Designing an Open Architecture for the Creative Industry

Christian Zinke, Michael Becker, Stephan Klingner

Designing an Open Architecture for the Creative Industry

Christian Zinke, Michael Becker and Stephan Klingner

Institute for Applied Informatics at the University of Leipzig,
Hainstraße 11, Leipzig, Germany
{zinke|becker|klingner}@infai.org

Abstract. Companies in the Culture and Creative Industry are characterized by highly networked value chains. However, this value network lacks a profound support in terms of information technology and structure resulting in time-consuming and error-prone manual labor. To overcome these challenges, we conceptualize an application framework for creative industries. In particular a software architecture design and a data design will be proposed with the help of a design science research approach. The goal of the resulting application framework is to support the digital transformation of companies in the creative industry towards collaborative networks.

Keywords: Music Business, Framework, Design Science Research, Collaborative Networks.

1 Introduction

The Culture and Creative Industry (CCI) is an emerging sector, with progressively increased significance. In 2014 the core sector of CCI had 4.4% of total GDP (558 billion Euro) and 3.8% of total EU workforce [1]. The current trend of digitalization leads to significant changes in the value chain of CCI, especially in music business [2–5]. For example, through digitization music production does no longer require a recording studio and new digital distribution channels like music and video streaming portals weaken the influence of major labels [3]. While these developments are headed by major companies, small enterprises lack accurate digital support within their value chain. This became a significant challenge for the domain as recent studies show [6]:

- The majority of companies in CCI are very small and small enterprises (99 %)
- An average of 4.33 employee per enterprise renders an efficient division of work impossible.
- Cooperation and highly interactive networks are a substantial part of the value chain (93 % of all enterprises)

Thus, the CCI in general and music business in particular, are already highly networked. According to Ewig's [7] collaboration business transformation pyramid the described small enterprises can be classified as a network on organizational structure level. This is challenged by a lack of information structure and technology, which

includes the absence of unified and standardized exchange formats within the domain. Due to the wide variety of data formats, manual consolidation of data sets is necessary, which is time-consuming and error-prone. Thus, SMEs in music business have a data-based information structure – which means that there is no cross-functional information flow [8]. In addition, conducted questionnaires and workshops show high usage of proprietary software (like Excel) as well as a lot of manual work, which can be categorized as a very low information technology maturity (see Figure 1). Though the dimensions of Ewig's business collaboration pyramid have interaction effects with each other, the remainder of this paper focuses on information structure and information technology due to space limitations. We aim at conceptualizing an Information System (IS) and, thus, for our paper organizational questions are of minor importance. Analytically, small enterprises are on a preliminary stage to (goal-oriented) collaborative network organizations (CNOs) [9], lacking suitable information technology. Among others the forms of CNOs are virtual organizations or virtual enterprises [10].
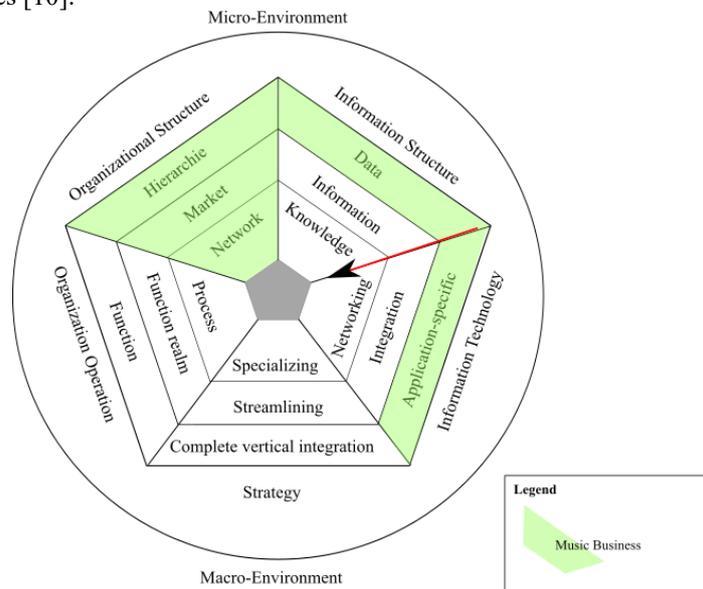

Fig. 1 Maturity of small enterprises within music business, according to Ewig [7]

The huge value network of the music industry and the resulting huge amount of requirements are challenging. Resulting mass-customization, from single worker (local workspace) via enterprise level (server) to collaborative network (cloud and web services), require a very flexible approach. However, the small stakeholders in music business lack the time, money and software solutions that fit their specific needs. Thus, open and scalable solutions are necessary to support the transformation process to CNOs. In summary, SME in music business need improved information structure and technology in order to evolve to a CNO (virtual network). Both will be addressed within this paper with the help of a design science research approach. The leading research question is, how to design an IS architecture to support digital transformation of SME in music industry?

2   Methodology

The presented paper is conducted through an IS Design Science Research approach. The major task and result of design science research is to create an artefact, which is "by definition, a purposeful IT artifact created to address an important organizational problem" [11]. We will focus on the design of a solution for the given problem. The detailed requirements – briefly described in chapter 1 – are derived from workshops and expert interviews with stakeholders of the music industry as well as analysis of available documentations (e.g. from label management or DDEX). Following common design science research methodologies for IS, the next step from requirements is the design of a possible solution, which will be presented next [12]. As a research in process this work is limited to a conceptual model as an artefact and cannot provide a concrete implementation or an evaluation [12]. Further on, the proposed concept will be used to iterate and collect additional requirements for real world application in order develop requirements that supports technical decisions on costs, complexity, performance and applicability. In order to evolve the focused dimensions of information structure and technology in music business, the paper focuses on Software Architecture Design and Data Design [13].

3   Design

Amongst other, Software Architecture Design can be defined as "a high-level abstraction of software systems" [14]. The presented requirements for such a software system architecture can be summarized as follows:

- Scalable from local applications to network solutions
- Support the digital exchange within music business value chain
- Flexible enhancement

As shown in **Fig. 2**, the proposed architecture of the framework consists of a runtime environment providing horizontal services. On top of the general environment, the installation of various small applications (so-called apps) shall be possible in order to support the individual value chain for each stakeholder within the value network. Each app provides specific functionalities that are required by one or more stakeholders.

The app management layer is responsible for integrating apps into the runtime environment. The apps are managed by the framework through simple policies (e.g. naming conventions, meta-data for apps (Dublin Core[1]), data access rules, dependencies) [15]. The management layer ensures the unified and standardized structuring of apps. In order to enable data exchange between different software systems and the platform interfaces are needed. Interfaces to other software systems are the first step moving to the integration stage on information technology level of the business collaboration pyramid [7]. However, the apps should be integrated on data side through common data schemas and protected through access rules.

---

[1] http://dublincore.org/

The message exchange layer enables the communication between different apps. Therefore, this layer may use the Music Business Ontology (MBO) [16] which specifies a unified message format. Based on this format, apps can access data provided by the data storage layer. The unified format allows for transformation of company-specific formats and, thus, reduces transaction costs.

Every company might maintain its own local installation of the framework. In addition to these local installations, a service directory containing apps has to be established. While an internal service directory only contains dedicated apps, an external directory which is available via internet connection, provides access to third party apps.
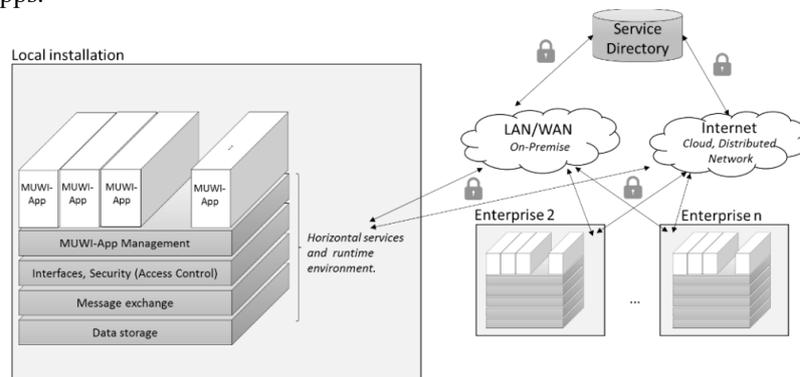

Fig. 2 Architecture to support small enterprises in music business

We will use the term framework, which consist of a set of elements and its relations, including economical, organizational and software-technical elements. An application framework is such a framework for other software artefacts (apps). An app is the smallest executable software artefact, which provides a specific functionality or service to the user. The proposed architecture of the app is described in Figure 3. In particular, an app has two main components: the user interface and the internal data processing services. In order to ensure compatibility to the platform three information are mandatory: a name (conforming to naming conventions), meta-data (see Dublin Core) and the dependencies. The dependencies are important to ensure functionality of the apps. In order to do so, we suggest that the used interfaces (e.g. to ERP or CRM systems), Linked Data (LD) Data Schema, LD Data Sources (e.g. SPARQL endpoints) and the platform version need to be described.

Besides software architecture, data design is an important aspect in transformation of companies towards collaborative networks. As stated above, open and scalable solutions are required and, thus, a linked data approach is proposed. Linked data is an approach applied for creating and distributing structured data on the Web with the help of RDF (Resource Description Framework) graphs. In order to make data accessible and enable automated processing, the Linked Data approach extents the vision of Semantic Web [17, 18]. Big players like BBC (British Broadcasting Corporation) already use this technology (with DBPedia[2] and MusicBrainz[3]) [19]. Linked data is

---

[2] http://wiki.dbpedia.org/

[3] https://github.com/LinkedBrainz/MusicBrainz-R2RML

suitable because heterogeneous data sources can be used across the value network, which will boost production efficiency of the small enterprises. However, Linked Data may lack in performance and practicability.
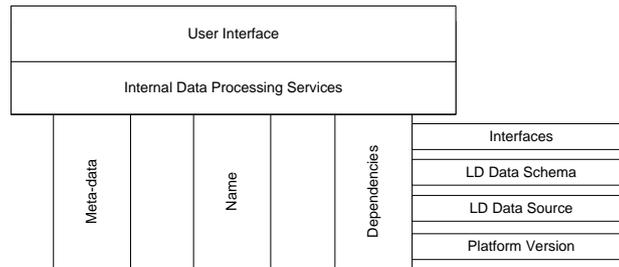

Fig. 3 MUWI-App architecture

The proposed framework allows to create flexible apps for specific purposes, which **access** and enhance information and **knowledge** within the **network** (on server or via internet) easily due to the linked data approach. On information technology side, the approach goes beyond the integration level, because it is not about integration of existing tools, it will combine information from various sources and supports information flows all over the value network. Due to the linked data approach, the provided information are stored as graphs and new knowledge can be easily discovered and extracted [20]. Thus, the approach has the potential to boost the music business to a knowledge level of information structure and network level on information technology side.

4   Related Work

This work has a strong link to software ecosystem (SE) research, which understands SE as "a set of actors functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artefacts." [21]. The MUWISTAR framework is a kind of a software ecosystem, and further a two-sided market solution, which addresses developers as well as customers [22]. Thus, the framework is a kind of a hub – an exchange platform [23], with many-to-many relations within the software supply chain [24]. While research on software ecosystem focuses on reports on procedures [25], the current work concentrates on software and data design. The framework allows for using apps as a kind of software component or plugin.

However, the research is also linked to software component research. There is a huge variety of components definitions and approaches, e.g. [26, 27]. By lack of space, we only introduce a small selection of software component literature. According to Heineman and Councill "[a] software component is a software element that conforms

to a component model and can be independently deployed and composed without modification according to a composition standard." [28]. The definition is extended by two other definitions: the component model and the software component infrastructure [27, 28]. The component model specifies the composition and interaction [28] – also called bindings [29]. The infrastructure ensures the compliant performance of the components [28], also described as system platform [29], which is the presented application framework. Software components approaches are quite similar to plug-in architectures, with practical differences in simplicity of configuration and administration [27].

5   Conclusion and Further Work

The progress of digitalization led to various challenges in the music business. While in the past the focus was on the substitution of physical distribution of music with digital alternatives, with the ascension of streaming and downloads today's challenges lie in the collection, processing and analyzing of data. To master the data, adequate software support is required.

Consequently, the paper proposes a concept for a software architecture and data design as a solution for the current challenges for the music business. This solution aims at providing a powerful, extendable software support, suitable also for the various individual companies and small and medium enterprises (SMEs). However, also organizational questions are important and one of the dimensions of collaborative networks, which have been also addressed in further works.

Yet this architecture is still mainly in the conceptual phase, so that various proposed technical approaches might be subject to changes to better suit the needs of the music business domain. As the concepts represent the best option from a theoretical perspective, it might be possible that requirements occur during implementation and practical evaluation, leading to architectural or technical changes. For example domain- or technology-specific parameters can lead to the substitution of the semantic web approach with common data format approaches, since technical decisions are always a trade-off between costs, complexity, performance and applicability. The technically most advanced solution does not inevitably represent the best approach. Therefore, the priority during the implementation is to create an easy usable and affordable system suiting the needs for the multitude of small and medium enterprises in the music business. This might require a revision of the technical concepts outlined in the paper.

Currently we are working on a first implementation of the presented concept as an open source solution. After developing a first prototype a workshop based evaluation with domain experts will follow. In the long run, we aim to establish a framework community within CCI, aiming at developers as well as users. At this moment the work is still in process and there are no technical artifacts developed yet. Following Gregor and Hevner [30], the solution maturity of the presented case is low, while the domain maturity is high – thus our solution have research opportunities and knowledge contribution for the domain (improvement).

References

1. Forum D'avignon: The economic contribution of the creative industries to EU GDP and employment. http://www.forum-avignon.org/sites/default/files/editeur/2014-Oct-European-Creative-Industry-GDP-Jobs-full-Report-ENG.pdf (2014)
2. Bauckhage, T.: Das Ende vom Lied. Zum Einfluss der Digitalisierung auf die internationale Musikindustrie. Stuttgart, 113f (2002)
3. Kusek, D., Leonhard, G., Lindsay, S.G.: The future of music. Manifesto for the digital music revolution. Berklee Press, Boston (2005)
4. Moreau, F.: The Disruptive Nature of Digitization: The Case of the Recorded Music Industry. International Journal of Arts Management **15**(2), 18–31 (2013)
5. Bourreau, M., Moreau, F., Gensollen, M.: The Digitization of the Recorded Music Industry. Impact on Business Models and Scenarios of Evolution. SSRN Journal (2008). doi: 10.2139/ssrn.1092138
6. Bertschek, I., Ohnemus, J., Erdsiek, D., Kimpeler, S., Rammer, C., Shala, E.: Monitoringbericht 2016: Ausgewählte wirtschaftliche Eckdaten der Kultur-und Kreativwirtschaft. Langfassung. ZEW-Gutachten und Forschungsberichte (2017)
7. Ewig, M.: Der Transformationsprozess zum collaborative business. Eine strategische, organisatorische und informatorische Betrachtung. Schriften zur Wirtschaftsinformatik, Bd. 17. Lang, Frankfurt am Main, New York (2006)
8. Ehrenberg, D., Ewig, M.: Collaborative Business — eine Herausforderung für die Wirtschaftsinformatik. In: Fink, K., Ploder, C. (eds.) Wirtschaftsinformatik als Schlüssel zum Unternehmenserfolg, pp. 43–71. DUV, Wiesbaden (2006)
9. Camarinha-Matos, L.M., Afsarmanesh, H.: On reference models for collaborative networked organizations. International Journal of Production Research (2008). doi: 10.1080/00207540701737666
10. Camarinha-Matos, L.M., Afsarmanesh, H.: Collaborative networks. A new scientific discipline. J Intell Manuf (2005). doi: 10.1007/s10845-005-1656-3
11. Vaishnavi, V., Kuechler, W.: Design Research in Information Systems (2004)
12. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. J. of Management Information Systems **24**(3), 45–77 (2008)
13. Chatterjee, S., Hevner, A.: Design Research in Information Systems. Theory and Practice, 1st edn. Integrated series in information systems, vol. 22. Springer US, Berlin (2010)
14. Solms, F.: Software Architecture Design (2014)
15. Weinreich, R., Sametinger, J.: Component models and component services: Concepts and principles. Component-Based Software Engineering: Putting Pieces Together, 22–64 (2001)
16. Schumacher, F., Gey, R., Klingner, S.: Semantics for the music industry. In: Sack, H., Filipowska, A., Lehmann, J., Hellmann, S. (eds.) the 10th International Conference, Leipzig, Germany, pp. 77–83. doi: 10.1145/2660517.2660531
17. Berners-Lee, T.: Linked Data. https://www.w3.org/DesignIssues/LinkedData.html (2009). Accessed 27 March 2017

18. Feigenbaum, L., Herman, I., Hongsermeier, T., Neumann, E., Stephens, S.: The Semantic Web in Action. Sci Am (2007). doi: 10.1038/scientificamerican1207-90

19. Bizer, C., Heath, T., Berners-Lee, T.: Linked data-the story so far. Semantic services, interoperability and web applications: emerging concepts, 205–227 (2009)

20. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - A crystallization point for the Web of Data. Web Semantics: Science, Services and Agents on the World Wide Web (2009). doi: 10.1016/j.websem.2009.07.002

21. Jansen, S., Brinkkemper, S., Finkelstein, A.: Business network management as a survival. Software ecosystems: analyzing and managing business networks in the software industry **29** (2013)

22. Roson, R., others: Two-sided markets: A tentative survey. Review of Network Economics **4**(2), 142–160 (2005)

23. Burkard, C., Widjaja, T., Buxmann, P.: Software Ecosystems. Bus Inf Syst Eng (2012). doi: 10.1007/s12599-011-0199-8

24. Ladid, L., Montes, A., Bruck, P.A., Ferri, F., Chbeir, R. (eds.): Proceedings of the Fifth International Conference on Management of Emergent Digital EcoSystems - MEDES '13. the Fifth International Conference, Luxembourg, Luxembourg, 28.10.2013 - 31.10.2013. ACM Press, New York, New York, USA (2013)

25. Manikas, K.: Revisiting software ecosystems Research. A longitudinal literature study. Journal of Systems and Software (2016). doi: 10.1016/j.jss.2016.02.003

26. Clemens, S., Dominik, G., Stephan, M.: Component software: Beyond object-oriented programming. Addison-Wesley (1998)

27. Duvigneau, M.: Konzeptionelle Modellierung von Plugin-Systemen mit Petrinetzen, vol. 4. Logos Verlag Berlin GmbH (2010)

28. Heineman, G.T., Councill, W.T.: Component-based software engineering. Addison-wesley Reading (2001)

29. Crnkovic, I., Sentilles, S., Vulgarakis, A., Chaudron, M.R.V.: A Classification Framework for Software Component Models. IEEE Transactions on Software Engineering (2011). doi: 10.1109/TSE.2010.83

30. Gregor, S., Hevner, A.R.: Positioning and Presenting Design Science Research for Maximum Impact. MIS Q **37**(2), 337–356 (2013)