Hypergraph-based framework for Information System design and development

András Béleczki[a], Bálint Molnár[b], Zoltán Vincellér[c]

[a]beleczki.andras@outlook.com

[b]Eötvös Loránd University
molnarb@inf.elte.hu

[c]Eötvös Loránd University
vzoli@inf.elte.hu

Abstract

The lifecycle of information systems is always containing several different phases which require different inputs from stakeholders, developers, key users and so on[2, 9]. This also means, that defining needs, making design decisions, and implementing them requires a variety of tools, frameworks, and applications. To meet every demand we use graphical diagrams, design, and conceptual documents, low code representations, metamodels, and actual source code. All of these are part of such a heterogenous environment that is very error-prone based on its nature.

We already discussed a possible mathematical approach, which could depict these heterogenous domains in a homogenous formal way: the hypergraphs[3–6]. But instead of creating a whole new framework, we propose to extend an existing one. The SAP Cloud Application Programming is an ORM-like framework[8], which covers the world of relational database development with easily understandable object-oriented entity definitions. It uses SAP HANA as its core which contains graph capabilities besides the default relational database features[7]. We already explored the options, of how hypergraphs - with bipartite graph representation - can be stored in a system like this[1]. However, creating a generic metamodel framework, where every attribute, document, scope, authorization - and so on - are stored as hypergraph nodes could overwhelm the system, which results in slower

runtime, higher resource consumption, and therefore: overall higher costs.

In this paper we propose a new approach: we create a classification for the hypergraph model which support the ontology-like description of each node instance. This would lower the entry number of the graph representation significantly and also helps the understanding of the metamodel by creating custom views.

References

[1] A. BÉLECZKI, B. MOLNÁR, B. SARKADI-NAGY: *Storing Hypergraph-Based Data Models in Non-hypergraph Data Storage*, in: Modern Approaches for Intelligent Information and Database Systems. Studies in Computational Intelligence, vol 769. Ed. by A. SIEMINSKI, A. KOZIERKIEWICZ, M. NUNEZ, Q. HA, Cham: Springer, 2018, pp. 51–59.

[2] A. BLOKDIJK, P. BLOKDIJK: *Planning and Design of Information Systems*, London: Academic Press, 1987.

[3] B. MOLNÁR: *Applications of hypergraphs in informatics: a survey and opportunities for research*, Ann. Univ. Sci. Budapest. Sect. Comput 42 (2014), pp. 261–282.

[4] B. MOLNÁR, BENCZÚR: *The Application of Directed Hyper-Graphs for Analysis of Models of Information Systems*, Information Systems Modeling Based on Graph Theory (2022).

[5] B. MOLNÁR, A. BENCZÚR, A. BÉLECZKI: *A Model for Analysis and Design of Information Systems based on a Document Centric Approach*, in: Intelligent Information and Database Systems IIDS, Berlin: Springer-Verlag, 2016, pp. 290–299.

[6] B. MOLNÁR, A. BENCZÚR, A. BÉLECZKI: *Formal approach to modelling of modern information systems*, Int. J. Inf. Syst. Proj. Manag (2016), pp. 69–89.

[7] M. RUDOLF, M. PARADIES, C. BORNHÖVD, W. LEHNER: *The Graph Story of the SAP HANA Database* (2013), pp. 403–420.

[8] SAP: *SAP Cloud Application Programming*, URL: https://cap.cloud.sap.

[9] J. A. ZACHMAN: *A Framework for Information Systems Architecture*, IBM Systems Journal 26.3 (1987), pp. 276–292.