Requirements Analysis for an Open iPaaS: Exploring the CSP, ISP, and SME View

Completed Research

Max-Marcel Theilig  
Technische Universität Berlin  
m.theilig@tu-berlin.de

Thorsten Pröhl  
Technische Universität Berlin  
t.proehl@tu-berlin.de

Rüdiger Zarnekow  
Technische Universität Berlin  
ruediger.zarnekow@tu-berlin.de

Abstract

Small and Mid-sized Enterprises (SMEs), in particular, suffer from cost-inefficient projects for the integration of cloud applications due to conceptual differences from local applications and a variety of related technical obstacles. Open source projects have been around for 20 years and tackle highly complex and comprehensive problems in a cost-effective manner. Following design science research, we gather and identify requirements for the development of an open integration Platform as a Service (iPaaS). We conducted a total of 20 interviews and obtained insights from cloud service providers (CSPs), integration service providers (ISPs), and SMEs. In this work, 641 requirements are collected and 67 sub-themes are identified as a result of extensive analysis. Furthermore, we outline 10 themes that align with three layers (provider, integration, and user) forming the target architecture of the underlying open source project in this paper. Finally, multiple insights are gathered for further employment, development, and research.

Keywords

Open Source, Integration Platform as a Service, SME, Requirements Engineering, Design Science.

Introduction

Historically grown, Small and Mid-sized Enterprises (SMEs) make predominant use of on-premise applications, which are connected to even more interfaces of other on-premise solutions or cloud services. Moreover, in a comprehensive connected network, the number of required point-to-point interfaces grows over proportionally with the number of connected applications. Therefore, a central hub solution is necessary. The concept of an integration Platform as a Service (iPaaS) (Potocnik and Juric 2012) is a recently investigated approach, which resembles the characteristics of an iPaaS for hybrid scenarios, connecting cloud and on-premise solutions. While there are various proprietary end-user solutions for connecting cloud services available on the market and large enterprises are able to finance high priced integration projects, there is a gap of solutions targeting integration problems of SMEs. This is mainly due to a number of requirements such as customizability, fast development, and an industry wide standard without vendor lock-in or licensing costs. Furthermore, emergent innovation can only be facilitated rather than planned by a platform owner (Tiwana 2014). Thus, to resolve this puzzle, an iPaaS, where multiple SMEs can bundle their engagement in an open source manner, seems to be a convenient solution. Following the open source idea, this paper aims to identify the requirements for an open iPaaS that come closest to what fulfills the major stakeholder’s (Cloud Service Providers, CSPs; Integration Service Providers, ISPs; and SMEs) needs to then reveal an artifact that constitutes a starting point for the following essential open source vision: “Release early. Release often. And listen to your customers” (Raymond 2001). Academics fill a research gap here because due to invested resources and alleged competitive disadvantage, closed integration projects often conceal their lessons learned regarding identified requirement themes. Towards this end, we conduct an exploratory requirements analysis for an independently funded project that targets the launch of an open integration platform connecting CSPs and SMEs. This case is particularly suited to
explore open approaches, as the platform owner is an independent and well-supported foundation that has already gathered significant experience from integration projects in the European SME market.

The remainder of this article is structured as follows: First, we present the theoretical background of integration projects and current open source activities in this field. Then, we introduce our methodological approach in the context of design science research. Next, based on this approach, we describe the requirements quantitatively and explain them from three different positions: addressed themes, stakeholder views, and open source relevance. The underlying open source project has already a vague idea of the target open iPaaS architecture and its components. Afterwards, this idea is described briefly and the derived themes are linked to this architecture. For practical relevance, the derived themes are linked and depicted in the acquired design artifact from project context. Finally, we outline the key findings, limitations, and directions for future research.

**Background and Related Work**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small and Mid-sized Enterprise (SME)</td>
<td>“[…] micro, small and medium-sized enterprises (SMEs) [are] made up of enterprises which employ fewer than 250 persons and […] turnover not exceeding EUR 50 million, and/or an annual balance sheet total not exceeding EUR 43 million.”</td>
<td>European Commission (2003)</td>
</tr>
<tr>
<td>Integration Service Provider (ISP)</td>
<td>These are service providers in business of integrating services of a given client (also referred to as service integrator).</td>
<td>Motahari Nezhad and Shwartz (2017)</td>
</tr>
<tr>
<td>Cloud Service Provider (CSP)</td>
<td>“Cloud computing […] is an emergent computing model. It facilitates easy access to remote resources, like computation […], which are provided by a third party, named Cloud Service Provider (CSP), by adopting the pay-on-demand paradigm.”</td>
<td>Information Resources Management Association (2016)</td>
</tr>
<tr>
<td>Openness</td>
<td>Openness is defined by the property rights of the platform technology. It is expressed by the platform supplier allowing platform users to interact with the platform, view, extend, change or spread its assets or components.</td>
<td>Muegge (2011); Anvaari and Jansen (2010)</td>
</tr>
<tr>
<td>Open Source</td>
<td>Defined by 10 criteria from the Open Source Initiative (OSI) that relate to the free distribution of the source code and license including derived work.</td>
<td>The Open Source Initiative (2007)</td>
</tr>
<tr>
<td>iPaaS</td>
<td>“iPaaS is a suite of cloud services that enable users to create manage, and govern integration flows connecting a wide range of applications without installing or managing any hardware or middleware.”</td>
<td>Serrano et al. (2014)</td>
</tr>
<tr>
<td>Application</td>
<td>An add-on software subsystem or service that connects to the platform to add functionality to it.</td>
<td>Parker et al. (2017)</td>
</tr>
<tr>
<td>Interfaces</td>
<td>Specifications and design rules that describe how the platform and applications interact and exchange information.</td>
<td>Tiwana (2014)</td>
</tr>
<tr>
<td>Contributor</td>
<td>Individuals or groups that develop one or more apps for the IT platform (also third-party developers or complementors).</td>
<td></td>
</tr>
</tbody>
</table>

**Table 1. Key Definitions of Integration Platforms**

In February 1998, the non-profit organization, Open Source Initiative, was founded by the open source pioneer Eric S. Raymond as the first of its kind. While in the first years that followed prominent arguments addressed the unavailability of open source software to commercial companies, there was an ideal imagination how open source works without the classic hierarchical structures and quality control (Raymond 2001). It took some years to develop its full potential and be accepted by their previous critics (Microsoft + Open Source 12-17-2016). Especially since from the business side, the concept of open source
seemed not applicable to IT or software business models. However, nowadays firms like SUSE, Red Hat or even Google show that it is possible to be commercially successful by adding payable services or building a business model around a technical base of open source driven projects. As recommended for every rigorous and resilient research project, we develop a theoretical pre-understanding of openness in integration platforms (Walsham 1995). This covers work on iPaaS and openness mechanisms as well as the current state of knowledge on IT integrations platforms in enterprise context. In general, IT platforms enable a digital business strategy for new technological means such as cloud computing or in-memory databases (Bharadwaj et al. 2013). For IT departments of larger enterprises, Enterprise Application Integration (EAI) aims to reduce application landscape complexity by connecting applications with a centralized and application-independent smart data management by means of standardized connectors as well as definable integration logic (Linthicum 2000). EAI has been vastly adopted by large enterprises but due to its cost intensive nature, it has not been successfully implemented in SME yet (e.g. Dell Boomi, Informatica Cloud, MuleSoft, and SAP HCI). In addition to Software as a Service (SaaS) and the arrival of a new service-oriented middleware technology standards, for example web services, there is a need to take integration to the next level (Linthicum 2006).

Nowadays, iPaaS allows users to integrate best-of-breed applications and resources from cloud services (Kleebeg et al. 2014). In fact, iPaaS solutions address the lower levels of data and functional integration within an organization (Ruh et al. 2002), which theoretically enables them to benefit from the potential of a larger ecosystem of SaaS solutions (Ondrus et al. 2015). However, when one takes a closer look at popular iPaaS implementations (e.g. Zapier or IFTTT), integrative capabilities of these solutions are restricted. Private users or SMEs may connect different applications and automate tasks in application scenarios but they are solely limited to the web and cloud sphere (Ng 2015). Furthermore, sufficiently skilled IT professionals have to use the provided SaaS applications and are not able to extend the system. Hence, they miss the benefits of integrating their important on-premise systems to closed iPaaS approaches. Consequently, an open source approach is designated to leverage the full integrative synergies that bring the CSP, ISP, and SME views together in an up-to-date environment incorporating all the advantages for such a multidimensional challenge. From a high level perspective, a very popular open source platform for cloud computing nowadays is OpenStack (Sefraoui et al. 2012). Beyond that, the economic side of open source projects is already well motivated (Riehle 2007). Taken together, we refer to the integration platform, its interfaces and complementary cloud applications, and the platform’s stakeholder as open platform ecosystem. The terminology related to integration platforms and what constitutes our area of definition in this study is summarized in Table 1.

Research Design

We adopt the design science research (DSR) method because this methodology focuses on rigorous creation and evaluation of IT artifacts to solve real-world organizational problems instead of simply describing or explaining them (Hevner et al. 2004; Peffers et al. 2007). The development and use of DSR is a reaction to the observation that behavioral driven research projects mostly lack relevance for applications in practice (Klein and Hirschheim 2003). Gregor and Jones (2007) explain that the accurately defined structure of these research projects have a high likelihood of resulting in a resilient artifact. Following the “design- and development-centered approach” of Peffers et al. (2007), we gather requirements and identify important themes of the target open iPaaS architecture (artifact) for the underlying open source project. We conducted several interviews from the three stakeholder perspectives and derived their respective requirements and design opinions. The identified themes for the design artifact qualifies as an “improvement” following the wording of Gregor and Hevner (2013) because the design is new but addresses an already known problem. Furthermore, this requirements analysis helps the open source project to concretize its platform idea and supports the development of the initial architecture.

The interview guidelines and enforcement questions are based on Kruse and Schmieder (2014). The whole guide was pretested with an expert from each stakeholder’s perspective (CSPs, ISPs, and SMEs) in order to verify what amount of time is needed for the interviews and whether the content coverage is appropriate. The interviews were of fairly unstructured nature so as to allow a thorough exploration of integration platform requirements for the respective organizations of each interviewee. Questions were asked directly but the interviews remained open for the investigation of other and related topics. The modular structure of the interview guide allows a prioritization and adaptation to the interviewee’s perspective according to
their competencies. In a semi-structured interview, people tend to not answer the question itself, but rather talk about what they associate with it and give a biased answer. Thus, this leads to many loosely coupled statements that potentially contain important information. For this reason, we linked the answers to the previous question, i.e. excerpt [05,Q3] relates to the 5th interview and the sequence that was answered after question 3.

The coding process as well as the content analysis generally follows Creswell and Poth (2017). Table 2 shows exemplary coding schemes for two interview excerpts. Moreover, the interesting segments are highlighted in order to identify the sub-themes, which were grouped to themes afterwards. These themes describe different aspects and topics around an open iPaaS.

<table>
<thead>
<tr>
<th>Interview Excerpt</th>
<th>Sub-Themes</th>
<th>Themes</th>
</tr>
</thead>
<tbody>
<tr>
<td>“We want to attach a mail service for sure.¹ On one hand the integration of a mail service¹ leads to the simplification that someone don’t has to copy the e-mail records into CRMs, e.g. SalesForces, anymore.² The e-mail conversation then would be accessible directly from the customer. On the other hand, we would prefer an interface for gender, name, company, and e-mail addresses to mail-campaign tools, like MailChimp, to realize campaigns easier;¹ And to avoid data exchange via xls or csv files.” [05,Q5]</td>
<td>¹) Mail Service (i.e. MailChimp)</td>
<td>Interface</td>
</tr>
<tr>
<td>“[…] implement modern concepts, like REST API⁴ or XAML tokens. [We had] many integration projects, from that we know there is a need for one central integration service [composited of various Microservices]. These Microservices need to centrally encrypt and protect the data stream from on-premise to cloud.⁵ In our new integration projects [with cloud services] the demand for similar [central platform, encryption] is high.” [13,Q5]</td>
<td>³) Microservice</td>
<td>Architecture</td>
</tr>
<tr>
<td></td>
<td>⁴) REST</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Exemplary Coding Scheme with Highlighted Segments

We sent out project descriptions and interview invitations to 40 companies, which we have selected according to a set of criteria: amount of employees, turnover, cloud experience, and integration attempts. Additionally, we were referred to suitable candidates from within the project consortium. Overall, we had a response rate of around 30%, with fewer responses for the desired SMEs perspective. Finally, we conducted 20 interviews, which cover the three perspectives. Some interview partners were able to give insights for two perspectives. A phone interview was conducted for each company with a duration varying between 24 and 82 minutes. Each interview was recorded with the consent of the partner for analysis purposes. Finally, we acquire evaluable insights for three views: 13 for CSPs, 5 for ISPs, and 5 for SMEs.

The interview partners represent companies of different sizes. The interviewee of the largest organization, a multinational organization in the field of telecommunication, was a partner ecosystem manager. Whereas the interviewee of the smallest company, a SaaS company in the economy of trust, was the CEO. The large portion of the organizations fit the SME definition, with representatives having a semi-technical background, i.e. Head of B2B Marketing, CTO, senior engineer or managers. These interviewed companies are located in Germany, Austria, and Switzerland (DACH) but run their business and sell their services across all European countries. Among these companies, two have substantial business in the US. Every company consumes cloud services and has already collected experience from integration projects. Some companies use only a small number of APIs to synchronize their data between different applications, whereas others started their own small integration buses or internal data integration platforms. The interviewed organization covered the following notable domains: telecommunication, infrastructure services, IT security solutions, ERP solutions, CRM and contact solutions, community supply management, logistics, and HR management. All interviewees were provided with a small project description and were directly made aware of the fact that the project is to be open source and that openness is a primary objective. The interviews have been conducted in German and transformed to English.
Findings and Interpretation

In this section, we present the results of our data collection as well as the functional and non-functional requirements of the open iPaaS. In fact, we conducted two rounds of coding. We first provide the descriptive statistics with regards to the results of our coding processes, which is followed by the description of the derived functional requirement themes. The open coding leads to 535 codes that are related to 641 interview quotes. 62 sub-themes were identified by using an axial coding. These sub-themes represent open codes for similar aspects of requirements and led to 10 themes. Figure 1 depicts the distribution of the interview quotes within the 10 themes.

![Figure 1. Overview of Requirement Distribution](image)

**Interface**: The main content of this theme falls in line with the definition from Table 1. Noteworthy sub-themes are REST, OAuth, and Active Directory. This theme has many requirements regarding explicit interfaces for applications or software providers, e.g. SAP, ZenDesk, SAGE or SalesForce. Interviewees made it very clear that there also has to be a downward compatibility, e.g. in terms of file formats such as csv, xlsx, xml or pdf. Lastly, we found a high demand for Remote Procedure Calls (RPC), since many interviewees feel a need for a representation of their business processes. Most interviewees see the theme of interfaces central for an open iPaaS solution as they express their belief in a critical mass that has to be reached thereby establishing the necessary “gold standard”. Which in turn facilitates the development of interfaces and is supposed to trigger the intended effect of openness from their point of view.

**Synchronization**: Very close to the interface requirements, synchronization mechanism need to be employed to constrain the phase transitions of a collection of processes in accordance with specifications. The topics addressed here include the definition of a leading system and master data versus transactional data synchronization. The consensus is that master data synchronization is a necessity, since all business critical overlying processes (e.g. invoicing) rely on it. When master data synchronization of the leading system is resilient, error tolerance for transactional data has only medium priority. Timely event-related synchronization is often required for transactional data, whereas master data is not perceived as time critical. In addition, reliable master data management was favored over instantaneous exchange by almost all experts.

**Architecture**: Extracted requirements cover three topics: data organization, Microservice architecture, and hardware. They are concerned with the high-level structure of the software, e.g. bulk processing versus continuous streaming, the relationship among software elements, and the role management. On the other hand, only few requirements regarding technical architecture came up regarding topics such as failover design, load balancing, server mirroring or high availability. Due to the lack for designable requests, some concrete requirements regarding inclusion of Public-Key-Infrastructure, Single Sign-On, reverse proxy, and SSL off-loading were specified.

**Compliance**: For interviewees, compliance, being largely about ensuring that business processes are executed as expected, falls in line with their goal of IT security: Meeting all of its business objectives by implementing systems considering IT related regulations and risks for the organization, its partners, and customers. Accordingly, all requirements in that theme revolve around regulatory circumstances, due to usage of an open iPaaS, being no thread to their product or business objectives. The implementation of a caching algorithm (due to reasons related to data backup, conflict resolution or delivery conformation) in the open iPaaS data hub is a high concern since regulatory conformance of data storage is vague. Finally, several interview partners emphasized message and communication encryption.
Business Model: Most quotes refer to the whole picture of a business model framework that needs to be simple enough to provide guidance in a design oriented, strategic, and analytic way for the organizations to apply their own business model. Interviewees specified requirements referring 1) customers and segments, 2) product and service offering, 3) the distribution model and partners, and 4) the model their software usually employs to generate revenue and profit. This falls in line with Kontio et al. (2005). Surprisingly most of them, after excogitating over the requirements, reasoned that the open iPaaS itself should not have a business model towards their cloud customers. Making use of the open iPaaS should be free. Even more, free use seems to be mandatory to reach a critical mass of customers and connected services. Therefore, hosting and maintenance costs of the ISP might be divided among the participating CSPs, such as a membership fee, instead of charging the consumer directly, here a SME.

User Experience: Interviewees make reservations that refer to encounters with an open iPaaS – not only active, personal use, but also from a more passive point of view. E.g., the stakeholders require an open iPaaS to enable their customers to map underlying business processes in an easy and convenient manner.

Communication: Interviewees report many problems in coordination and communication with previous integration projects, especially in cloud cases. This is followed by a negative impact on project quality, when it is not clear who is responsible for a particular area and tasks as well as responsibilities cannot be communicated properly. The cloud stakeholders require reducing the duplication of effort and enhancing coordination related to development and maintenance. To solve these issues on a higher level, the demand for a coordinated initiative to establish one single open iPaaS as the “gold standard” is high.

Documentation: We find that mainly SMEs and CSPs care about general domains of documentation: 1) system overview, 2) system components, and 3) system details. Interviewees claim that these domains are essential to collaborate on an open source platform. In addition, they suggest that a low level of documentation will even have an adversarial impact on the interest in the project in general. CSPs draw attention to the fact that their own conventions, documentation, and support guideline needs to be referenced to provide an ideal level of support quality.

Resources: Requirements associated with resources refer to persons, assets, and materials, which shall be used to develop, establish, and maintain an open iPaaS. Surprisingly, no direct requirements for capital where mentioned. Interviewees require sufficient resources to fulfill the service level agreement (SLA) towards their customers, including availability, capacity, response time, and support capability. It follows the logic of their concerns about performance respectively insufficient resource of an open iPaaS.

Testing & Deployment: Very few of the requirements are in the theme of Testing and Deployment. Since the underlying open source project is at an early stage this is plausible, because most open source projects do not require test planning (Zhao and Elbaum 2003). In the interviewees perception, testing should be taken over by volunteers and a system of voting, compared to the Apache project, is required before deployment. The main concern is a slow deployment process on the open iPaaS side that may affect the perceived service quality of the connected software. Therefore, the release management process has to reflect much of the standard open source software development principles and practices (Erenkrantz 2003).

It is also worth mentioning that requirements regarding Testing and Deployment were only raised from the CSP perspective.

Stakeholder Views

Regarding the different stakeholders, we gather 361 (56%) requirements from CSP perspective, 126 (20%) requirements from ISP perspective, and 154 (24%) requirements from SME perspective. We found that grouping them by the corresponding layer (see Figure 2 in the following section), the User Layer is underrepresented with 38 (6%) of all requirements. Thus, the main requirements from the stakeholder’s perspectives are listed as follows:

CSP View: A variety of available connectors (critical mass) has to be reached in order to achieve a solid network effect. Statutory provisions have to be addressed accordingly. The concept needs to be future-proof, due to the sunk costs associated with a possible changeover.

ISP View: A suitable business model and advisory is needed because interviewees have very limited experience with open source business models. The open iPaaS solution needs to be flexible and easily scalable to fit the available IT capabilities. By hosting the open iPaaS, ISPs believe that the responsibility is inherited and, therefore, they require a well-documented environment.

SME View: At the very least, currently used interfaces have to be supported for facilitating the initial migration to open iPaaS. Agreeing on a standard for synchronization is important for SMEs. Reliable service
quality is mandatory, since their business relies on it. Moreover, SMEs mention that committing to a comprehensive solution increase their fear of a total blackout.

**Open Source Relevance**

After focusing on the stakeholder views, we evaluated all interviews regarding the open source aspects. The following advantages revisited by the interviewees cover:

- Using and sharing open source components has the potential to reduce development time and effort, technological heterogeneity, and licensing costs.
- Quality assurance and bug detection is ensured by a broad community of independent contributors.
- The community facilitates user innovations for a superior product and ensures competent support.
- Cloud adoption costs are reduced and the risk of a vendor lock-in is repelled by mutual dependency.
- A higher compatibility and interoperability follows an established open iPaaS “gold standard”.

While there are a lot of advantages, stakeholders also mentioned the following potential downsides:

- There are increased legal risk due to uncertain intellectual property issues, potential regulatory violations, and undefined areas for an open iPaaS within jurisdiction.
- Contributing value without predictable business outcome is perceived as a risk.
- There is a lack of evidence for superior open source software quality.
- Unsupported code is a significant issue because removal of existing functionalities is unpredictable.

A very common exemplary excerpt and benefits that are addressed by an open iPaaS of each perspective are depicted in Table 3.

<table>
<thead>
<tr>
<th>View</th>
<th>Interview Excerpt</th>
<th>Addressed Benefits</th>
</tr>
</thead>
</table>
| CSP  | As a small organization, you have to face the competition anyway. Therefore, it is the best to provide an open source approach without fear of customer migration. [15.Q3] | • Reach broad customer base  
• Reduced initial acquisition effort  
• “Gold standard” origination |
| ISP  | Customers mainly claim not about the delivered product (interface) but about the integration process. The project is always too expensive, too complex, and takes too long. [10.Q4] | • Short notice availability  
• Established development  
• Reduced interface heterogeneity |
| SME  | All in one suites come with high license costs, maintenance fees, and a vendor lock-in. An open source platform would be an ideal solution to increase the degrees of freedom in our IT portfolio. [14.Q5] | • Cost reduction  
• Repel vendor lock-in  
• Solution flexibility |

Table 3. Example of Stakeholder Views and Open Source Benefits

**Insights for Underlying Architecture**

The deduced themes are mapped in Figure 2 alongside with the architecture idea of the underlying open source project. Furthermore, this figure presents three layers grouping the 10 themes into overlapping fields. The ISP provides cloud applications from the Provider Layer by connecting these services via API. In order to offer an extensive portfolio of services they are able to extend the open iPaaS on their own and take advantage of previously contributed work. Additionally, the ISP hosts the provided integration and data hubs in a convenient way. By instantiating its own open iPaaS on a private infrastructure, a SME may even become its own ISP (Internal IT Department). SME are then no longer constrained by a limited application portfolio or missing cloud integrations. They may encourage own employees to contribute to the open iPaaS by developing interfaces or take advantage of independently developed content.

The following shows the insights we find necessary to establish an open iPaaS:

*Insights regarding the Provider Layer:* The platform governance for participating CSPs should 1) be an extended Service-Oriented Architecture governance as it, 2) covers cloud application performance,
backward compatibility, continuous support, and security, 3) be able to detect quality of service and SLA violations, and 4) inform users as well as CSPs about service incidents. Furthermore, an open iPaaS is supposed to provide interfaces for performance and business activity monitoring and, on the technical side, supplies users with cloud-oriented bandwidth with metrics defined across multiple cloud applications. In conclusion, critical success factors are also valid in open source.

**Insights regarding the Integration Layer:** On-premise apps and cloud applications need to expose their functionalities via APIs in order to connect with an open iPaaS. The open iPaaS is required to provide the ability to connect those different systems in hybrid scenarios. Interfaces also should enable service orchestration into business processes composed of diverse cloud services or on-premise applications by allowing RPCs. SME customers expect a flexible business model of an open iPaaS that fits their accompanying processes. Hosting and maintenance costs are supposed to be shared among ISP and CSP. Developing a suitable open participation model is crucial for the success due to the generation of a critical mass of contributors, associated CSPs, and provided resources of the ISP. Data from different sources should always be complete and consistent. In other words, an open iPaaS should offer data synchronization mechanisms to provide automatic data transformation and migration. As data integrity and IT security are considerably crucial issues, an open iPaaS should also ensure that information retrieval, extraction, mediation, transformation, and propagation is done in a secure manner. On the other hand, the necessary storage and backup systems have to be compliant with regulatory demand.

**Insights regarding the User Layer:** The platform users, SMEs, have to be able to subscribe a service directly through the hosted instantiation of the open iPaaS as they would do to any other cloud service, i.e. over a cloud marketplace. Since access to cloud resources is perceived as more limited than on-premise resources, external APIs have to be attachable. Hence, in the case where needed functionalities are not being provided, documentation of an open iPaaS has to be comprehensive for SME associated developers to contribute and eradicate capability gaps. These contributors increase the functionality by adapting software connectors for used on-premise applications or adding mapping schemes for new cloud services.

**Figure 2. Architecture Idea of Underlying Open Source Project, Covered Requirement Themes, and Corresponding Layers**

**Conclusion**

This research identifies requirements for development and early release of an open iPaaS for different stakeholders. To be able to derive such requirements, we addressed the question: What requirements do CSPs, ISPs, and SMEs have regarding an open iPaaS? We cover three stakeholder views (CSP, ISP, and
SME) by conducting 20 interviews. These interviews lead to 641 requirements, 67 sub-themes, and 10 themes.

Several limitations may affect our results. The first limitation is regarding sampling and number of interviews. The group of interviewees is drawn from organizations only in the cloud domain and located in the DACH region. While we believe that active cloud enterprises are representative for organizations selecting and applying open iPaaS solutions, further generalization towards undiscovered organizations employing only on-premise solutions is recommended. Moreover, while the amount of 641 requirements from the 20 participants is limited, it is, however, appropriate for research studies at the current maturity stage of the iPaaS domain. With regards to the sample, examination of the coverage statistics presented in Figure 1 and underrepresented User Layer shows that there is an imbalance between the number of requirements per corresponding layer and also per conducted view. This phenomenon is likely caused by the particular interest compositions per perspective of the interviewed enterprises and by the fact that the main product of the enterprises itself was located in the cloud business area. Therefore, we recommend future studies and developmental iterations to incorporate a larger amount of requirements, preferably from a SME point of view that is not yet strongly engaged in the cloud business. A mix of different industries to further validate the current set of requirements as well as to compare between different industries with the goal of providing situational sets of functional requirements would be advantageous. Lastly, in this study, we had a semi-structured interview approach for identifying requirements related to an open iPaaS solution. While this explorative method is appropriate for initial scope of the project, a quantitative approach such as a survey has to be performed in a future study in order to validate and prioritize the identified requirements. In this context, additional use of open questions will reveal further requirements.

From a scientific perspective, the contribution in this paper provides a fundament for further research by the gathered requirements, possible survey questions, and items for validation. From a practical angle, an organization, especially in the open source context, is able to implement the proposed design of an open iPaaS that considers the shown key modules and themes. Organizations, foundations or ISPs benefit from the requirements and themes that guide the process of prioritizing CSPs and their services. Moreover, they implement core functionalities such as data synchronization mechanisms and provide resources to the open iPaaS solution. This facilitates the development of an open iPaaS and justifies their juxtaposing against current closed iPaaS approaches in terms of business goals and integrative abilities.

Acknowledgements

The work presented in this paper has been funded by the Federal Ministry of Economics and Technology of Germany in the project Open Integration Hub (acronym: OIH, project number 01MT17002G). The authors wish to express their gratitude to Frank Türling and the Cloud Ecosystem e.V. for their assistance and commitment to the project.

REFERENCES


Microsoft + Open Source. 12-17-2016. Microsoft joins the Linux Foundation.


