1 Introduction

Globalization and digitization pose continuous challenges to already established processes in government and administration. The standardization of existing processes to improve cooperation between institutions, to reduce overall costs or to increase efficiency is a part of daily business on several levels: global (e.g. worldwide / EU), national (e.g. in Germany or North Rhine-Westphalia) or local (e.g. Aachen). Over the past years this led to IT infrastructures and applications that eventually cumulated into ubiquitous IT services that support a variety of processes. However, the supported business processes are subject to constant change and are regularly updated in accordance to new regulations or due to optimization. Each change to a previously defined process can therefore result in several changes within IT services. The resulting frequency, impact of change and variety of processes in turn require flexible IT services and applications supporting these processes.

1.1 Motivation

Due to increased mobility and the rising number of students and researchers, universities find themselves confronted with increased demands on IT services. However, the issue does not only affect the organizational level. Students and employees have already changed their requirements for IT service providers and thus also for universities in general. Not only IT services and applications are becoming increasingly important for the processes of universities but also the everyday life of students and employees. This in turn makes the quality of IT services a location advantage and leads to increased competition among the universities for the best and most attractive IT service portfolio. A development shown in the Horizon Report series, one of the most regarded studies concerning the development of education, for several years [2]–[4]. To meet these developments, these reports have consecutively identified upcoming major challenges in the higher education sector as follows:

- Bring Your Own Device (BYOD)
- The Internet-of-Things

- Improving Digital Literacy
- Adaptive, Personalized and Mobile Learning
- Balancing Our Connected and Unconnected Lives
- Improving Digital Literacy
- Artificial Intelligence

In order for IT services of universities to remain operational, competitive and meet the expectations of governments, researchers and students, IT service providers are required to manage a growing, heterogeneous service landscape. While many standardized solutions exist, tailoring the services towards local needs and integrating the services with each other to create a consistent user experience requires an in depth knowledge of business processes of the university. For general process oriented management systems several methodologies and certifications, for example according to IT Infrastructure Library (ITIL) and ISO 20000 [5] or according to the ISO 9000 standards family [6], are available and are widely used in practice. Instead of focusing on the business processes to be supported, development processes for IT services and applications follow specific software engineering methodologies that rather aim for software quality than the quality of the supported business processes. Organizations that act as an IT service provider and that also tailor applications to fit to business processes, are therefore left in a tension field between their own processes for management, operation and engineering of services as well as the business processes they aim to support.

Increased digital literacy among the users additionally raises the need towards integration of centrally provided IT services within local and decentralized, environments of users. This trend becomes even more evident when considering that data driven and machine learning applications are gradually becoming affordable and more common in the local environment of the users. As press reports indicate there some prominent approaches towards integrating off the shelf machine learning with local business processes in the areas of research and education; some were more successful [7] than others [8]. Especially within environments that depend on business processes and data modeled in existing system landscapes application technologies for individualization and automation requires significant work to integrate data and make it accessible by applications.

Considering processes within the university context there are two predominant areas of processes: supporting scientific research and supporting teaching and learning. IT systems of central and decentralized service providers that try to digitize and improve processes in any of these areas are generally referred to as *eScience* or *eLearning* systems respectively. Enabling this highly specialized system landscape to interact within a process oriented environment is the fundamental challenge to provide added value services for students and employees of the university.

1.2 Related Research

Even though the regarded processes have a specific implementation at the university, it is obvious that the mentioned complex of challenges is neither specific to RWTH Aachen University nor to a university's context in general but generalizes to all kinds of processes oriented businesses. Various research groups and projects reduce complexity by modeling and try to find iterative or partial solutions to some facet of the aforementioned challenges.

The research area of *Ubiquitous Computing* focuses on technical and socio-technical challenges that arise from the decreasing size and increasing availability of computing services in general. Some research groups already focus how universities are influenced by this development. Supporting business processes with IT systems requires transferring to a consistent, often machine readable, description as discussed by the research area of *Business Process Modeling*. To successfully build IT systems supporting these processes, the research area of *IT System Architectures* defines and evaluates different variants of their implementation. These implementations of business processes potentially deal with personal or confidential information. Additionally to assessing their architectural properties, the research area *IT Security* evaluates the security of process supporting systems.

1.2.1 Ubiquitous Computing in University Processes

The rise of fully connected applications and always available and low-cost wireless connections have changed the view how computing is perceived in our daily lives. Processes in organizational and personal context are exceedingly drained by computing services. Weiser therefore coins the term *Ubiquitous Computing*; stating that eventually all computing processes are taking place in the background and without explicit interaction of the user [9]. This development is accompanied by an ever decreasing size of connected computing devices. While the Internet was initially intended for social interactions, currently devices and human users can freely participate in interactions. However, Ciortea *et al.* also require more specialized software architectures to account for this changed usage behavior [10]. One of these requirements is that information is available in a machine readable format that can be interpreted and accessed by devices easily as proposed by Gerber *et al.* and Han *et al.* [11], [12].

Within university context the research area *prevasive university* specializes on ubiquitious computing of research and educational processes. The fields of eLearning and eScience are therefore in focus of several research groups and projects. Juling describes that mobility and ubiquity of information technology is a sign of the times. Global accessibility, reachability and therefore mobility have become part of our daily life [13]. He demands that existing and future computing systems have to be designed in a way that they can actively use information in collaborative, flexible, reliable and secure ways. Transferred to universities he states that eScience requires a comprehensive digital infrastructure. He further requires to foster the creation, pooling and usability of IT systems, resources and infrastructure to provide additional value to researchers in different scientific disciplines. The use cases by Van Garderen and Kirsten *et al.* further show how such IT systems can support scientific processes at universities [14], [15].

Barkhuus *et al.* discuss the different needs of students related to ubiquitous services offered by their university. They focused on the role of technology in the classroom as well as the social arrangements in which the students were situated [16]. While the technological basis has severely changed since the study in 2004, the social roles, relationships and responsibilities of the students are mostly comparable. The most important issues of the pervasive university pointed out are organizational like reliability and ease of use. Engaging students when building process supporting IT services can further increase satisfaction as shown by Kupila *et al.* [17]. Additionally, as pointed out by Gikas *et al.*, matching IT services with students' devices plays an important role within the supported processes [18].

During the past years mobile and web technology advanced drastically. Lucke *et al.* introduce a model for the pervasive university that takes the heterogeneous existing system landscape into account [19]. An example a prototype that is based on some university business processes in the field of eLearning is proposed and shown to be integrated into other systems. Lehsten *et al.* extend this model with a concept to integrate multiple context aware services [20]. The context of the user is aggregated by a central services and distributed to the attached services. This points out the issue of a heterogeneous system landscape that needs to be integrated to support ubiquitous processes. As shown by Schmees *et al.* and Haim many universities are facing this problem and had to develop various local solutions to handle it [21], [22]. Mincer-Daszkiewicz and Barata *et al.* show two example architectures for Campus Management as a supporting process of Education and Research [23], [24]. While Barata *et al.* show the requirements on a newly developed Student Lifecycle Management (SLM) system, Mincer-Daszkiewicz adopt an existing SLM system to changing processes and higher load due to increasing number of users. While their implementations do not follow any explicitly modeled processes they raise the issue of security of personal data when crossing process and system boundaries.

1.2.2 Business Process Management

Continuous improvement of organizations requires to model the ways in which they provide value to their customers or users. Business Process Management (BPM) therefore is used to build a common understanding on these operations and workflows within an organization in a way that can be heeded by employees of the organization. Resulting comprehensive models can then be applied by IT system architects building infrastructures to support the processes and therefore allow further digitization of the organization as described by van der Aalst *et al.* [25].

Modeled business processes furthermore form the basis for change management and optimization of business processes. Schulte *et al.* further shows that the close connection to IT systems allows optimizing used resources which becomes explicitly beneficial in elastic cloud environments [26].

Apart from the technical description of a system BPM also reflects procedures carried out by employees. Giaglis *et al.* propose that modeled business processes should therefore fit two dimensions: on the one hand technical requirements like formal, quantitative and stochastic modeling or documentation, adaptability and objectivity; on the other hand social requirements like feasibility, communication or user friendliness [27]. Based on these requirements they show that BPM and its validation by simulation are valuable for modeling intra- and inter organizational processes. Wißotzki additionally points out that transferring business processes to IT systems often can only be performed by domain experts but is still an underestimated in many environments [28].

Aguilar-Savén shows that several languages with different properties have emerged in order to describe business process models [29]. The author shows that a multitude of BPM languages and supporting tools already exist. Additionally, a proposed classification framework provides some guidance for business analysts or IT system architects to chose among the available languages. Sandkuhl *et al.*, however, point out that BPM needs to become a common practice providing additional value for the organization and not a purpose on its own [30]. They envision that also non-experts model processes, sometimes without knowing. Modeled processes are viewed as a kind of captured knowledge that can be shared within the organization, is used for decision making but has fewer requirements towards formality and completeness.

1.2.3 Architectures for Process Supporting IT Systems

Future distributed IT systems therefore need to consider that more devices are using the Internet to communicate. Within the research area of the so called *Internet-of-Things* Miorandi *et al.* identify heterogeneity, scalability, semantic interoperability and privacy preservation as some of the key challenges that IT systems need to support [31]. Facing some of these challenges Han *et al.* propose a full-IP ecosystem using the Representational State Transfer (REST) paradigm that is already predominant in modern Internet applications [12]. The authors further claim that composing services from existing workflows allow novel and creative approaches for future applications and devices. As shown by O'Reilly, the basic design pattern of Service Oriented Architectures (SOAs) that bases software on independent services that in turn base on business processes, is already part of many common Internet applications [32]. Commodity Internet standards like Hyper Text Transfer Protocol (HTTP) make these kinds of services widely applicable but also led to competing implementations as pointed out by Tihomirovs *et al.* [33].

Using standardized protocols and architectures distributed systems provide some features required by Miorandi *et al.* Nevertheless, Taheriyan *et al.* point out that semantic interoperability need particular attention [34]. The authors propose a method to integrate SOAs with each other using linked data. While manually lifting existing SOA is connected to high costs, they propose a semi-automated approach that infers semantic information by using the existing services. Especially when proving access to complex functionalities being understandable makes services reusable. To maximize reusability, Zhu claims that internal complexity and therefore the value of a service should be high while access to it should be easy and cheap [35]. A common understanding of services furthermore is required for the integration of services within an organization or across organization boundaries as shown by Ebert *et al.* [36]. Composing and integrating services from various IT systems may result in additional latencies as shown by Göb *et al.* reducing user experience with slow network connections or small compute devices [37]. Using prefetching middlewares as part of the SOA, the authors are able to actively work against the issue effectively shifting compute or network intensive workloads from users' devices to a server infrastructure.

Gruber *et al.* points out that IT systems are becoming increasingly complex and state that this complexity needs to be properly managed to retain maintainability of the system [38]. Kecskemeti *et al.*, Viennot *et al.* and Salvatierra *et al.* show approaches of technically decomposing IT systems to an SOA to foster scalability of the infrastructures [39]–[41]. *Microservice* architecture as, for example, proposed by Namiot *et al.* is used to reduce dependencies and allow better scalability of SOAs [42]. By breaking the SOA into small, independent services each of which fulfills only few well defended tasks. Namiot *et al.* further show that traditional SOA are scaled by duplication or partitioning, the independence of separate modules instantiated as *Microservices* also offer new possibilities to scale the system by functional decomposition. This is especially important as more and more devices are accessing the services. Cloud technologies furthermore allow scaling of IT systems without direct dependency to physical hardware or on site resources.

Mircea *et al.* and Low *et al.* conclude migration of existing IT systems is especially an issue for complex process supporting systems within large organizations [43], [44]. As remarked by Serrano *et al.* existing legacy systems are usually not well documented and reengineering is likely expensive [45]. Instead, the authors claim it is desirable to wrap existing systems with a service layer to allow usage of modern technologies. Newly developed applications, however, often already adapt to new infrastructural and architectural requirements [46]–[48]. The introduced multitude of small services and abstraction layers additionally raise new challenges for software development processes in traditional environments. Schleicher *et al.* and Zimmermann *et al.* show in their works that an additional dependency is introduced as virtualization and cloud environments pose new technical and organizational requirements to the deployed software and development teams [49], [50]. To meet some of these technical challenges Toffetti *et al.* propose a service registry allowing self organization [51], while Balalaie *et al.* show how development and release processes have to be changed to fit to *Microservice* architectures [52].

A shown, breaking down complex systems into independently maintained units can ease the development process for the separated services. After all, services in their entirety need to provide value to their users. van der Aalst *et al.* and Schonenberg *et al.* therefore require that services need to be interoperable and thus can participate in the same organizational processes [53], [54]. To lift existing databases to an interoperable state Wiederhold proposed a mediator architecture [55]. Mediators are services that collect information from these databases and convert the contained data set to a common semantics. Papakonstantinou *et al.* and Wiederhold *et al.* have proposed several computational operations that need to be supported on the datasets to perform this [56], [57]. Over the last years there were several systems that based on the mediator architecture that are specific to certain disciplines for example web crawling [58], bioinformatics [59] or learning repositories [60]. While these initial works focus on databases most results can be transferred to SOA: Barthe-Delanoë *et al.* show how a mediator information system can be used to achieve and evaluate interoperability of process supporting services [61]. Based on these findings, the research group around Bénaben *et al.* have develeoped and gradually refined a process model that allows managing orchestration of processes, data conversion and service selection within a mediator information system [62]–[64].

Apart from technical challenges, Aerts *et al.* claim that software architectures, modeled processes and business needs have to be aligned in order to provide the maximum value to the users [65]. The authors regard an IT system as a composition of layers decoupling technological, application and business models that need to be integrated. To overcome gaps between modeled processes and the supporting software Zimmermann *et al.* propose an integrated SOA that allows adoption of changing business processes and needs [66]. The authors translate architectural domains into a service oriented architecture ontology that allows automatic inference and therefore supports decision processes within a practical implementation. In subsequent works Zimmermann *et al.* and Kaidalova *et al.* show how their architecture adapts to the change of business needs induced by recent developments like the *Internet-of-Things* or micro-granular architectures like *Microservices* in various contexts [50], [67]–[69]. To practically achieve this alignment Montesi proposes an approach unifying implementation of SOA and BPM to create process-aware services [70].

1.2.4 Evaluation of Process Supporting IT Systems

Process supporting IT systems mostly result from a software engineering process. Tarr *et al.* introduce Separation of Concerns (SoC) as a method for evaluating software architectures in general [71]. The authors introduce three goals, *Impact-of-Change*, *Reuse* and *Traceability*, that should be pursued across four dimensions of concerns: *Feature*, *Unit-of-Change*, *Customization* and *Data*. Even though the concepts behind SOAs allow separating services according to these dimensions, Mateos *et al.* show that also applications using the SOA need an additional layer of abstraction to avoid coupling to specific service implementations [72]. Bianco *et al.* formulate fundamental design questions to be considered during the software engineering process and map them to potential business risks [73]. A catalog of detailed questions presents a guideline for technical evaluation of SOAs.

The model introduced by Rathfelder *et al.* does not only evaluate the technical maturity of an SOA but also considers non-technical domains like organizational structure, development and governance processes [74]. Based of five maturity levels within these domains the authors show various challenges, benefits, risks being encountered as SOA become more mature. Welke *et al.* propose a similar model and additionally point out that at the final maturity level business processes and services become equivalent [75]. While the assessment of some domains is subjective, both works point out the importance of governance and measurable quality indicators. For this kind of practical evaluation of SOAs Bogner *et al.* provide a measurable quality model that especially assesses the maintainability of *Microservice* architectures based on its structure [76], [77]. La Jung *et al.* measure the quality of an SOA by *Quality-of-Service* levels perceived by users of the SOA [78].

Independently of their architecture process supporting IT systems therefore need to be adaptable to business process models. Based on common patterns Russel defines a conceptual foundation using a catalog of criteria that Process-Aware Information System (PAIS) should meet to ensure they are able to align to changing business needs [79].

1.2.5 Security of Process Supporting IT Systems

Information security is a basic need and common business risk that relates to all kinds of information systems. The international standard ISO

27000 therefore defines goals that should be achieved by organizations and their IT systems [80]. The primary security goals of confidentiality, integrity and availability are subject in research areas regarding the security of process supporting IT systems. Karlapalem *et al.* apply these goals to distributed systems [81]. For their proposed security management system, the authors claim that controlling authorization, authentication and anonymity of all involved parties is fundamental to meet the security goals. In order to pursue securirty goals from the viewpoint of business processes Herrmann *et al.* propose extensions for BPM frameworks [82]. Basin *et al.* analyze how security restrictions effect business processes [83]. The authors further propose an algorithm balancing risk and cost of security policies [84]. Looking at different business risks, security related issues are the top concerns identified by González *et al.* when working with distributed service providers [85].

Within distributed system landscapes it therefore necessary to adhere common workflows and protocols for authentication and authorization. Memon *et al.* propose a centralized service that translates between different authentication flows and maps corresponding identities for other systems [86]. The SOA proposed by Lablans *et al.* provides centralized approach to consistent pseudonomization within a distributed system with high requirements towards anonymity [87]. Especially for large, decentralized system landscapes Grabatin *et al.* raise that data quality becomes an issue and therefore require a level of assurance management [88]. Several research groups analyzed attack schemes on authorization and authentication workflows for distributed systems [89]–[92]. Over the past years Li *et al.* and Yang *et al.* used their results to exploit several flaws within these workflows, allowing them to break into several applications at a time using the same vulnerabilities [93], [94].

Increased security often implies a reduction of user experience. Additional to the technical security of information systems therefore the users' perceptions on security is also addressed by research groups. Feth therefore proposes *Security Awareness*, *Trust* and *Perceived Security* as user centric security goals that should additionally be considered [95]. Looking at common misconceptions, Hof defines practical guidelines for usable and secure systems and uses them to evaluate systems [96].

Usability of security mechanisms becomes even more important as users interact with them using a multitude of devices as pointed out by Montero *et al.* [97] The approach of las Cuevas *et al.* uses machine learning techniques to create user profiles and detect changes in user behavior and potential security breaches [98].

1.3 Research Outline

Using the previously discussed motivation and research results as a basis, this work discusses the fundamentals to prepare the existing decentralized IT system landscape to meet the rising demands of students and researchers and to meet the requirements towards customizable process support. Apart from providing the theoretical basis for assessing and building this system landscape it is the goal to practically prove its functionality across several real world case studies supporting processes in the areas of eLearning and eScience.

1.3.1 Research Questions

To better cope with the latest developments and fast changing requirements, increased mobility and higher degree of customization SOAs are widely used. However due to the broad fields of application of SOAs during the past years a multitude of incompatible standards, formats and architectures have been developed and are now part of virtually any application framework. While choosing the right SOAs is a critical step for the short term success it is most important to build a comprehensive and consistent model of the data exchanged between the systems. To generate this common model several processes at the university need to be analyzed in how they use the available data.

Research Question 1: How can the architecture provide a comprehensive, process independent and reusable model to access existing data stored in heterogeneous distributed systems of the university?

In any software system if it is a SOA or a more traditional monolithic software a central asset is reliability. While in a monolithic environment reliability may be controlled centrally, in a distributed SOA other measures have to be provided to assert this kind of service quality. Naturally these should base on the three key concepts from information security, confidentiality, integrity and availability. As the SOA will participate in a multitude of processes these need to be endorsed already by its design. Further security and reliability needs like authenticity, non-repudiation, accountability or anonymity should furthermore be attainable by an implementation the architecture.

Research Question 2: How can the architecture provide means to increase reliability of these distributed services for existing and new processes?

According to the need for reliability and security of the operators and users of the service, the protection of personal data needs to be enforced. By extending processes and sharing data across their boundaries, data will eventually be automatically accumulated to data sets and may become available to a broader audience. Especially for personal data the general right of personality needs to be protected. The resulting infrastructure needs to be designed in order to allow applications and processes to support regulations such as European and German data protection laws as well as organizational regulations.

Research Question 3: How can protection of personal or confidential data be supported by the architecture when sharing data across process and system boundaries?

Processes supported by the SOA were defined by stakeholders from several contexts. The architecture has to generalize and bridge between these contexts to enable spanning processes across organizational and technical borders. To compare the contexts of different processes three categories are regarded:

- *Local context* defined by a single or few closely related process, like a local Learning Management System (LMS) instance with local user accounts
- *Cooperative context* defined by processes that partially work across technical boundaries, for example by integrating instances of a library management system into an LMS using the same user accounts.

- *Federative context* defined by processes that work across technical and organizational boundaries, for example by integrating a third party storage provider into multiple LMS.

Research Question 4: How can the architecture generalize to support processes working across systems in distributed local, cooperative and federative service landscapes found in the university sector?

Answering the research questions will lay the foundation towards implementing process oriented services supporting eLearning and eScience processes on top of the existing decentralized IT system landscape. The resulting services and applications should furthermore grant a high degree of customization for students and employees of the university.

1.3.2 Contributions

By answering the research questions this thesis contributes to the areas of decentralized and service oriented system architectures. This work focuses on asserting security and integration of legacy systems within these systems. The architecture constructed and evaluated in this works lays the theoretical basis for cooperative and federative distributed services sharing users' identities across process, system and organizational boundaries. The proposed and evaluated services allow managing of the flow of information and authorizations of systems within decentralized service infrastructures and especially consider user perceived security.

Additional to the proposed architecture, a framework to assess the quality of a process supporting SOA was developed. This framework brings together different evaluation criteria from BPM, SoC and IT Security and requires the compatibility with commodity Internet standards. This framework should form a basis for the design of upcoming IT systems supporting processes in distributed environments. It further provides guidelines on how to establish necessary technical and organizational structures to successfully implement and operate this kind of system.s

In conjunction with this framework a model that can be used to lift a legacy monolithic system landscape to allow microservice like interaction, replacement and independence of systems. This is especially important since IT systems are subject to constant change. This model paves the way to gradually migrate from monolithic systems to SOA and then allows the migration of single responsibilities to other systems. By using widely available Internet technologies, it allows opening legacy systems towards use cases that were previously unsupported by systems themselves.

Apart from theoretical basis, the implementation of the system at RWTH Aachen University allows students and researchers provides a practical implementation of services that support university wide processes. The service infrastructure can further be used to customize these processes to integrate them into existing local or decentralized process supporting systems.

1.3.3 Outline of the Thesis

The following Chapter 2 presents a requirement analysis for a distributed service architecture to support processes using IT systems at RWTH Aachen University that bases on several case studies. These requirements are used to build a generalized architecture and derives several dimensions of evaluation criteria that can be used to assess how a distributed system can support decentralized processes. Based on these dimensions an n-tier architecture is proposed and evaluated, that should be put into practice at the university.

Chapter 3 will then introduce a distributed organizational model and key technologies used for an implementation of the architecture. Based on the case studies and derived requirements the chapter will commence to describe the implementation of the system that scales across local, cooperative and federative contexts. The resulting SOA will present an exemplary set of interfaces that can in turn be used to implement the various case studies.

After the intoduction of the theoretical framework and its implementation Chapter 4 will discuss the previously introduces case studies featuring processes from eLeaning, eScience and organizational support.

The chapter will show how processes are modeled on top of the architecture and how their implmentation is evaluated accoring to qualitative and quantitative usage metrics.

Finally Chapter 5 will conclude the thesis and sums up how the proposed architecture influences the implementation of process supporting applications at the university. The chapter also evaluates the implementation and the case studies according to dimensions of evaluation defined in Chapter 2. Based on this evaluation, the chapter answers concludes answers to the research questions and proposes some related future research areas.

1.3.4 Course of Research

An abstraction layer will be developed that is verified using case studies.

As opposed to the order in this work implementation of case studies and the theoretical foundation for the architecture were developed in parallel and were refined multiple times to gradually derive and answer the research questions. Different phases of the course of research can be roughly aligned to a *PDCA-Cycle* as described by Deming [99]. Using the phases as follows:

Plan

Basing on current results and new research ideas, the *Plan*-Phase is used to analyze and formalized the current state of the research. This generates new approaches and provides theoretical input that can be applied within case studies. Work in this phase formally contributes towards answering the research questions.

Do

The work during the *Do*-Phase focuses on applying insights gained during the research in order to advance the case studies and generating measurable results. Furthermore, this possibly arises issues that cannot be solved using the current architecture.

Check

During the *Check*-Phase presentations and articles are used to review the current state of research and present it to peer researchers. Discussions with peers allow identifications of gaps and future research goals and case studies.

Adjust

The *Adjust*-Phase then is used to integrate the current state of research into the implementation of the reference architecture. Basing on the feedback from use cases and publications, the implementation thus incrementally converges to reflect the current state of the research.

Figure 1.1 shows a selection of milestones from the course of research according to the PDCA-phases. The starting point was the introduction of a university mobile app at RWTH Aachen University: *RWTHApp*. The application required an infrastructure to access data from various systems and additionally itself became a platform supporting several eLearning processes. Various presentations showed positive feedback within the community and motivated formalization of the created infrastructures for eLearning process support. While eLearning processes primarily targeted students of the university, additional processes required supporting various groups of users within the organization and across organization boundaries. This scalability across organizational and process boundaries eventually allowed the architecture to capture other process areas like eScience.

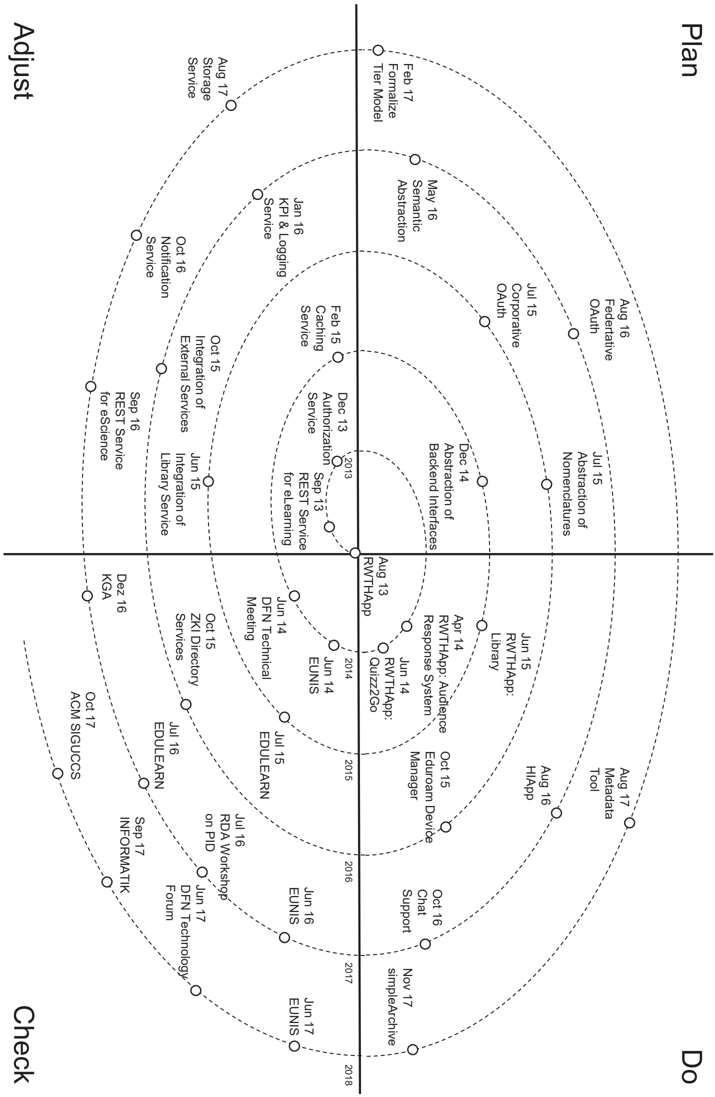

Figure 1.1: Visualization of the course of research