensus of the ratings was derived and entered into the final table.

An expandable version of Fig 2 is available at https://doi.org/10.5281/zenodo.6655243

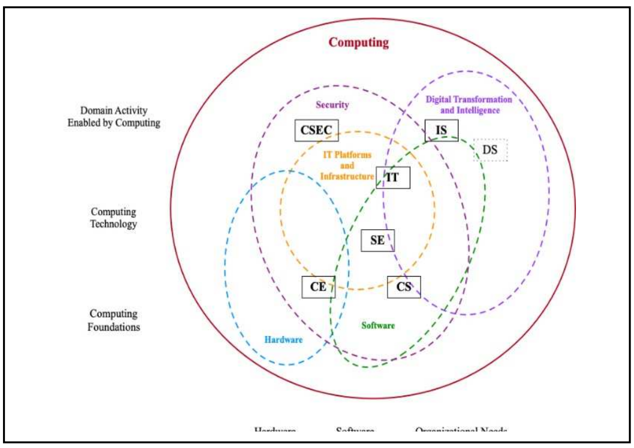

Fig 2: Knowledge Areas of the six computing disciplines

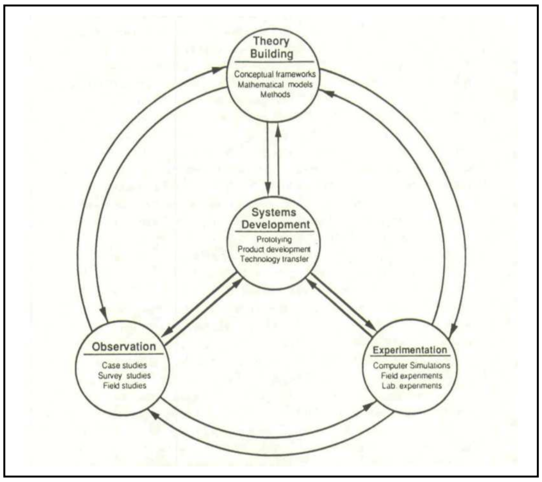

Fig. 3. A Multimethodological Approach to IS Research ex [2]

For example, in the theme *4. Software Development* and the sub section *4.4 Software Design* the experts considered for the discipline of Computer Science programming, should have a knowledge level between 2 and 4. By contrast students studying Software Engineering would have an expected knowledge level of programming between 4 and 5.

B. The Visualization Tool

In this system building phase guided by the model in Fig 3, the visualization tool was conceptualized, interactively built and tested then trialled in the field. This process was based on the *conceptual framework* provided by the Landscape of Computing Knowledge Table (LoCKt) an interactive process of *prototyping* to develop the tool was undertaken. Once a viable version was available a series of workshops with academic members of professional societies and industry professionals were conducted as *field studies* to populate and refine the tool.

III. THE KNOWLEDGE TABLE AND TOOL

A. The Landscape of Computing Knowledge table

The table derived from the process in II A is depicted in Table 1.

The table was further enhanced by identifying the highest (5) and lowest range of ratings (0) for each discipline. These were then shaded in yellow for the highest rating and grey for the lowest rating. These colorings help users quickly identify the main emphasis for each of the six disciplines. The levels reflect the consensus view of the appropriate emphasis for each discipline. For example, the discipline of Computer Engineering has a prime emphasis on *6. Hardware* where all the sub sections are rated high and has a lower emphasis on topics such as *4. Software Development,4.5 Platform based Development.*

A further depiction of the LoCKt was created as a radar chart shown in Fig 4 available on page 66 of the CC2020 report. [2] and an expanded version is available at Zenodo https://doi.org/10.5281/zenodo.6655243

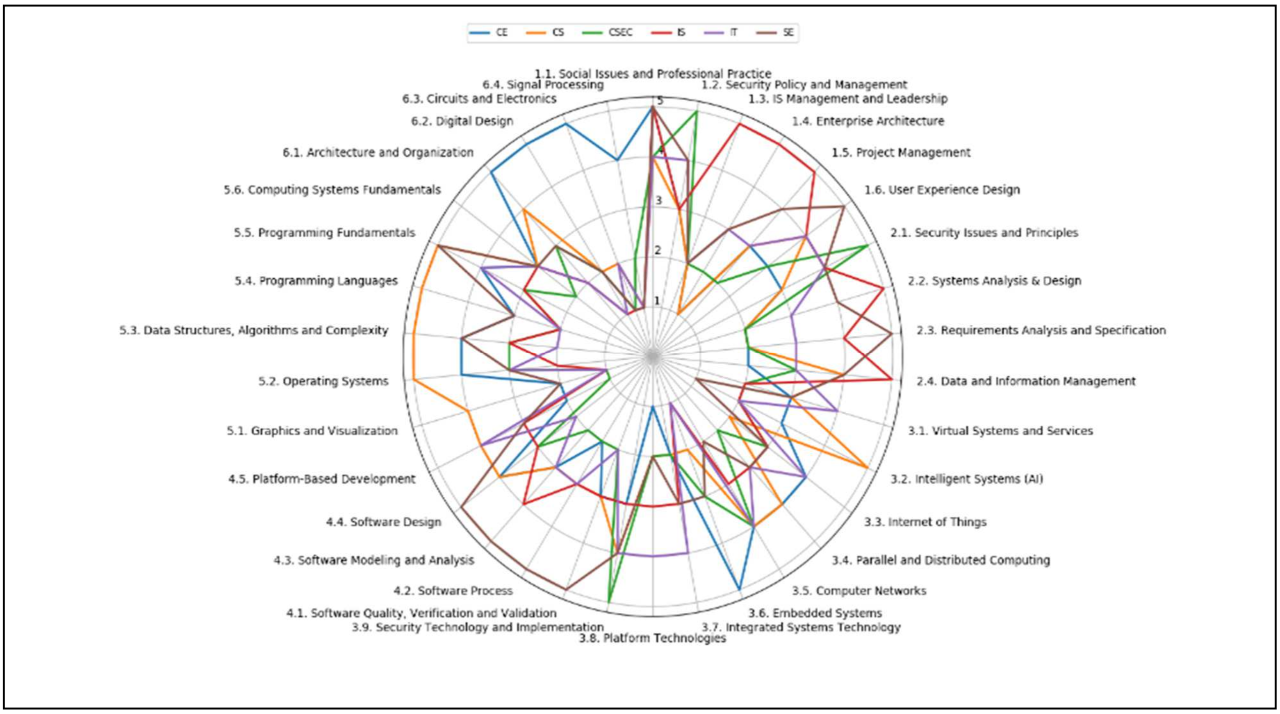

Fig. 4 Radar Chart of the Landscape of Computing Knowledge Areas

TABLE 1 CC2020 Landscape of Computing Knowledge Table

		CE		CS		CSEC		IS		IT		SE	
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
1. Users and Organizations	1.1. Social Issues and Professional Practice	2	5	2	4	2	4	3	5	2	4	3	5
	1.2. Security Policy and Management	1	3	2	3	4	5	2	3	2	4	2	4
	1.3. IS Management and Leadership	0	2	0	2	1	2	4	5	1	2	1	2
	1.4. Enterprise Architecture	0	1	0	1	1	2	3	5	1	3	1	3
	1.5. Project Management	1	3	2	3	1	2	4	5	2	3	2	4
	1.6. User Experience Design	1	3	2	4	1	3	2	4	3	4	3	5
2. Systems Modeling	2.1. Security Issues and Principles	2	3	2	3	4	5	2	4	3	4	2	4
	2.2. Systems Analysis & Design	1	2	1	2	1	2	4	5	1	3	2	4
	2.3. Requirements Analysis and Specification	1	2	1	2	0	2	2	4	1	3	3	5
	2.4. Data and Information Management	1	2	2	4	2	3	3	5	2	3	2	4
3. Systems Architecture and Infrastructure	3.1. Virtual Systems and Services	1	3	1	3	1	2	1	2	3	4	1	3
	3.2. Intelligent Systems (AI)	1	3	3	5	1	2	1	2	1	2	0	1
	3.3. Internet of Things	2	4	0	2	1	3	1	3	2	4	1	3
	3.4. Parallel and Distributed Computing	2	4	2	4	1	2	1	3	1	3	2	3
	3.5. Computer Networks	2	4	2	4	2	4	1	3	3	4	2	2
	3.6. Embedded Systems	3	5	0	2	1	3	0	1	0	1	0	3
	3.7. Integrated Systems Technology	1	2	0	2	0	2	1	3	3	4	1	3
	3.8. Platform Technologies	0	1	1	2	1	2	1	3	2	4	0	2
	3.9. Security Technology and Implementation	2	3	2	4	4	5	1	3	2	4	2	4
4. Software Development	4.1. Software Quality, Verification and Validation	1	3	1	3	1	2	1	3	1	2	3	5
	4.2. Software Process	1	2	1	3	0	2	1	3	1	3	3	5
	4.3. Software Modeling and Analysis	1	3	1	3	1	2	2	4	1	3	4	5
	4.4. Software Design	2	4	2	4	1	3	1	3	1	2	4	5
	4.5. Platform-Based Development	0	2	2	4	0	1	1	3	2	4	1	3
5. Software Fundamentals	5.1. Graphics and Visualization	1	2	2	4	0	1	1	1	0	1	0	2
	5.2. Operating Systems	2	4	3	5	2	3	1	2	1	3	1	3
	5.3. Data Structures, Algorithms and Complexity	2	4	4	5	1	3	1	3	1	2	2	4
	5.4. Programming Languages	2	3	3	5	1	2	1	2	1	2	2	3
	5.5. Programming Fundamentals	2	4	4	5	2	3	1	3	2	4	3	5
	5.6. Computing Systems Fundamentals	2	3	2	3	1	2	2	3	1	3	2	3
6. Hardware	6.1. Architecture and Organization	4	5	3	4	1	3	1	2	1	2	1	3
	6.2. Digital Design	4	5	1	2	0	2	0	1	0	1	0	2
	6.3. Circuits and Electronics	4	5	1	2	0	1	0	1	1	2	0	1
	6.4. Signal Processing	3	4	0	1	0	2	0	1	0	1	0	1

B. The Visualization Tool

The visualization tool (*Viztools)* see Fig 5, was developed to meet the following goals.

- to collect the structure and focus of computing degrees globally
- To compare individual degrees with the ACM/IEEE curricula recommendations
- To compare computing degrees across institutions globally
- To enable industry to understand the different areas and contribute to curriculum development

▪ Fig. 5. Intro screen of the CC2020 Visualization Tool

From the field studies these further potential uses evolved:

- For academia to better understand industry
- To enable highlighting of emerging areas in industry and research

1) Embedding the six curricula recommendations The six standard curricula recommendations from the LoCKt were embedded in the database underlying the web based tool to enable users to compare against any of these curricula. The contents of these standard curricula were drawn from the six discipline columns of the LoCKt. See Table 1. Also available at Zenodo https://doi.org/10.5281/zenodo.6655243

2) Creating individual profiles Users are guided to create a profile for their own degree program(s) by assigning a value between 0 and 5 to each of the 34 sub sections. A value of 0 means it is not taught in this program while a value of 5 would mean it is of prime importance to this program. When completed this can be compared against the standard approved ACM/IEEE curricula in the tool. Fig 6 compares a program from UTEC in Lima, line graph, with the standard Computer Science profile (shaded area).

Individual programs, once data has been entered into the tool, may be compared with any of the standard discipline profiles.

A further feature of the tool is the ability to compare an individual program with any other program in the database. Comparisons can be one to one or one to a group which may or may not include a standard discipline profile.

3) Creating Job Profiles As well as degree program profiles, rather than using a programmatic focus, the tool supports mapping of job profiles to curricula and programs which can highlight gaps or different emphases. An employer can create a profile by using the values in each of the 34 sub sections of the LoCKt and then compare that profile against degree programs or any of the standard discipline profiles.

Fig 6 Fig Program Comparison

IV. Discussion and Reflection

A. Discussion

1) Competency and Knowledge

In the CC2020 report the development of curricula based on a competency approach was recommended. Competency has been defined to include

knowledge + skills + dispositions performed through a task within a context

The visualization tool is built upon the LoCKt based on the knowledge and skill level content of the six approved curricula. Further work is currently being undertaken to build a tool that will include the disposition elements.

2) Workshops

Once the tool was developed a series of workshops (field studies) were held with academics, selected computing curriculum experts, members of professional societies and industry professionals to populate and refine the tool. To date some three local and national workshops have been conducted in New Zealand [8,9,10] one national workshop in Japan [11], one regional workshop at the Centro Latinoamericano de Estudios Informatica (CLEI) conference in South America (comprising 10+ countries) [12] and one global workshop at the 2022 ACM Special Interest Group on Computer Science Education (SIGCSE) Technical Symposium [13].

In each of the workshops the participants created their own degree profiles and not only were able to compare them to the six approved curricula profiles but also to each other's degree programs. These profiles had been loaded and are currently in the Viztools database.

At the industry focused workshop the authors presented a role play of a Secure DevOps Engineer (DevOps is a set of practices that combines software development and IT operations) job profile and mapped that to the standard programs and also to local degree programs.

In this process of exploring the functionality of the visualization tool with live data and users, the workshops have served as field studies as highlighted in Fig 3 enabling interaction with the prototyping process of systems development and in turn feeding back to augment the conceptual framework of the LoCKt model.

B. Reflection

1) Levels

In the process of conducting the workshops it became apparent that the notion of level (0 – 5) and how it was determined posed some challenges for participants. LoCKt provides a range for each sub section and these ranges were derived by a consensus of opinion by experts from each discipline.

However, for individual users they could only enter one level which reflected the level of emphasis of that sub section (For example, *5.2 Operating Systems* is a sub section). in their program, Participants discussed with the workshop leaders how to interpret a level defined for each subsection eg *5.2 Operating Systems* is defined in a range of 3 – 5 for the Computer Science standard curriculum profile.

It was then necessary to interpret the level applicable in the selected program chosen by the participants, if it was of major emphasis in their program they would enter 5 however if it was emphasized but did not appear at a dominant level in the program, then they would probably enter a 3.

The key choice when allocating these ratings was whether the ratings applied at an overall level for the program and expectations of a graduate or did the ratings only apply at the course level.

In sub section 5.5 Programming Fundamentals which covers all levels of programming, the Computer Science standard curriculum profile has a range of 4 – 5 where Software Engineering ranges from 3 – 5. Most programs that teach advanced programming would enter a 5.

2) Professional Knowledge

On observing the LoCKt in the theme or topic *1 Users and Organizations*, professional knowledge is touched on in sub sections 1.1, 1.2 and 1.5. However, this proved problematic in the workshops for programs that had a greater emphasis on professional knowledge. These are further developed in the CC2020 report and need to be included in revisions of the approved curricula documents and in some way reflect the emphasis that CC2020, responding to industry feedback, placed on them.

3) Different Program Structures

The participants had to decide how to represent their programs particularly when they involved multiple majors. This is easily resolved by creating a separate profile for each major or program within an institution.
Different countries around the world have differing durations of baccalaureate study with three- or four-year degrees being the most common. Therefore, it is assumed the final year of either duration represents achievement at the same level.
Partial or shorter programs (eg certificates, diplomas, associate degrees, micro-credentials) can be interpreted using the tool however it is less likely that topics would be rated at level 5.

4) Quality Assurance. Feedback from expert consultation and workshop participants indicated the potential for the tool to support quality assurance activities such as accreditation of programs, monitoring and review of programs and moderation of individual courses.

5) Future Work

At present there is only one approved post graduate (graduate) curricula, Master of Information Systems (MIS). The tool is currently unable to include more advanced programs.

The Data Science curricula was approved in February 2021 so was not included in the CC2020 report or the LoCKt. Future work will include this discipline in the LoCKt and visualization tool.

V. Conclusion

This paper has reported on the development of the CC2020 Landscape of Computing Knowledge table (LoCKt) and the visualization tool (Viztool) that were created as part of the CC2020 project and report. All areas of computing with ACM/IEEE approved curricula were included in the project, table and tool.

Field studies conducted through workshops with academics, computing curricula experts and industry professionals were instrumental in populating the database of the tool with actual program data sourced from universities and industry around the world. These field studies also contributed to profiling computing programs and professional roles at the local, national and global level.

Feedback from the workshops and related activity has indicated areas where the tool can be augmented and further informing frameworks and curricula can be included. It is pleasing to know this work is underway.

The work reported here promises to be of significant value in helping academics visualise their own computing baccalaureate programs and comparing them to the ACM and IEEE approved curricula. Similarly, it will have value for industry stakeholders to visualize job profiles in relation to specific employment needs, competencies and programs.

Acknowledgment

The authors would like to thank the sponsors of the Computing Curriculum 2020 project, members of the Steering Committee and Task Force and Carlos Cupe of UTEC, Lima.

References

[1] Computing Curriculum 2005: The Overview Report. https://www.acm.org/education/curricula-recommendations

[2] Clear, A., Parrish, A. et al "Curriculum Curricula 2020: Paradigms for Global Computing Education.:: 2021 https://www.acm.org/education/curricula-recommendations.

[3] Takada, S., Cuadros-Vargas, E., Impagliazzo, J. *et al.* Toward the visual understanding of computing curricula. *Educ Inf Technol* **25,** 4231–4270 (2020). https://doi.org/10.1007/s10639-020-10127-1

[4] Nunamaker, J., Chen, M., and Purdin, T.: "Systems Development in Information Systems Research", Journal of Management Information Systems, 1990, 7, (3), pp. 89-106

[5] D. Cruzes, and T Dyba "Recommended Steps for Thematic Synthesis in Software Engineering" In proceedings International Symposium on Empirical Software Engineering and Measurement, 2011 978-0-7695-4604-9/11 $26.00 © 2011 IEEE DOI 10.1109/ESEM.2011.36 pp 275 -284

[6] Liu, K., "Semiotics in Information Systems Engineering", Cambridge University Press, Cambridge, U.K, 2000.

[7] R. K. Stamper, "Information in Business and Administrative Systems", (John Wiley and Sons, New York, NY), 1973.

[8] A. Clear "The Landscape of Computing: Benchmarking ITP Computing Degrees" In Proceedings of the 12th CITRENZ Annual Conference pp 100 – 101 July 2021 https://www.citrenz.ac.nz/conferences/2021/pdf/20.%20The%20Landscape%20of%20Computing%20Benchmarking%20ITP%20Computing%20Degrees.pdf

[9] A. Clear and T. Clear Workshop "Global Benchmarking and Visualising of ITP degree programs" In Proceedings of the 12th CITRENZ Annual Conference pp 108 – 109 July 2021 https://www.citrenz.ac.nz/conferences/2021/pdf/25.%20Workshop%20Global%20Benchmarking%20and%20Visualizing%20of%20ITP%20Degree%20Programs.pdf

[10] A. Clear and T. Clear. "Tech Degrees: International Developments" Invited Presentation ITx Innovation Days (ITx2021), Auckland, New Zealand & Wellington, New Zealand 2021 https://itx.nz/Programme/717/Tech-Degrees-International-Developments

[11] S. Takada: "Toward the visual understanding of computing curricula", Forum on Information Technology (FIT 2021), Aug 27 2021, (in Japanese).

[12] E. Cuadros-Vargas and A. Sabiguero (2021). Panel on Computing Curricula at CLEI2021. San Jose de Costa Rica 2021

[13] A. Clear, E. Cuadros-Vargas and S. Takada. "CC2020 Visualization Tool". In Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE 2022). Association for Computing Machinery, New York, NY, USA, 1063–1064. DOI:https://doi.org/10.1145/3478432.3499029