TECHNISCHE UNIVERSITÄT
CHEMNITZ

Department of Computer Science
Distributed and Self-organizing Systems

Master thesis

Enhancing Knowledge Work Through Crowdsourcing in Adaptive Case Management Systems

Peter Thiel

Chemnitz, the 1st December 2014

Examiner: Prof. Dr.-Ing. Martin Gaedke
Supervisor: MSc Sebastian Heil

Thiel, Peter
Enhancing Knowledge Work Through Crowdsourcing in Adaptive Case Management Systems
Master thesis, Department of Computer Science
Technical University Chemnitz, December 2014

Abstract

Increasing the productivity of Knowledge Workers is considered as an important management challenge for an information society. The complexity of Knowledge Work confronts Knowledge Workers with tasks outside their expertise or overextend their physical capabilities. Crowdsourcing solves this problem by allowing Knowledge Workers to employ a global pool of work force for various types of applications.

While utilizing Crowdsourcing, Knowledge Workers have to deal with additional efforts which reduce their overall productivity. A holistic solution, that combines the scopes of Knowledge Work and Crowdsourcing, promises to overcome these problems.

This thesis analyses state of the art approaches within Knowledge Work and Crowdsourcing, and introduces a concept which allows the integration of Crowdsourcing in the process of Knowledge Work. In addition to the theoretical examinations, an implemented prototype demonstrates the applicability of the concept.

Contents

List of Figures

List of Tables

ACM	Adaptive Case Management
ACMS	Adaptive Case Management System
AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
BPM	Business Process Management
CSS	Cascading Style Sheets
HIT	Human Intelligence Task
HTML	HyperText Markup Language
IIS	Internet Information Services
JSON	JavaScript Object Notation
OECD	Organisation for Economic Co-operation and Development
SQL	Structured Query Language

1 Introduction

Since the beginning of the 20th century job profiles underwent a remarkable change from manual labour to mental labour. As economy and technologies developed production processes increased in complexity. Manual labour got replaced by automated machines and highly specialized workers became more important. This change caused a new type of post-modern society which is known as information society.

The information society as popularized by Bell [5] describes a society which generates wealth through creating and sharing knowledge and information rather than producing industrial goods. Bell's thesis is based on observations by Machlup [37] who recognized an increase in employment in knowledge-oriented professions. The change in employment introduced a new type of worker: The Knowledge Worker.

The term Knowledge Work was first defined by Drucker. In differentiation to common known types of workers a Knowledge Worker is described as a worker who uses knowledge rather than muscle power to fulfil his tasks. [20]

Drucker also emphasized that Knowledge Work will become crucial for the wealth and growth of a modern society building a economy mainly based on knowledge. This prediction is backed up by statistics of the OECD [45, 42] which are showing an increasing global employment in knowledge related sectors. Raising the productivity of Knowledge Workers therefore will be one of the biggest challenges of management in the 21th century. [21]

1.1 Motivation

Improving the productivity of Knowledge Workers requires an analysis of the nature of Knowledge Work and the implementation of suitable approaches.

Knowledge Work is a creative and innovative type of work addressing novel and complex work processes which are performed by highly specialized and autonomous working labour. Knowledge Work does not follow a strict routine, thus detailed work tasks are hard to predict. [31, 16, 51]

Because of the uncertainty of Knowledge Work, Knowledge Workers can be confronted with tasks they cannot cope with, e.g. tasks which lie outside of their expertise or overextend their physical capabilities. Therefore they need assistance with the execution or acceleration of specific tasks. For example, the classification of large amounts of content like pictures is time-consuming for single workers and could be improved through the support of other workers.

Outsourcing provides the main source for improving work through additional work force. Outsourcing describes an concept to assign internal processes to external contracted providers. This approach promises to reduce costs by allowing each partner to focus on its core competencies [26]. However, finding suitable work force is not trivial since it includes a iterative process of searching and selecting potential partners.

Crowdsourcing is a more promising approach that allows to overcome problems common to Outsourcing. Crowdsourcing, as defined by Howe and Robinson, is described as „the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call“. Crowdsourcing in practice means that the so called announcer publishes a task on a particular website to an undetermined community, also called the crowd. The crowd selects and solves tasks individually based on their will to compete and contribute. The individual solutions are then delivered back to the announcer. [30]

Crowdsourcing can be used in many ways, which is indicated through the variety of existing applications implementing the concept in different topics, e.g. classifying craters on planetary surfaces [56], deciphering scanned text [25], or discovering new galaxies [3]. While the concept of Crowdsourcing is not new, the use of web-based technologies has made Crowdsourcing practicable for a globalized world. [9]

Knowledge Workers can use Crowdsourcing to efficiently employ a large pool of work force and retrieve suitable solutions for specific problems. Tasks like classifying large amounts of content become simple and no longer time-consuming to single Knowledge Workers.

1.2 Problem statement

As laid out in 1.1 Crowdsourcing promises valuable improvements to Knowledge Work.

While many applications of Knowledge Work and Crowdsourcing are conceivable there is no solution that meets the intersections of Knowledge Work and Crowdsourcing (Figure 1.1).

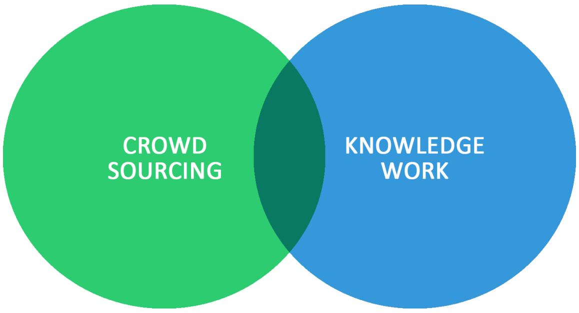

Figure 1.1: A suitable solution will meet the intersection of Knowledge Work and Crowdsourcing.

Knowledge Workers can either use solutions in the scope of Knowledge Work or solutions in the scope of Crowdsourcing. Using solutions of both scopes simultaneously requires Knowledge Workers to manage multiple solutions. This reduces the productivity of Knowledge Workers and conflicts with the overall goal of the thesis to enhance Knowledge Work through Crowdsourcing.

Using solutions in the scope of Crowdsourcing forces Knowledge Worker into foreign domains. To utilize Crowdsourcing they have to gain additional knowledge while trying to understand new concepts and functionalities. This requires efforts and reduces the productivity of Knowledge Workers additionally. Furthermore, problems can occur through improper use of Crowdsourcing [49].

Knowledge Workers need a solution that combines the scope of Knowledge Work and the scope of Crowdsourcing in a holistic way. This approach promises to reduce usage complexity, enable Knowledge Workers to stay in their domain of expertise, and finally raise the productivity of Knowledge Workers.

These considerations show that a precise statement about how Knowledge Work can be enhanced through Crowdsourcing is not trivial and requires further research.

1.3 Motivation Scenario

This section describes a scenario to illustrate the need for a solution that enhances Knowledge Work through Crowdsourcing. The scenario deals with a start-up which has invented a new product.

Inventing new products implies to find appealing visual and verbal characteristics. This example investigates how a start-up finds the perfect package design and name for their newest product.

John is an ambitious chemist and has invented his own hair shampoo which he thinks will be revolutionary. John founded his own start-up trying to organize production and marketing by himself. To organize his start-up John is using a software tool which helps him planing the production processes.

Besides several other tasks John has to find proper packaging for his new product. While John is an expert in chemistry, he has no experience and knowledge in professional design. So John decides to outsource these tasks to qualified personnel.

To find the right work force John will announce tasks to agencies, receive several applications, select workers, negotiate payment and sign contracts. This process will take time and add additional costs for paying the agencies. In addition, John will be bound to the signed contracts and therefore bound to the delivered solutions. Since John wants to find the best suitable designs for his products this approach is not effective.

John decides to use Crowdsourcing to announce tasks to a wider audience and retrieve various solutions. That is why he reads about the basic concept and applications of Crowdsourcing. Because John is not used to Crowdsourcing, he searches and selects platforms at random.

After a while John finds a platform which offers Crowdsourcing for product designs. After creating an account he has to understand the functionalities of the service implementation. Finally, he manages to create an announcement for his product, providing a specific description and several parameters, e.g. a amount and quality of expected results. After having received various suggestions John selects and rewards the best suiting product designs.

Now John remembers that he also needs a proper product name. Because John is convinced of Crowdsourcing he wants to use this method again. In search of a suitable platform, he registers a new account, tries to understand the specific functionalities and finally creates an appropriate announcement to receive several results.

The scenario proofs that Crowdsourcing can enhance Knowledge Work. However, John is forced to leave his domain of expertise which means undesired efforts. Furthermore John now uses different approaches that enable him to manage his startup and to use Crowdsourcing. This adds complexity to his work and decreases his productivity additionally. If John had access to a solution that combines the different approaches the scenario would look different.

1.4 Objective

This thesis focusses on how Knowledge Work can be enhanced through Crowdsourcing. This contains the analysis of state of the art approaches in the areas of Knowledge Work and Crowdsourcing as well as a concept, implementation and evaluation of an appropriate approach or respectively a combination of approaches for solving the outlined problem.

To help Knowledge Workers using Crowdsourcing it is desirable to create or extend a solution which integrates Crowdsourcing platforms into the process of Knowledge Work. The integration has to be modular and capable of embedding existing or novel Crowdsourcing platforms. The demonstration of the solution based on real existing scenarios is also part of the objective.

1.5 Organization of the Thesis

Chapter 1 introduced the thesis „Enhancing Knowledge Work through Crowdsourcing“ by emphasizing motivation and problems, described in a realistic motivation scenario illustrating the practical relevance.

Chapter 2 defines requirements for a suitable solution based on the problem statement in chapter 1. Further the chapter introduces state of the art approaches and evaluates the suitability for a suitable solution.

Chapter 3 describes a concept that will meet the defined requirements by reflecting the results of chapter 2.3.

The implemented prototype of the concept is described in chapter 4 followed by a summarizing evaluation in chapter 5.

The thesis finally is completed by a conclusion and a perspective on potential future research objects in chapter 6.

2 State of the Art

This chapter defines requirements based on the outlined problems described in chapter 1. Further, this chapter introduces known approaches and evaluates their contribution to a suitable solution based on these defined requirements.

2.1 Requirements

This section describes requirements for a suitable solution that will enhance Knowledge Work through Crowdsourcing. The following requirements are introduced:

1. Knowledge Work
2. Crowdsourcing
3. Multiplatform
4. Quality
5. Applicability

Each requirement comes with a separate scale to evaluate its implementation by a certain approach. The scale is visualized through an ascending division in the following ratings:

- ○
- ◐
- ●

Implementations which are evaluated with the rating ● fulfil a requirement completely. The rating ◐ illustrates that a approach fulfils a requirement to a limited degree. If an approach is rated with ○ the corresponding requirement is not fulfilled.

2.1.1 Knowledge Work

Knowledge Workers need the empowerment to organize processes adaptively[1]. Drucker highlighted that Knowledge Work is a process of innovation and therefore creative work. Swenson confirmed this definition and pointed out that Knowledge Work follows no strict routine. This implies that detailed tasks of Knowledge Work are hard to predict. Further, Swenson emphasized that „adaptive processes only really become processes when the components are assembled at run time by the user“. Thus, managing unpredictable tasks means to provide concepts which enable adaptive process planing while executing the process. [20, 51]

Furthermore, Knowledge Workers need support in knowledge management. Drucker described a Knowledge Worker as someone „knowing more about their job than anyone else in the organization“ [19]. This is confirmed by Davenport [16], emphasizing that knowledge is the highest value of a Knowledge Worker. To preserve the growing knowledge of a Knowledge Worker it is important to provide suitable management support.

Knowledge Workers need support in communication and collaboration to exchange knowledge and information. Davenport [16] describes Knowledge Work as a process of collecting, processing, generating and distributing knowledge to support problem solving and decision making in a complex working environment. Mundbrot [40] confirmed this, emphasizing the complexity and novelty of Knowledge Work processes that require interaction with others.

In summary, the requirement „Knowledge Work“ has to meet the following standards:

1. Enable adaptive process organization
2. Support knowledge management
3. Support collaboration

Table 2.1 describes the evaluation scale for the requirement Knowledge Work.

[1] Adaptive: „Internal changes to an entity caused by outside conditions that become permanent and make the entity more fitting to those new conditions. It does imply that those changes are performed by means of the entity itself and not by some external force.“ [46]

Rating	Description
●	The approach is optimal, implementing all mentioned standards
◐	The approach is applicable, supporting knowledge management as well as communcation and collaboration
○	The approach is not applicable, implementing none of the mentioned standards

Table 2.1: Evaluation scale for the requirement Knowledge Work

2.1.2 Crowdsourcing

As laid out in 1.1, Knowledge Workers can be confronted with tasks they cannot cope with. The unpredictability of Knowledge Work results in complex tasks that will exceed the capabilities of a single Knowledge Worker. Crowdsourcing provides a useful concept for outsourcing tasks efficiently by using web-based technologies. Crowdsourcing will enable Knowledge Workers to find proper work force and retrieve suitable solutions.

In addition to Howe [30] a more integrated and consistent description of Crowdsourcing is given by Estelles-Arolas and Gonzalez-Ladron-de-Guevara in [23, p. 197]:

> „Crowdsourcing is a type of participative online activity in which an individual, an institution, a non-profit organization, or company proposes to a group of individuals [i.e. „crowd“] of varying knowledge, heterogeneity, and number, via a flexible open call, the voluntary undertaking of a task. The undertaking of the task, of variable complexity and modularity, and in which the crowd should participate bringing their work, money, knowledge and/or experience, always entails mutual benefit. The user will receive the satisfaction of a given type of need, be it economic, social recognition, self-esteem, or the development of individual skills, while the requester will obtain and utilize to their advantage what the user has brought to the venture, whose form will depend on the type of activity undertaken.“

In summary, Crowdsourcing is defined by the following characteristics as proposed in [23]:

1. There is a clearly defined crowd
2. There exists a task with a clear goal
3. The recompense received by the crowd is clear
4. The compensation to be received by the crowdsourcer is clearly defined
5. It is an online assigned process of participative type
6. It uses an open call of variable extent
7. It uses the internet

Table 2.2 describes the evaluation scale for the requirement „Crowdsourcing“.

Rating	Description
●	The approach is applicable, enabling Crowdsourcing as defined
○	The approach is not applicable, not enabling Crowdsourcing as defined

Table 2.2: Evaluation scale for the requirement Crowdsourcing

2.1.3 Multiplatform

A suitable solution enables Knowledge Workers to use of various and arbitrary Crowdsourcing platforms, i.e. Multiplatform Crowdsourcing. Several different platforms implement Crowdsourcing for a specific application [30]. Thus, domain specific work force is spread across different providers. A solution that enables Multiplatform Crowdsourcing allows Knowledge Workers to employ a far bigger labour pool then a single Crowdsourcing platform would. This raises the possibility to receive more suitable solutions and find suitable assistance for a specific task.

Further, a suitable solution enables not only the use of existing but also of future Crowdsourcing platforms. The possibilities of Crowdsourcing are not yet fully explored [30]. New platforms will provide novel applications of Crowdsourcing or improve functionalities of existing ones, thus enabling the use of future Crowdsourcing platforms is important.

Crowdsourcing leverages the phenomenon of „Wisdom of the Crowds“ where a large group of individuals makes better, more informed decisions and choices than individuals or a small group of experts. Application of „Wisdom of the Crowd“ has the following requirements: diversity, independence, decentralization, and aggregation of opinions [50]. By enabling various Crowdsourcing platforms the conditions under which these requirements are met can be improved.

In summary, the requirement „Multiplatform“ has to meet the following standards:

1. Enable use of various Crowdsourcing platforms
2. Enable use of arbitrary Crowdsourcing platforms
3. Enable use of future Crowdsourcing platforms

Table 2.3 describes the evaluation scale for the requirement Multiplatform.

Rating	**Description**
●	The approach is optimal, enabling the use of various and arbitrary, existing and future Crowdsourcing platforms
◐	The approach is applicable, enabling the use of various, existing and future Crowdsourcing platforms
○	The approach is not applicable, not enabling the use of various Crowdsourcing platforms

Table 2.3: Evaluation scale for the requirement Multiplatform

2.1.4 Quality

Crowdsourcing also includes the quality management of solutions contributed by the crowd.

Employment of a large group of individuals allows the retrieval of various solutions. However, not all contributions are applicable. It is herefore necessary to evaluate the quality of individual contributions.

Because of the potentially large number of retrieved solutions, the evaluation is a time-consuming process for Knowledge Workers. This conflicts with the overall goal of raising productivity. Thus, the results have to be evaluated through appropriate mechanisms that allow Knowledge Workers to find applicable results. A suitable solution provides mechanisms that enable the evaluation of the quality of the results delivered by the crowd.

Table 2.4 describes the evaluation scale for the requirement „Quality“.

Rating	Description
●	The approach is optimal, providing mechanisms to evaluate the quality of results delivered by the crowd
○	The approach is not applicable, not providing mechanisms to evaluate the quality of results delivered by the crowd

Table 2.4: Evaluation scale for the requirement Quality

2.1.5 Applicability

Section 2.1.1 defined requirements for a solution based on the characteristics of Knowledge Work. This section describes the requirement that a solution has to be applicable for Knowledge Workers.

As stated in 2.1.1 Knowledge Workers are experts in a specific domain. Activities in foreign domains require additional efforts to become familiar with new knowledge and achieve experience with new concepts and features. In addition, the application of unfamiliar concepts bears risks of misuse [49]. A suitable solution allows Knowledge Workers to stay in their domain of expertise. This will avoid efforts and risks, resulting into higher productivity of Knowledge Workers.

Employing different approaches or applications raises usage complexity. These context switches mean additional efforts for Knowledge Workers, lowering their productivity. Therefore, unnecessary context switches have to be avoided to lower usage complexity and improve Knowledge Work.

In summary, the requirement „Applicability“ has to meet the following standards:

- Allow Knowledge Workers to stay in their domain of expertise
- Avoid unnecessary context switches

Table 2.5 describes the evaluation scale for the requirement Applicability.

Rating	Description
●	The approach is optimal, requiring no additional knowledge, avoiding unnecessary context switches
◐	The approach is applicable, requiring no additional knowledge or avoiding unnecessary context switches
○	The approach is not applicable, requiring additional knowledge and context switches

Table 2.5: Evaluation scale for the requirement Applicability

2.2 Approaches

The following section provides summaries of approaches which address the previously defined requirements in chapter 2.1 and promise a valuable contribution to an applicable solution. The examined approaches are grouped into:

- Approaches that support Knowledge Work, i.e. Knowledge Work Approaches
- Approaches that enable Crowdsourcing, i.e. Crowdsourcing Approaches
- Approaches that support Knowledge Work and enable Crowdsourcing, i.e. Hybrid approaches

2.2.1 Knowledge Work Approaches

The group of Knowledge Work approaches describes those approaches that are currently in use or promise a suitable utilization to support Knowledge Work. Following the definitions of Knowledge Work introduced in chapter 2.1.1 and as stated in [51] the focus of Knowledge Work approaches are processes and data. Thus, the examined representatives are divided into the group of process-centric approaches and the group of data-centric approaches.

Process-centric Approaches

Process-centric approaches support Knowledge Work through a focus on pre-defined processes. The process is the central theme used to manage Knowledge Work. Examined representatives of this group are:

- Enterprise Resource Planing
- Business Process Management (BPM)
- Social Software

Enterprise Resource Planing is a combination of various software applications which address the organization and management of processes in different business fields, e.g. resources, customers, markets or production. Enterprise Resource Planing focuses on support and optimization of processes by providing internal and external information that are process related. The information is stored and managed in a database management system and shared between the different components, e.g. Product Lifecycle Management, Supply Chain Management or Customer Relationship Management. [44]

BPM is a management principle that focuses on process optimization. As emphasized by [33], BPM can be seen in line with Scientific Management. The structure of a process is analysed to reproduce detailed steps, modelling the work flow of a process. This enables the identification of improvements by measuring and monitoring the ongoing performance of a process. As stated in [1] BPM involves „methods, techniques, and software to design, enact, control, and analyze processes involving humans, organizations, applications, documents and other sources of information".

In [33] the life-cycle of a process in BPM is divided into the steps Analysis, Design, Implementation, Management and Improvement. Processes are analysed in order to design work plans which will then be used to implement the execution of a process in a pre-defined way. Improvements or changes to a process initialize new iterations of the process life-cycle. Modelling processes in BPM is standardized through Business Process Modelling Notation which provides graphical elements to simplify the creation and understanding of workflows and processes [27].

Social Business Process Management enhances BPM by integrating Social software (i.e. Web 2.0) into processes. Social Software incorporates web-based applications that enable the participation of users in the process of content creation. Those applications support interaction and collaboration between users, improving the communication and coordination in communities, e.g. chats or social network platforms. Collaboration tools are used to enable coworking on tasks and to share information among multiple participants. They allow the execution of complex work processes in a distributed work environment. Web-based technologies enable collaboration independent of physical presence via various implementations, e.g. email, wikis, blogs or project management platforms [43].

Data-centric Approaches

Data-centric approaches support Knowledge Work in the context of given data. In contrast to process-centric approaches data is central theme to define the management of Knowledge Work. The group of data-centric approaches is represented through:

- Enterprise Content Management
- Case Management
- Productive Case Management
- Adaptive Case Management (ACM)

Enterprise Content Management is a combination of applications, methods and functionalities that are used to capture, manage, store, preserve, and deliver structured and unstructured information, e.g. document management, workflow management or archive management. Enterprise Content Management controls and supports processes by providing relevant content to authorized users. It enables the unified access to various types of content, avoiding redundancy of data and documents. [47, 2]

Case Management is a concept of providing data and information for the processing of specific cases, e.g. the examination of a patient in a hospital. Content (e.g. documents, etc.) is related to a case which consists of specific tasks to achieve a specific goal. Cases are defined by users and belong to one or more individuals (i.e. stakeholders) and can be situated in different states. Information and knowledge is stored and modified along the execution of a case. Thus the focus of Case Management is to document content, support collaboration and enable interaction. [39, 2]

Production Case Management is an approach to produce a specific case management application that implements certain pre-defined operations which are used for adaptive case definitions. Production Case Management is designed to handle situations with many variations between individual cases. The approach allows the involvement of users actively in deciding the course of events for a case. The operations' power is specified in advance. [52]

ACM is enhances Case Management by enabling adaptive creation and modification of tasks and goals for a case. Goals define overall achievements for a case and consist of different tasks that define the processing of a case. Case templates allow to reuse process patterns that contain recurring tasks. These templates can be created and modified by users at the runtime of a case, enabling the possibility to improve processes fast and effective. Users can define rules that allow process automatization and state tracking. Thus, the adaptive way of handling processes and data is characteristic for ACM. [51, 2]

2.2.2 Crowdsourcing Approaches

Crowdsourcing is implemented through various platforms[2]. This section introduces groups of approaches that implement the concept of Crowdsourcing as defined in chapter 2.1.2. The approaches are divided into dedicated Crowdsourcing approaches and intermediary Crowdsourcing approaches.

Dedicated Approaches

Dedicated Crowdsourcing approaches implement the concept of Crowdsourcing as defined in chapter 2.1.1 in a specific field of application[3]. The group of dedicated Crowdsourcing approaches is represented by:

- Amazon Mechanical Turk
- CrowdFlower
- Crowdspring
- InnoCentive

Amazon Mechanical Turk is described as an „artificial intelligence“ service that performs human intelligence tasks (i.e. „HITs“). HITs are tasks that cannot be easily processed by machines, but by humans. Such a task could be for example „identify the color of a car in the presented picture“. Tasks are solved by applying human intelligence and the skills of workers. Assigned workers are rewarded after executing a HIT successfully. Amazon Mechanical Turk represents a virtual work environment where human beings interact with each other and perform economic activities for their own advantage. [4]

CrowdFlower is similar to Amazon Mechanical Turk, allowing users to access an online work force to execute problem specific tasks. CrowdFlower cooperates with partners in so called „Channels“ building a network of distributed labour pools. CrowdFlower allows decisions on how the retrieved results, called „Judgements“, have to

[2] An overview of existing platforms that implement Crowdsourcing is retrievable from `http://www.crowdsourcing.org/directory` [14]

[3] [17] divides Crowdsourcing into 27 categories of applications

be collected and aggregated. Test questions are used to measure the accuracy of contributors. Skill levels of workers help to find appropriate work force. [12]

Crowdspring allows users to employ a global workforce for creative tasks, i.e. logo design, web design, company name, product name, packaging design. A requester can announce a project in a certain category to a creative community, promising a specific recompense. The provided results are evaluated through a contest, allowing participants to rate each others ideas. The requester then decides who contributed the best solutions. [15]

InnoCentive provides a Crowdsourcing platform to enable scientific problem solving in the fields of engineering, computer science, math, chemistry, life sciences and physical sciences. Seekers can announce so called „Challenges“ to solvers. Challenges are announced along with a financial award that is given to the solver who provides the most suitable solution. [32]

Intermediary Approaches

The group of intermediary approaches leverages dedicated Crowdsourcing approaches in context of specific problems. An intermediary layer connects internal system functionalities with external Crowdsourcing platforms through an API. Representatives of intermediary Crowdsourcing approaches are:

- CrowdDB
- CrowdQ
- CrowdForge
- CrowdSearcher

CrowdDB supports complex queries in the areas of databases and information retrieval by making use of Crowdsourcing. The approach integrates human input in traditional database systems. An extension of SQL (i.e. CrowdSQL) is used to model, manipulate, and query the provided data. The interaction with the crowd (e.g. Amazon Mechanical Turk) is provided through automatic user interface generation, based on available schemas and queries. [24]

CrowdQ uses Crowdsourcing to gain knowledge of query structure and entity relationships. Log mining, Natural Language Processing, and Crowdsourcing are combined to generate query templates that can be used to answer classes of questions. [18]

Crowdforge uses Crowdsourcing platforms like Amazon Mechanical Turk to generate task workflows. Therefore, the crowd is used to break a complex task recursively into simple tasks that can be executed by single workers. The particular results of multiple simple tasks are combined afterwards to a holistic result. This approach also allows the design and execution of work flows in a collaborative way. [35]

CrowdSearcher allows to use Crowdsourcing for search-related tasks through involvement of social networks, e.g. Facebook, LinkedIn and Twitter. It describes a search paradigm that embodies crowds as sources for the information seeking process. CrowdSearcher is implemented through a platform where users define customized tasks and announce those tasks to a selected network. [7]

In addition to the original idea of CrowdSearcher, the prototype[4] also enables the user to announce tasks to Amazon Mechanical Turk or a custom Crowdsourcing platform called TaskExecutionFramework.

2.2.3 Hybrid Approaches

The group of hybrid approaches includes approaches that introduce concepts to support Knowledge Work and enable the use of Crowdsourcing. Representatives of hybrid approaches are:

- Enterprise Crowdsourcing
- Business Process Crowdsourcing
- Human-provided Services

Enterprise Crowdsourcing enables Crowdsourcing in enterprise environments by integrating the crowd into processes of a company. The concept distinguishes Crowdsourcing into Intra-corporate Crowdsourcing and Corporate Crowdsourcing. In Intra-corporate Crowdsourcing the crowd is defined by individuals inside of an enterprise. Corporate Crowdsourcing means to employ a public crowd outside of an enterprise. [29]

Business Process Crowdsourcing applies Enterprise Crowdsourcing in context of BPM. Specific functionalities of Crowdsourcing platforms like Amazon Mechanical Turk are used to enable Crowdsourcing for particular processes in a specific enterprise domain. [36]

Human-provided Services describes a concept of a two-way participation of humans in service-oriented systems. User defined interaction models allow participants (i.e. software or humans) to provide and request services [48]. Human-provided Services

[4] An executable prototype of CrowdSearcher can be found on `http://demo.search-computing.com/cs-demo`

implements the concept of human computation. Human computation can bee seen as similar to Crowdsourcing, since both highlight the involvement of humans in the execution of a task. In contrast to Crowdsourcing, human computation focuses on replacing computers with humans and not on replacing traditional workers with an undefined large group of people. [29]

2.3 Evaluation

This section provides an evaluation of the approaches introduced in chapter 2.2 based on requirements for improvement of Knowledge Work through Crowdsourcing as defined in chapter 2.1. An overview and conclusion of the results of the evaluation are given in chapter 2.4.

2.3.1 Knowledge Work

Process-centric Knowledge Work approaches, like BPM, are characterized by time-consuming predictive process analysis and design of work flows. Adaptations in processes require the redefining of the designed work flows. The complexity of Knowledge Work requires an adaptive organization of processes as stated in chapter 2.1. Process-centric approaches are counterproductive to this condition. BPM can be improved by Social Software (i.e. Social Business Process Management), supporting collaboration and management of knowledge. Because process-centric approaches do not address all stated standards, this group is evaluated with the rating ◐.

Data-centric Knowledge Work approaches provide context specific support through appropriate data. Case Management enables the collaboration and management of knowledge in the context of cases. Productive Case Management improves the concept by providing predefined activities, enabling Knowledge Workers to define cases autonomously. In ACM cases are defined and configured dynamically at runtime. Users can change cases on the fly responding to current conditions. Thus Knowledge Workers are not only involved in the execution of a case, but also help to improve them. Templates allow to reuse occurring process patterns. ACM supports Knowledge Workers in collaboration and management of knowledge and information and enables adaptive process organization. Thus the group of data-centric approaches implement all required standards, resulting in the rating ●.

Dedicated Crowdsourcing approaches and intermediary Crowdsourcing approaches provide support in collaboration with bigger work forces. However they provide no functionality for adaptive process organisation or management of knowledge and information. Because dedicated Crowdsourcing approaches and intermediary Crowdsourcing approaches wont be applicable for the requirement Support Knowledge Work both groups are rated with ○.

Hybrid approaches support Knowledge Workers in collaboration and knowledge management. However, Business Process Crowdsourcing leverages the concept of BPM. As mentioned above, BPM cannot provide adaptive process organisation to Knowledge Workers. Also Human-provided Services do not address this condition. Therefore the group of hybrid approaches is rated wit ◐.

2.3.2 Enable Crowdsourcing

Process-centric Knowledge Work approaches focus on supporting processes. Social Software supports Knowledge Workers in aspects of collaboration and communication. Crowdsourcing applications can be implemented through Social Software. However process-centric approaches are not designed to enable the use of Crowdsourcing in general. The group of process-centric approaches is rated ○.

Also none of the examined representatives of data-centric Knowledge Work approaches enables Crowdsourcing as defined in chapter 2.1.2, thus this group is rated ○.

Dedicated Crowdsourcing approaches enable Crowdsourcing as defined in requirement 2.1.2, e.g. CrowdFlower or Amazon Mechanical Turk. Thus, the group of dedicated Crowdsourcing approaches implement all mentioned functionalities, resulting in the rating ●.

Intermediary Crowdsourcing approaches enable the use of Crowdsourcing platforms through an API. However, none of the examined representative enables Crowdsourcing independently of dedicated Crowdsourcing approaches. Thus, the group of intermediary Crowdsourcing approaches is rated ○.

The examined representatives of hybrid approaches enable the use of Crowdsourcing by internal or external crowds, or the use of Crowdsourcing related functionalities. Like in the group of intermediary Crowdsourcing approaches none of the mentioned representatives enable Crowdsourcing independently of dedicated Crowdsourcing approaches. Therefore, the group of Hybrid approaches is also rated ○.

2.3.3 Multiplatform

Knowledge Work approaches won't enable the use of Crowdsourcing, as stated in 2.3.2. Thus both groups, process-centric and data-centric approaches, can not fulfil the requirement Multiplatform. Both groups become rated ○.

All mentioned representatives of dedicated Crowdsourcing approaches enable the use of Crowdsourcing through a single platform. CrowdFlower extends the range of possible work force through Channels. However, CrowdFlower cannot allow the use of various and arbitrary Crowdsourcing platforms. Thus, the group of dedicated Crowdsourcing approaches is rated ○.

Intermediary Crowdsourcing approaches leverage dedicated Crowdsourcing approaches. Most of the mentioned representatives focus on the use of single platforms such as Amazon Mechanical Turk or Facebook. CrowdSearcher provides an approach that allows the use of various and arbitrary, future and existing Crowdsourcing platforms of any kind. The group of intermediary Crowdsourcing approaches is rated ●.

Human-provided Services enable Knowledge Workers to use a variable network of work force. Business Process Crowdsourcing allows the use of Crowdsourcing in various domains of an enterprise. However, none of the examined representatives introduce an approach to fulfil the requirement Multiplatform as defined in 2.1.3 to a satisfying degree. Thus, the group of hybrid approaches is not applicable, resulting in the rating ○.

2.3.4 Quality

As stated in 2.1.4 retrieving results from the crowd requires mechanisms for quality management.

None of the mentioned representatives of Knowledge Work approaches can fulfil this requirement. Thus both groups are rated ○. This evaluation applies also to the group of hybrid approaches since none of the mentioned representatives allow quality management for solutions delivered by a crowd.

In all of the examined dedicated Crowdsourcing approaches quality management is integral part to ensure customer satisfaction. Even though not all existing Crowdsourcing platforms offer appropriate mechanisms, the requirement is fulfilled by some representatives like CrowdFlower or Amazon Mechanical Turk. Thus the group of dedicated Crowdsourcing approaches is rated ●.

After examining representatives of intermediary Crowdsourcing approaches only CrowdSearcher provides mechanisms for quality management. The platform implements reactive Crowdsourcing [8] to dynamically adapt the systems in correspondence to the quality of results and performance of contributors. Thus, the group of intermediary Crowdsourcing approaches is rated ●.

2.3.5 Applicability

Process-centric Knowledge Work approaches require pre-defined process modelling through standardized notations like Business Process Modelling Notation. The application of Business Process Modelling Notation requires the acquisition and understanding of the correct syntax of the notation which is an additional effort for Knowledge Workers. This conflicts with the overall goal to raise productivity of Knowledge Workers and forces them to leave their domain and switch context between different applications. Swenson [53] confirmed this and emphasized that Business Process Modelling Notation is not suitable for Knowledge Workers. Thus, this group is rated ○.

Data-centric Knowledge Work approaches do not require pre-defined processes. ACM allows Knowledge Workers to work autonomously without the requirement to learn additional notations. Thus, data-centric approaches allow Knowledge Work to stay in their domain. However data-centric approaches still require Knowledge Workers to switch between different applications. Therefore, this group is applicable in context to the requirement Applicability, resulting in the rating ◐.

Dedicated Crowdsourcing approaches and intermediary Crowdsourcing approaches require Knowledge Workers to gain additional knowledge in the field of Crowdsourcing and switch between different Crowdsourcing platforms. Both groups of Crowdsourcing approaches are not suitable for Knowledge Workers, rated ○.

Hybrid approaches combine Knowledge Work and Crowdsourcing. Business Process Crowdsourcing provides the option to integrate various Crowdsourcing platforms into work processes. However, process-centric approaches are not suitable for Knowledge Workers as mentioned before. Also Human-provided Services require additional knowledge in usage. In summary this group is not suitable for Knowledge Workers, resulting in the rating ○.

2.4 Conclusions

An overview of the previous evaluations of the approaches is given in table 2.6. The evaluation scale is described in chapter 2.1.

The evaluated groups are abbreviated as follows:

- KW-PC represents the group of process-centric Knowledge Work approaches
- KW-DC represents the group of data-centric Knowledge Work approaches
- CS-DE represents the group of dedicated Crowdsourcing approaches
- CS-IN represents the group of intermediary Crowdsourcing approaches
- HYB represents the group of hybrid approaches

The mentioned requirements from chapter 2.1 are abbreviated as follows:

- Req-KW represents the requirement Support Knowledge Work
- Req-CS represents the requirement Crowdsourcing
- Req-MP represents the requirement Multiplatform
- Req-QU represents the requirement Quality
- Req-AP represents the requirement Applicability

As indicated by table 2.6, each requirement is particularly well addressed by at least one specific group of approaches.

Data-centric Knowledge Work approaches provide suitable solutions to implement the requirement Knowledge Work. Process-centric Knowledge Work approaches, data-centric Knowledge Work approaches and hybrid approaches enable Knowledge Workers to manage knowledge and gain support in collaboration. However, only data-centric Knowledge Work approaches enable Knowledge Workers to organize processes in an adaptive way.

Dedicated Crowdsourcing approaches provide suitable solutions for the requirement Crowdsourcing as defined in 2.1.2. Intermediary Crowdsourcing approaches and hybrid approaches also allow the use of Crowdsourcing, but require dedicated Crowdsourcing approaches, what means that they cannot meet this requirement.

Approaches	Req-KW	Req-CS	Req-MP	Req-QU	Req-AP
KW-PC	◐	○	○	○	○
KW-DC	●	○	○	○	●
CS-DE	○	●	○	●	○
CS-IN	○	○	●	●	○
HYB	◐	○	○	○	○

Table 2.6: Evaluation overview

Intermediary Crowdsourcing approaches are suitable to enable the use of various and arbitrary, existing and future Crowdsourcing platforms, fulfilling the requirement Multiplatform to a satisfying degree.

The requirement Quality is implemented by dedicated Crowdsourcing approaches and intermediary Crowdsourcing approaches. However, not all representatives of both groups provide appropriate mechanisms.

Only data-centric approaches fulfil the requirement Applicability to a satisfying degree. This group enables Knowledge Worker to stay in their knowledge domain and avoid context switches between different applications.

Illustrating a combination of Knowledge Work and Crowdsourcing, hybrid approaches should actually provide appropriate concepts for a suitable solution. However, none of the mentioned representatives is able to fulfil the stated requirements to a satisfying degree. Thus, a combination with other approaches is necessary, but neither the concept of Business Process Crowdsourcing nor Human-provided Services promise a solution.

Data-centric Knowledge Work approaches provide a suitable initial point for a concept, addressing the needs of Knowledge Workers in an optimal way. However, they enable no Crowdsourcing, provide no functionality to enable the use of various Crowdsourcing platforms and implement no mechanisms for quality management. On the other hand, dedicated Crowdsourcing approaches and intermediary Crowdsourcing approaches provide applicable solutions to meet these specific requirements, but cannot fulfil the requirements Knowledge Work and Applicability. Nevertheless, each approach promises a valuable complementary asset for a suitable solution. Thus, a combination of these approaches seems to be reasonable.

In conclusion, the combination of data-centric Knowledge Work approaches, intermediary Crowdsourcing approaches and dedicated Crowdsourcing approaches illustrates the best applicability for a suitable solution. The following chapter leverages this finding, introducing a suitable concept that implements the combination of the selected approaches, fulfilling the defined requirements from chapter 2.1 in a holistic way.

3 Concept

This chapter introduces a concept that provides a suitable solution for the overall goal to enhance Knowledge Work through Crowdsourcing, by implementation of the requirements from chapter 2.1.

The evaluations in 2.4 have shown that the combination of data-centric Knowledge Work approaches, intermediary Crowdsourcing approaches and dedicated Crowdsourcing approaches provides a solution that can meet all requirements to a satisfying degree.

The preliminary considerations of the concept are described in 3.1, followed the introduction of the architecture of the concept in 3.2. Chapter 3.3 describes the different components in detail. Processes that combine functions of different components are described in 3.4.

3.1 Preliminary Considerations

The nature of Knowledge Work requires Knowledge Workers to deal with unpredictable work processes. Information and knowledge that during these processes are valuable assets that require management and distribution between Knowledge Workers. Since Knowledge Workers are experts in their domain, activities outside of their domain mean additional efforts and unnecessary risks.

Process-centric Knowledge Work approaches support Knowledge Work through predefined process work flows, restricting Knowledge Workers in adaptive process organisation. BPM requires Knowledge Workers to design models of processes, forcing them to leave their domain of expertise and lowering their productivity even further.

In contrast data-centric Knowledge Work approaches focus on the support of Knowledge Work in context of given data instead of pre-defined work flows. This allows Knowledge Workers to organise processes in an adaptive way which allows them to stay in their domain of expertise and raising the productivity of Knowledge Workers.

Comparison of process-centric Knowledge Work approaches with data-centric Knowledge Work approaches indicates that a data-centric approach is more qualified to meet the requirements Knowledge Work and Applicability.

ACM turned out to be especially applicable for adaptive process organization, knowledge management, and for supporting collaboration among Knowledge Workers. ACM combines goals, tasks, and content belonging to a specific case. Users define task states and schedule the execution of specific tasks. Templates provide reusable cases which can be modified by users during the runtime of a case. Content is stored in relation to cases and accessed by different participants which enables collaboration and raises productivity of Knowledge Workers.

ACM fulfils the requirement Knowledge Work to a satisfying degree. Thus ACM is used as the starting point, building the foundation for an applicable solution. Because ACM does not fulfil all defined requirements, a combination with other approaches is necessary.

Dedicated Crowdsourcing approaches implement Crowdsourcing in a specific field of application. The concept uses dedicated Crowdsourcing approaches to meet the requirement Crowdsourcing. Requirement Multiplatform demands that a solution uses multiple Crowdsourcing platforms. This means that the concept has to describe a solution that combines different dedicated Crowdsourcing approaches.

Intermediary Crowdsourcing approaches demonstrate a solution for the implementation of the requirement Multiplatform. In particular, CrowdSearcher has proofed that this requirement can be met in a satisfying way. To the best of my knowledge there is no intermediary Crowdsourcing approach - enabling the use of various and arbitrary, existing and future Crowdsourcing platforms - that can be combined with an ACM based system. Thus, the concept describes a custom intermediary Crowdsourcing approach to meet this specific requirement.

The evaluations in 2.3 have also shown that dedicated Crowdsourcing approaches and intermediary Crowdsourcing approaches are solutions to meet the requirement Quality. Thus, both approaches are used to provide applicable mechanisms.

In summary the concept uses a system based on ACM to implement the requirement Knowledge Work, i.e. Adaptive Case Management System (ACMS). To meet the requirements Crowdsourcing, Multiplatform, and Quality the ACMS is combined with a custom intermediary Crowdsourcing approach (i.e. Crowdsourcing Engine) that connects the ACMS with different dedicated Crowdsourcing approaches (i.e. Crowdsourcing Services).

Figure 3.1 illustrates a simplified view of the concept visualizing the combination of different approaches and their derived representatives.

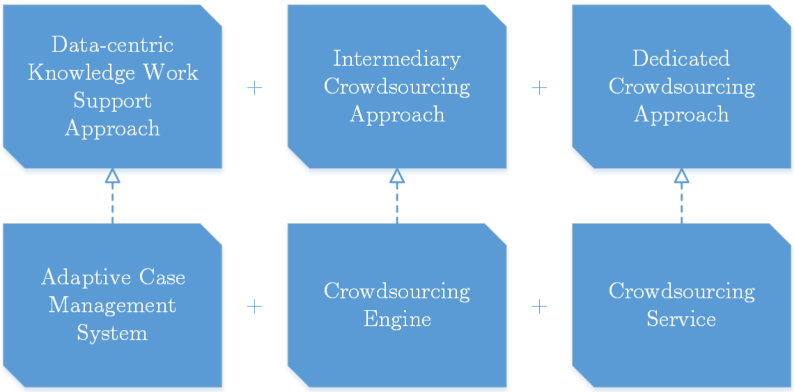
Data-centric Knowledge Work Support Approach
+
Intermediary Crowdsourcing Approach
+
Dedicated Crowdsourcing Approach
Adaptive Case Management System
+
Crowdsourcing Engine
+
Crowdsourcing Service

Figure 3.1: Simplified view of the concept visualizing the combination of different approaches and their derived representatives

3.2 Architecture

The preliminary considerations in 3.1 have introduced basic components of the concept, i.e. ACMS, Crowdsourcing Engine, Crowdsourcing Services. This section introduces the architecture of the concept by elaborating components that are required for an holistic solution.

The requirement Applicability, indicates that Knowledge Workers need suitable interfaces for Crowdsourcing within an ACMS. This means possible applications of Crowdsourcing for specific elements of an ACMS have to be found. Candidates are tasks and goals of an ACMS, as defined in 2.2.1.

As mentioned before, detailed processes of Knowledge Work are difficult to predict. Finding applications of Crowdsourcing in an ACMS by analysing Knowledge Work is inapplicable. Instead, the classification by Schenk & Guittard [49] is used to abstract the application of Crowdsourcing as follows:

- Using Crowdsourcing for simple tasks
- Using Crowdsourcing for complex tasks
- Using Crowdsourcing for creative tasks

Simple tasks are easy to describe and do not require a high level of expertise and effort. These tasks are executed by a large pool of workers. Using Crowdsourcing for simple tasks enables Knowledge Workers to find cost-efficient work force for a task, e.g. processing of data.

Complex tasks are characterized by uncertainty. Processing of a complex task and its outcome are unpredictable [10]. The crowd can be used to solve complex tasks by contributing their individual expertise and skills. Using Crowdsourcing for complex tasks allows Knowledge Workers to solve tasks outside their expert domain, e.g. solve scientific problem.

Creative tasks are characterized by their creativeness and the uniqueness of solutions. In contrast to complex tasks they aim to benefit from the crowd's creativity rather than solving a specific problem. Crowdsourcing allows Knowledge Workers to retrieve new ideas from the crowd and to generate creative content to solve creative tasks.

Tasks of an ACMS can be used as an interface for Knowledge Workers to use Crowdsourcing for simple, complex and creative tasks. This approach allows Knowledge Workers to find proper work force for complex and creative problems and get assistance in time-consuming processes. While a task provides no functions for Knowledge Workers to use Crowdsourcing as defined in 2.1.2, the concept introduces a new element: the Crowdsourcing Task.

In the context of complex tasks it is possible to define goals as an additional interface for the use of Crowdsourcing in an ACMS. Crowdforge, introduced in 2.2.2, has shown that Crowdsourcing can be used to break complex tasks into simple tasks. Knowlegde Workers can use Crowdsourcing to identify tasks or Crowdsourcing Tasks in a goal. However, basic goals provide no functions for the use of Crowdsourcing as defined in 2.1.2. Thus, the concept introduces an additional element to the ACMS: the Crowdsourcing Goal.

The requirement Multiplatform demands the use of various and arbitrary, existing and future Crowdsourcing Services. The Crowdsourcing Engine allows interaction with various Crowdsourcing Services. To enable the use of future Crowdsourcing Services the solution needs a component with the option to add and modify Crowdsourcing Services as needed. The component Crowdsourcing Adapter encapsulates functions of a specific Crowdsourcing Service and allocates them to the Crowdsourcing Engine. This approach fulfils the requirement Multiplatform to a satisfying degree, improving the maintainability of different Crowdsourcing Services additionally.

As laid out in 2.3 the requirement Quality can be met by various approaches. Thus, multiple components are used to implement applicable mechanisms, e.g. Crowdsourcing Adapter and Crowdsourcing Service.

This examination has shown that the concept requires additional components to implement a holistic solution. In summary the components of the developed concept are:

1. Adaptive Case Management System
2. Crowdsourcing Task and Crowdsourcing Goal
3. Crowdsourcing Engine
4. Crowdsourcing Adapter
5. Crowdsourcing Service

Figure 3.2 visualizes the general composition of the concept.

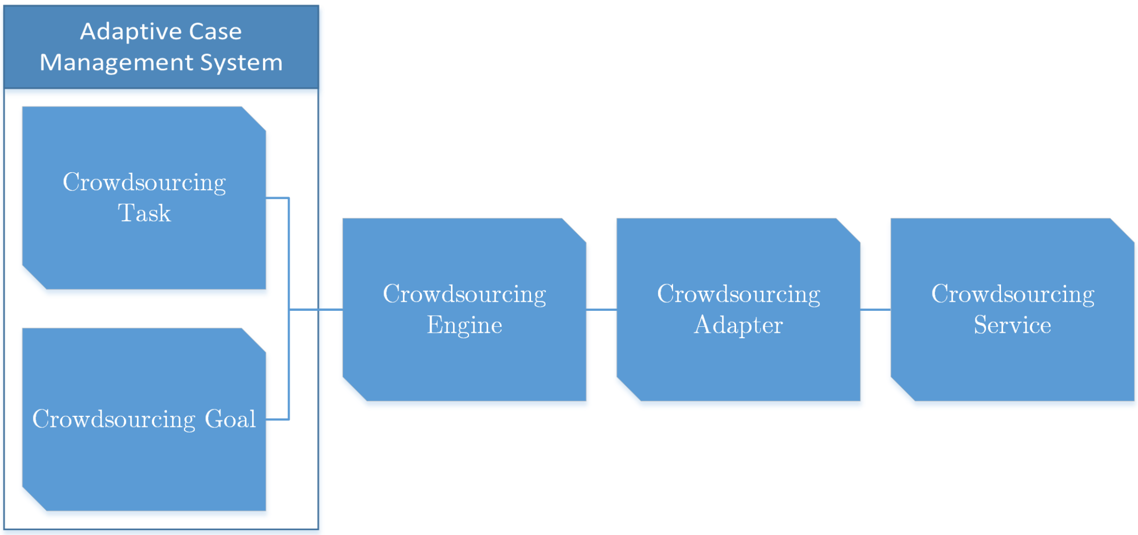

Figure 3.2: General composition of the concept

3.3 Components

Chapter 3.2 introduced the architecture of the concept. This section describes the components in detail and defines their responsibilities and dependencies.

3.3.1 Adaptive Case Management System

The component ACMS provides an appropriate environment that implements the requirement Knowledge Work and Applicability.

The ACMS implements a system that is modelled after the principles and definitions of ACM as introduced in 2.2. The basic elements of ACM are cases, goals and tasks. As laid out in 3.2 these elements are not suitable for the use of Crowdsourcing in an ACMS. Thus, the concept extends the ACMS by introducing new elements: Crowdsourcing Task and Crowdsourcing Goal. Figure 3.3 illustrates the composition of such an ACMS.

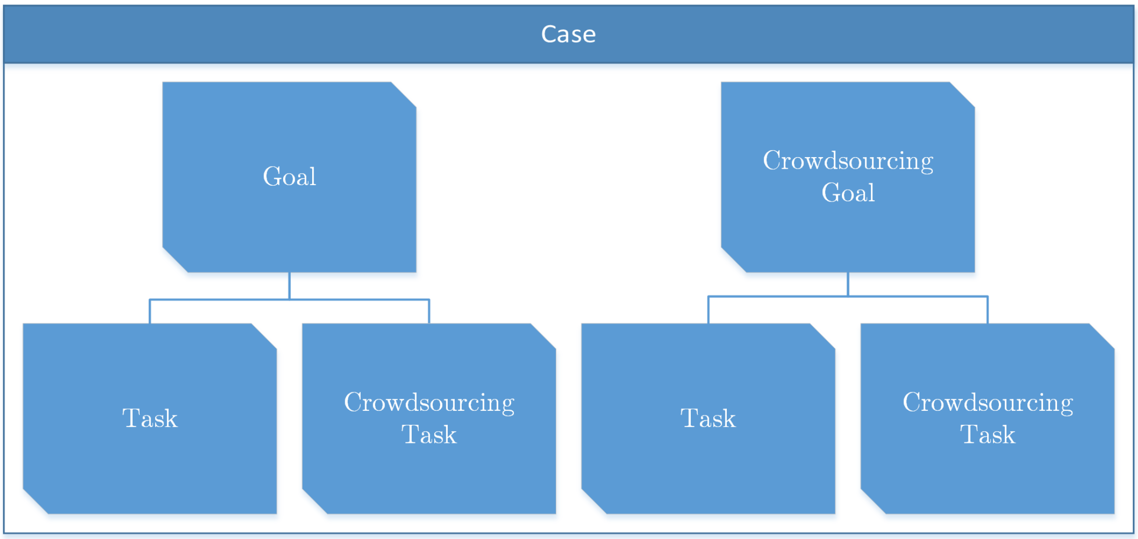

Figure 3.3: Composition of a case in an ACMS extended by Crowdsourcing Task and Crowdsourcing Goal

In context of the concept the component ACMS is responsible to enable and coordinate transactions between the component Crowdsourcing Task, the component Crowdsourcing Goal and the component Crowdsourcing Engine. Further the ACMS is in charge of providing additional functionality e.g. enable storage and retrieval of data that is related to a Crowdsourcing Task or a Crowdsourcing Goal.

3.3.2 Crowdsourcing Task and Crowdsourcing Goal

The component Crowdsourcing Task and the component Crowdsourcing Goal are used by Knowledge Workers to use Crowdsourcing in an ACMS.

Crowdsourcing Task

The Crowdsourcing Task outlines an interface within the ACMS for Knowledge Workers to define an arbitrary type of task that can be processed by using Crowdsourcing, i.e. simple tasks, complex task, creative task. The Crowdsourcing Task uses the Crowdsourcing Engine to announce a task to various Crowdsourcing Services and to retrieve results from the crowd.

While extending a basic task of an ACMS, the Crowdsourcing Task generally has a title and a description. The description of a Crowdsourcing Task is used to define what is expected to be done by the crowd. Following the requirements introduced in 2.1.2 and 2.1.4, the Crowdsourcing Task has several parameters more. Thus, the Crowdsourcing Task defines a recompense for the participation as well as the expected quality of results used by the Crowdsourcing Adapter and Crowdsourcing Service to evaluate contributors and results.

The requirement Multiplatform means that a solution has to enable to use of various Crowdsourcing Services. Each Crowdsourcing Service provides different features to use Crowdsourcing in a different field of application. The Crowdsourcing Task provides Knowledge Workers a set of operations that reflect features of a specific Crowdsourcing Services, e.g. text translation, data classification, solving of scientific problems. Thereby each operation adds specific parameters to the Crowdsourcing Task, e.g. set of options for a classification. The operations that can be used in a Crowdsourcing Task are defined by the Crowdsourcing Engine and are implemented by a specific Crowdsourcing Service. This approach enables Knowledge Workers to use features of various Crowdsourcing Services in an ACMS without having to leave their domain of expertise or switch context between different applications.

In summary, a Crowdsourcing Task is defined by the following parameters:

- Title
- Description
- Recompense for participation
- Expected quality
- Operations

In context of the concept the component Crowdsourcing Task is responsible for receiving the definitions of a task given by Knowledge Workers and the delivery of these definitions to the Crowdsourcing Engine. Further, the Crowdsourcing Task is responsible for the retrieval and visualization of the results delivered by the Crowdsourcing Engine. The features to announce the definitions to Crowdsourcing Services and to retrieve results from Crowdsourcing Services are implemented by the Crowdsourcing Engine.

Crowdsourcing Goal

The Crowdsourcing Goal is an additional interface within the ACMS that enables Knowledge Workers to use Crowdsourcing to identify tasks or Crowdsourcing Tasks of a goal.

The Crowdsourcing Goal requires identical parameters as the Crowdsourcing Task, e.g. title, description, recompense, quality of results. As stated before, a goal defines an overall achievement for a case. Thus, a goal can be considered a complex task. This means that the Crowdsourcing Goal requires no operations that specify the type of task. The description of the Crowdsourcing Goal is used to give a general definition of the goal. The crowd is analyzes the description and identifies potential tasks of the goal.

In context of the concept the component Crowdsourcing Goal is responsible for the recording of the definitions of a goal given by Knowledge Workers and the delivery of these definitions to the Crowdsourcing Engine. Further, the Crowdsourcing Goal is responsible for processing of the results, e.g. adding identified tasks or Crowdsourcing Tasks to a goal.

3.3.3 Crowdsourcing Engine

Because of the diversity of Crowdsourcing Services the concept requires a component which enables the coordination of transactions between the Crowdsourcing Task or Crowdsourcing Goal and various Crowdsourcing Services. The Crowdsourcing Engine illustrates an intermediary layer which routes transactions and data between these components.

These transactions involve to announce a Crowdsourcing Task or Crowdsourcing Goal to Crowdsourcing Services and deliver results from Crowdsourcing Services to a Crowdsourcing Task or Crowdsourcing Goal.

Announcing a Crowdsourcing Task, as visualized in Figure 3.4, requires the instantiation of specific Crowdsourcing Adapters in context of specific operations. The Crowdsourcing Adapter standardizes transactions and data between the Crowdsourcing Engine and a specific Crowdsourcing Service. As stated in 3.3.2, the Crowdsourcing Engine defines a set of operations that represent features of various Crowdsourcing Services. Thus, each operation is implemented by a specific Crowdsourcing Service. The Crowdsourcing Engine processes the data provided by a specific Crowdsourcing Task and instantiates Crowdsourcing Adapters in correspondence to defined operations.

The instantiation of different Crowdsourcing Adapters means to split a Crowdsourcing Task into multiple sub-tasks, and distribute the processing of the Crowdsourcing Task to different Crowdsourcing Services. The Crowdsourcing Engine is responsible to create these sub-tasks by duplicating basic parameters of a Crowdsourcing Task (e.g. title, description, quality, amount), assign one or more of the operations that have to be processed, and hand these sub-tasks over to the Crowdsourcing Adapters which are responsible to process the assigned operations through a specific Crowdsourcing Service.

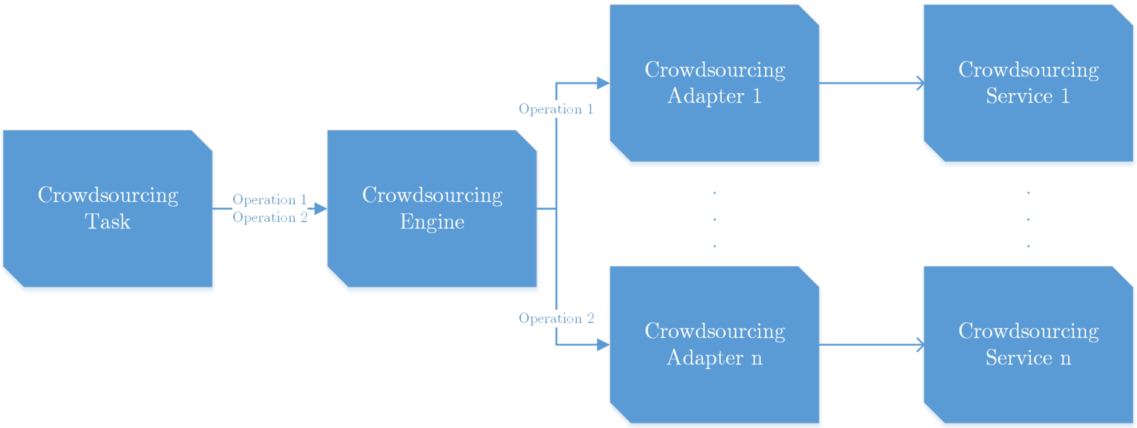

Figure 3.4: Instantiation of different Crowdsourcing Adapters based on defined operations

Distributing a Crowdsourcing Task to several Crowdsourcing Services, as visualized in Figure 3.5, requires retrieval and combination of results from different sources. Each Crowdsourcing Service provides results in a different format and structure. The Crowdsourcing Adapter has to transform the results received by the Crowdsourcing Service and deliver them to the Crowdsourcing Engine in a standardized format and structure. This transformation also means to reflect the quality of results, e.g. best solution, all solutions. The Crowdsourcing Engine combines the different standardized results and delivers them to the Crowdsourcing Task as an aggregated result.

In context of the concept the component Crowdsourcing Engine has the following responsibilities:

- To coordinate transactions between the Crowdsourcing Task or Crowdsourcing Goal and various Crowdsourcing Adapters.
- To process data provided by a Crowdsourcing Task or Crowdsourcing Goal, and Crowdsourcing Adapter.

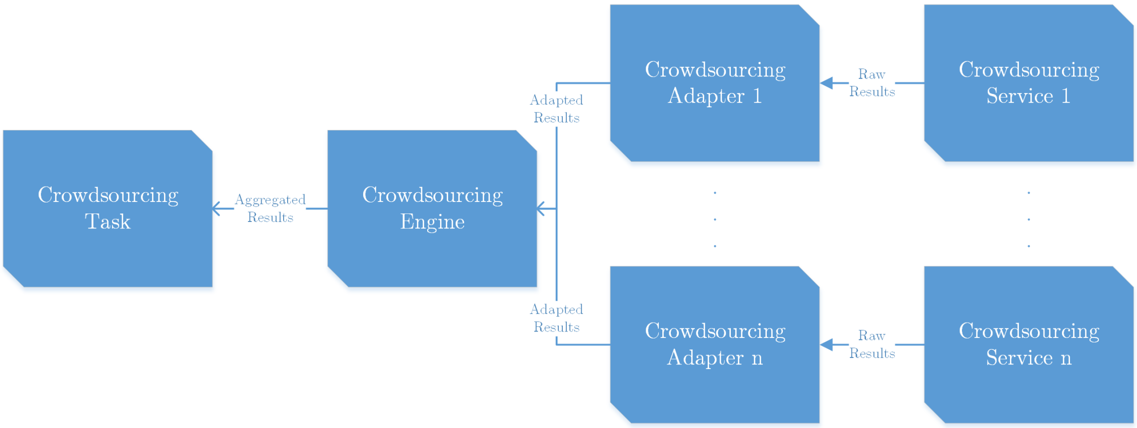

Figure 3.5: Aggregation of results retrieved from different Crowdsourcing Services

- To instantiate appropriate Crowdsourcing Adapters in context of the data delivered by the Crowdsourcing Task or Crowdsourcing Goal
- To split and distribute a Crowdsourcing Task or Crowdsourcing Goal to different Crowdsourcing Adapters.
- To provide functions to announce a Crowdsourcing Task or Crowdsourcing Goal, or to retrieve results for a Crowdsourcing Task or Crowdsourcing Goal.
- To define a set of operations which reflect the features of specific Crowdsourcing Services.
- To define standardized format and structure of data reflecting the results delivered to a Crowdsouring Task or Crowdsourcing Goal.

3.3.4 Crowdsourcing Adapter

The Crowdsourcing Engine enables the use of Crowdsourcing Services by interacting with the API of a specific Crowdsourcing Service. Each Crowdsourcing Service specifies a different API with different functions and different format and structure of data. The concept needs a component that allows bidirectional communication between the Crowdsourcing Engine and the Crowdsourcing Service. This implies to transform data and execute appropriate functions in both directions.

The Crowdsourcing Adapter bundles functions of a particular Crowdsourcing Service in a maintainable component and enables transactions between the Crowdsourcing Engine and a specific Crowdsourcing Service. Each Crowdsourcing Adapter provides a set of standardized functions for the Crowdsourcing Engine, e.g. announce a task or retrieve results. The Crowdsourcing Adapter maps these standardized functions to functions provided by a specific Crowdsourcing Service.

Before transferring data to the Crowdsourcing Engine or the Crowsourcing Service the Crowdsourcing Adapter has to transform the data into a specific format and structure. Announcing a Crowdsourcing Task requires the transformation of data delivered by the Crowdsourcing Task into a specific format and structure defined by the Crowdsourcing Service. Retrieving results from Crowdsourcing Services requires the transformation of data delivered by the Crowdsourcing Service into a specific format and structure defined by the Crowdsourcing Engine. The Crowdsourcing Adapter implements specific functions to enable this transformations.

Allowing Knowledge Workers to stay in their knowledge domain means that not all specific parameters can be implemented by the Crowdsourcing Task or Crowdsourcing Goal. Thus, the Crowdsourcing Adapter is responsible for the definition of default parameters in context of a specific Crowdsourcing Service.

The Crowdsourcing Adapter can be used to implement the requirement Quality. Using statistics delivered by a specific Crowdsourcing Service allows to evaluate the quality of results. A reasonable method to measure the quality is the relation of appropriate and inappropriate results delivered by a specific contributor. This approach enables to evaluate the quality of results based on the reliability of contributors. More methods can be found in several papers [8, 57] or applications [54].

In context of the concept the component Crowdsourcing Adapter has the following responsibilities:

- To map functions provided to the Crowdsourcing Engine to functions provided by a specific Crowdsourcing Service.
- To enable bidirectional transformation of data between the Crowdsourcing Engine and a specific Crowdsourcing Service.
- To apply mechanisms to evaluate the quality of results retrieved by a Crowdsourcing Service.

3.3.5 Crowdsourcing Service

The component Crowdsourcing Service implements Crowdsourcing in a specific field of application.

The Crowdsourcing Service provides an API used by the Crowdsourcing Adapter to announce a Crowdsourcing Task or Crowdsourcing Goal to and retrieve results from the crowd. Thus, the Crowdsourcing Service is responsible for defining format and structure of data that reflects definitions and results of a specific Crowdsourcing Task or Crowdsourcing Goal.

The Crowdsourcing Service is responsible to implement the requirement Quality. Methods that can be used to evaluate the quality of results and performance of contributors have been described in 3.3.4. As stated in 2.3, not all Crowdsourcing platforms provide mechanisms to evalute the quality of results and contributors. In this case the Crowdsourcing Adapter is responsible to implement appropriate mechanisms.

In context of the concept the component Crowdsourcing Service has the following responsibilities:

- To enable Crowdsourcing in a specific field of application
- To provide an API to announce a Crowdsourcing Task or Crowdsourcing Goal to the crowd and receive results from the crowd
- To apply mechanisms to evaluate the quality of results retrieved by the crowd

3.4 Processes

The architecture of the concept has been introduced in 3.2, while 3.3 has described the different components in detail. This chapter describes processes that combine functions of the different components in a comprehensive way.

The following processes are necessary for the application of the concept:

1. Announcing of a Crowdsourcing Task to a Crowdsourcing Service
2. Retrieval of results for a Crowdsourcing Task

3.4.1 Announcing of a Crowdsourcing Task to a Crowdsourcing Service

This section describes the process of announcing a Crowdsourcing Task to a Crowdsourcing Service.

Initially the Crowdsourcing Engine retrieves the defined parameters for a specific Crowdsourcing Task. The Crowdsourcing Engine processes the delivered data, analyzes defined operations and instantiates appropriate Crowdsourcing Adapters. The Crowdsourcing Engine then transmits the defined parameters to a instantiated Crowdsourcing Adapter. The Crowdsourcing Adapter transforms the data retrieved by the Crowdsourcing Engine into the format and structure expected by a specific Crowdsourcing Service. The Crowdsourcing Adapter then invokes the Crowdsourcing Service to announce a task, delivering the transformed data. Finally, the Crowdsourcing Service announces the defined Crowdsourcing Task to the crowd.

Figure 3.6 visualizes the process of how to announce a Crowdsourcing Task to a Crowdsourcing Service.

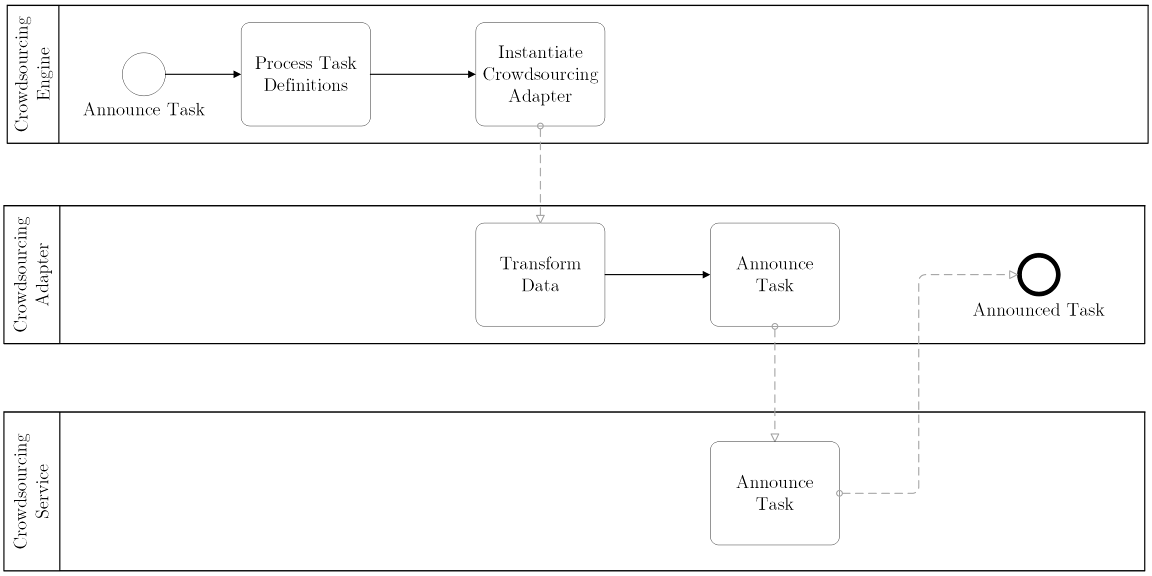

Figure 3.6: Announcing a Crowdsourcing Task to a Crowdsourcing Service

3.4.2 Retrieval of Results for a Crowdsourcing Task

This section describes the process of how to retrieve results for a Crowdsourcing Task.

Initially the Crowdsourcing Engine receives the request to retrieve results for a specific Crowdsourcing Task. The Crowdsourcing Engine then instantiates associated Crowdsourcing Adapters to retrieve results from a specific Crowdsourcing Adapter. The Crowdsourcing Adapter on his part invokes functions to retrieve and process results from a specific Crowdsourcing Service. The Crowdsourcing Service hands the results of the crowd over to the Crowdsourcing Adapter in a specific format and structure. The Crowdsourcing Adapter then transforms the results in a standardized format and structure defined by the Crowdsourcing Engine. The standardized results are delivered to the Crowdsourcing Engine that combines the results from different Crowdsourcing Adapters. Finally, the Crowdsourcing Engine delivers the aggregated results to the specific Crowdsourcing Task.

Figure 3.7 visualizes the process of how to retrieve results for a specific Crowdsourcing Task.

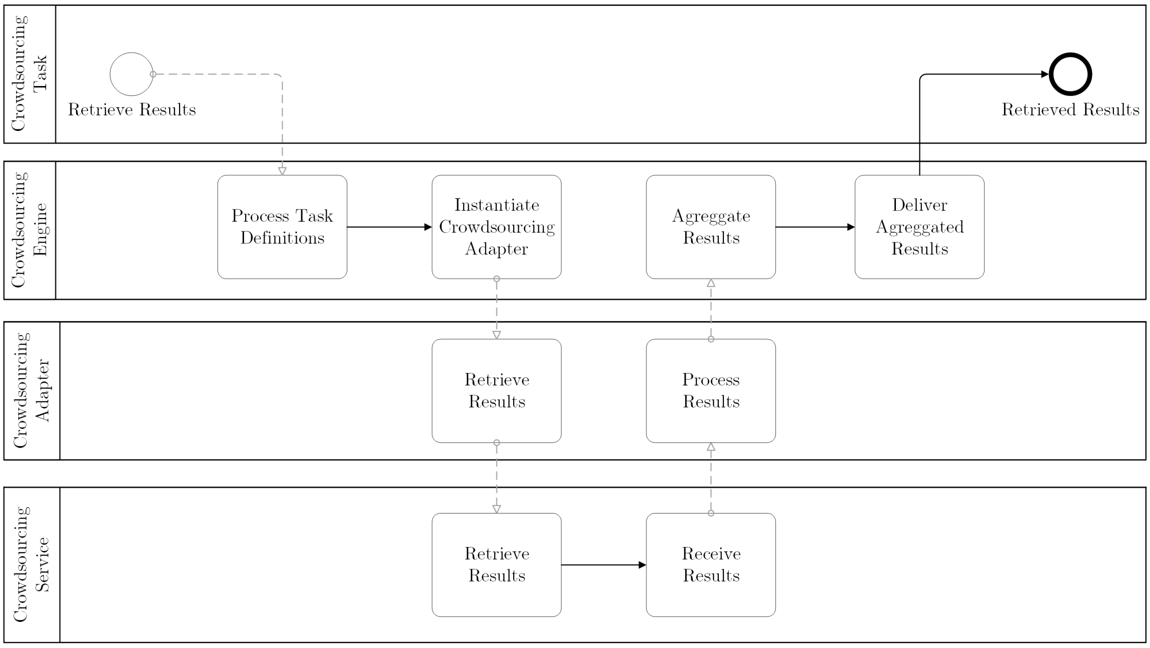

Figure 3.7: Transactional view of the concept

3.5 Summary

The concept introduces a combination of data-centric Knowledge Work approaches, intermediary Crowdsourcing approaches and dedicated Crowdsourcing approaches. Preliminary considerations in chapter 3.1 show that this approach allows the formulation of a holistic solution that fulfils all requirements to a satisfying degree.

The examination has shown that ACM is especially applicable, fulfilling the requirement Knowledge Work to a satisfying degree. Thus ACM is used to define a basic system for a solution, i.e. ACMS.

The examination also show that no intermediary Crowdsourcing approach exists that can be combined with an ACMS to allow the use of various Crowdsourcing platforms. Thus, the concept introduces a custom intermediary Crowdsourcing approach, the Crowdsourcing Engine.

The concept extends an ACMS through Crowdsourcing Task and Crowdsourcing Goal to allow Knowledge Workers to use Crowdsourcing in an ACMS. The Crowdsourcing Task enables Knowledge Workers to use Crowdsourcing Services for several types of tasks without forcing them to leave their domain of expertise. The Crowdsourcing Goal allows Knowledge Workers to identify tasks or Crowdsourcing Tasks of an goal.

The Crowdsourcing Engine connects the Crowdsourcing Task and the Crowdsourcing Goal with various Crowdsourcing Services interacting with specific Crowdsourcing Adapters. The Crowdsourcing Service implements Crowdsourcing in a specific field of application, providing functions to a specific Crowdsourcing Adapter through an API. The Crowdsourcing Adapter transforms data and maps corresponding functions between a specific Crowdsourcing Service and the Crowdsourcing Engine. This approach allows the implementation of a solution that enables the use of arbitrary exisiting and future Crowdsourcing platforms.

The Crowdsourcing Adapter and the Crowdsourcing Service implement mechanisms to fulfil the requirement Quality. The Crowdsourcing Task defines a treshold that allows to evaluate the quality of results delivered by the crowd.

In summary, the concept shows a solution for Knowledge Workers to use various Crowdsourcing platforms in an ACMS without forcing them to leave their knowledge domain or to switch between different applications.

The following chapter describes the implementation of a prototype, demonstrating the applicability of the concept. The chapter discusses general aspects of the implementation in context of the developed concept as well as aspects in regard of the implemented prototype.

4 Implementation

This chapter introduces an implemented prototype of the concept, described in 3.

The implementation is based on the existing ACMS „VSRCM“ developed at the professorship „Verteilte und Selbstorganisierende Rechnersysteme (VSR)“ at the Technical University of Chemnitz. Main developer of VSRCM is MSc Sebastian Heil. VSRCM is a web-based application hosted on an IIS, and is implemented in C# using the framework .NET MVC 4. Authentification of users is realized through the use of WebID. Further, VSRCM enables the use of web-services such as Google Drive to store and retrieve content, and Trello for advanced activity management. Additionally, the application implements Collaborative Adaptive Case Management with Linked Data [28]. The approach leverages WebID and RDF to enable Knowledge Workers to identify automatically those experts that can contribute to an ACM case.

The application is presented to the user through the GUI of a web browser, visualizing HTML through CSS and providing certain functionalities through Javascript. The instructions by the user are delivered to the server through HTTP requests. Following the principle of MVC, server-side controllers process the requests and execute appropriate functions. Corresponding data of the ACMS is stored in XML files.

Implementation of a solution that requires minimal changes to the ACMS is preferred. For this purpose the prototype is combining the component Crowdsourcing Engine and Crowdsourcing Adapter in a framework, i.e. Multiplatform Crowdsourcing Framework.

The concept further requires extension of the ACMS through Crowdsourcing Task and Crowdsourcing Goal, enabling Knowledge Workers to use Crowdsourcing in an ACMS. Within the scope of this thesis, the prototype demonstrates the implementation of the component Crowdsourcing Task in an ACMS.

Chapter 4.1 introduces the implementation of the Multiplatform Crowdsourcing Framework in detail. The implementation of component Crowdsourcing Task is introduced in chapter 4.2.

4.1 Multiplatform Crowdsourcing Framework

The Multiplatform Crowdsourcing Framework combines the components Crowdsourcing Engine and Crowdsourcing Adapters as a comprehensive component. This approach enables the combination of functions of the different components, serving them to the ACMS in a holistic way. The design, architecture and components of the Multiplatform Crowdsourcing Framework are introduced in the following sections.

4.1.1 Design

Considering the composition of ACMS, Multiplatform Crowdsourcing Framework and Crowdsourcing Service, the following options can be derived:

A) Implementing the Multiplatform Crowdsourcing Framework inside of the ACMS, i.e. internal framework

B) Implementing the Multiplatform Crowdsourcing Framework outside of the ACMS, i.e. external framework

As mentioned in chapter 3.1, there is currently no approach that allows the use of various and arbitrary, existing and future Crowdsourcing platforms. Thus, it is preferable to improve the reusability of the Multiplatform Crowdsourcing Framework for further implementations.

Realizing the Multiplatform Crowdsourcing Framework inside of the ACMS requires additional changes in an existing ACMS, and means reduced reusability for further implementations. In contrast, realizing the Multiplatform Crowdsourcing Framework outside of the existing ACMS requires minimal changes in the ACMS and enables further implementation to leverage the solution.

The prototype implements the Multiplatform Crowdsourcing Framework in PHP as an external framework (Figure 4.1). Functions of the framework can be accessed through an API.

The concept introduces operations which allow Knowledge Workers to use Crowdsourcing for simple tasks, complex tasks, and creative tasks. The prototype demonstrates the implementation of the operations „Writing“ and „Categorization“ by using the Crowdsourcing Services Amazon Mechanical Turk and CrowdFlower, since both provide an API and features to process the operation.

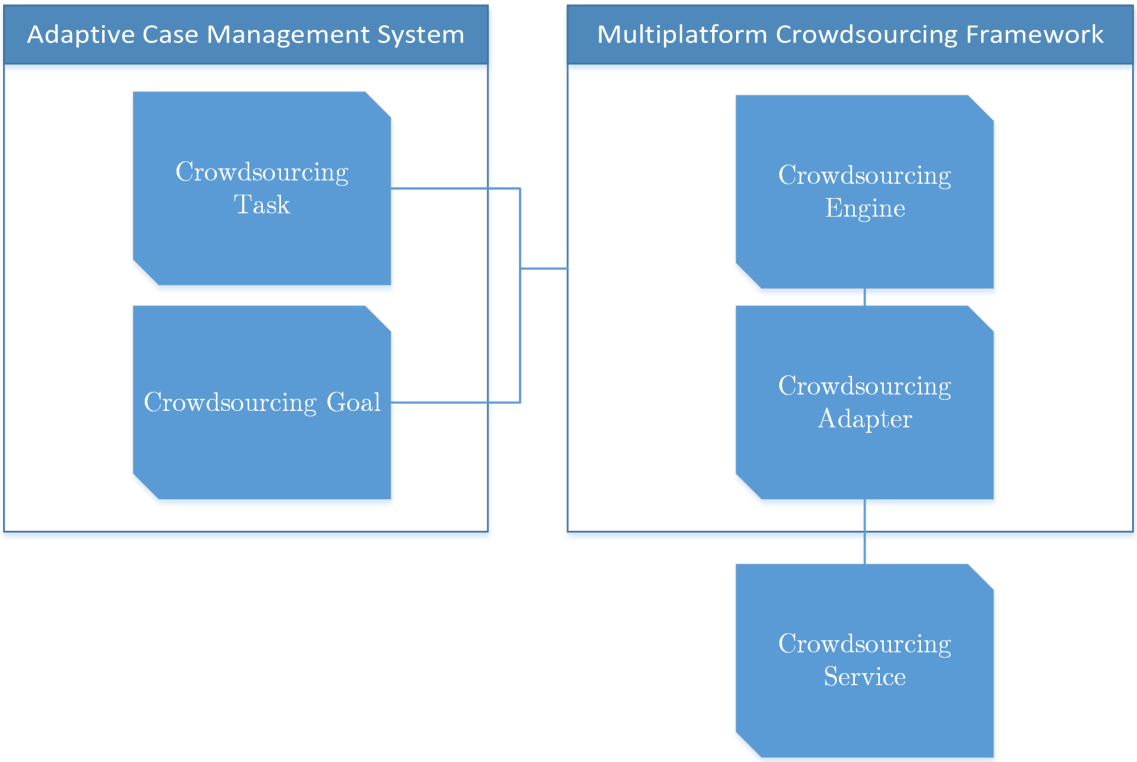

Figure 4.1: Composition of ACMS, Multiplatform Crowdsourcing Framework and Crowdsourcing Service

The operation Writing allows Knowledge Workers to use Crowdsourcing for writing arbitrary text, e.g. summaries of large papers or comments about papers. The operation is implemented by the use of the API of Amazon Mechanical Turk [55].

The operation Categorization allows Knowledge Workers to use Crowdsourcing for categorizing data, e.g. categorizing a picture or categorizing a text. The operation is implemented by the use of the API of Crowdflower [13].

Further the implemented framework uses the data format JSON. This format allows the flexible modification of data structures while enabling the processing of data through front-end (e.g. Javascript) and back-end programming languages such as PHP.

4.1.2 Architecture

The Multiplatform Crowdsourcing Framework consists of the following components:

- Controller
- Helper
- CrowdsourcingEngine
- CrowdsourcingAdapterFactory
- CrowdsourcingAdapter
- CrowdflowerAdapter
- MechanicalTurkAdapter
- Crowdflower
- MechanicalTurk

Figure 4.2 on page 54 visualizes the composition of the Multiplatform Crowdsourcing Framework. The following chapter describes the components in detail.

4.1.3 Components

Chapter 4.1.2 has introduced the components of the framework. This chapter describes the components in detail.

Controller

The component Controller receives function calls from external systems and routes them to the component CrowdsourcingEngine. Further this component is responsible to return an response to the external system in regard of the results of the called functions.

Helper

The component Helper is used as a helper class for the framework, providing functions to save and load tasks and settings.

CrowdsourcingEngine

The component CrowdsourcingEngine is used as the main component to create a specific Crowdsourcing Task and to receive Results for a specific Crowdsourcing Task. The CrowdsourcingEngine uses the component helper to load the settings file „settings.main.json“ and to store data related to a task in a JSON file. The function createTask is used to to create sub-tasks of a Crowdsourcing Task in regard of defined operations. To determine which Crowdsourcing Adapter is responsible for which operation the component uses the function getPreferedServiceForOperation. If a sub-task is created sucessfully by a Crowdsourcing Adapter the component receives an id for the sub-task which is used by a specific Crowdsourcing Service to identify the sub-task. To retrieve and aggregate results of a Crowdsourcing Task the component uses the function getResults. Therefore the function loads the stored Crowdsourcing Task and uses the stored id's of the sub-tasks to retrieve results from a specific Crowdsourcing Adapter. To aggregate results from different Crowdsourcing Adapters the component defines a structure that includes all results and if defined the best result. Additionally the component implements the function readTask to retrieve a stored Crowdsourcing Task.

CrowdsourcingAdapterFactory

The component CrowdsourcingAdapterFactory is used by the component CrowdsourcingEngine to initialise the components CrowdflowerAdapter and MechanicalTurkAdapter through a factory pattern.

CrowdsourcingAdapter

The component CrowdsourcingAdapter is used as an interface class for the component CrowdflowerAdapter and MechanicalTurkAdapter. This component standardizes functions that can be accessed by the Crowdsourcing Engine. These functions include to create a task for a specific Crowdsourcing Service (createTask) and to retrieve results from a specific Crowdsourcing Service (retrieveResults).

CrowdflowerAdapter

The component CrowdflowerAdapter implements the Crowdsourcing Adapter that is responsible for transactions with the Crowdsourcing Service Crowdflower. The component provides the standardized functions createTask and getResults as well as specific functions to transform operations (buildCML, translateOperation). The function buildCML transforms the definitions of a Crowdsourcing Task into the format CML (Crowdflower Markup language). The markup is used by CrowdFlower to define instructions of a job. The CrowdflowerAdapter uses the component Crowdflower to create a job, to set channels and data of the job, and finally to order the job on the Crowdsourcing platform CrowdFlower. If a job is created successfully the component returns the id of the created job to the component CrowdsourcingEngine. Further the CrowdflowerAdapter uses the component Crowdflower to retrieve results for a job. The received data then is mapped to the standardized format and structure that is expected by the component CrowdsourcingEngine.

MechanicalTurkAdapter

The component MechanicalTurkAdapter implements the Crowdsourcing Adapter that is responsible for transaction with the Crowdsourcing Service Amazon Mechanical Turk. The component provides the standardized functions createTask and getResults as well as specific functions to transform operations (buildQuestionForm, translateOperation). The function buildQuestionForm transforms the definitions of a Crowdsourcing Task into the format QuestionForm. The format is used by Amazon Mechanical Turk to define a HIT. The MechanicalTurkAdapter uses the component MechanicalTurk to create a HIT on the Crowdsourcing platform Amazon Mechanical Turk. If a HIT is created successfully the component returns the id of the created job to the component CrowdsourcingEngine. Further the MechanicalTurkAdapter uses the component MechanicalTurk to retrieve results for a job. The received data then is mapped to the standardized format and structure that is expected by the component CrowdsourcingEngine.

Crowdflower

The component Crowdflower demonstrates the implementation of the Crowdsourcing Service Crowdflower. To call functions of the API of Crowdflower the component uses CURL.

MechanicalTurk

The component MechanicalTurk demonstrates the implementation of the Crowdsourcing Service Amazon Mechanical Turk. To call functions of the API of Crowdflower the component uses the function file_get_contents.

4.1.4 Processes

Announcing of a Crowdsourcing Task to a Crowdsourcing Service

The concept describes the principle to announce a Crowdsourcing Task to various Crowdsourcing Services.

Initially the Crowdsourcing Engine analyzes which operations have been used by a specific Crowdsourcing Task. Based on the defined operations, the Crowdsourcing Engine instantiates different Crowdsourcing Adapters, and splits the Crowdsourcing Task into multiple sub-tasks. Each sub-tasks then is assigned to a specific Crowdsourcing Adapter.

The prototype uses a settings file that defines which operations are implemented by which Crowdsourcing Adapter. Based on these definitions the component CrowdsourcingEngine determines which Crowdsourcing Adapters have to be instantiated.

In case of the operation Writing the component CrowdflowerAdapter is instantiated, in case of the operation Categorization the component MechanicalTurkAdapter is instantiated. The component CrowdsourcingEngine then splits the Crowdsourcing task into multiple sub-tasks and hands them over to the Crowdsourcing Adapters. Finally the component CrowdsourcingEngine stores the parameters and identifiers of sub-tasks of the Crowdsourcing Task through the component Helper.

Further, the concept describes that the Crowdsourcing Adapter transforms the data delivered by the Crowdsourcing Engine into a format and structure expected by a specific Crowdsourcing Service. The transformed data then is used to announce the Crowdsourcing Task to the Crowdsourcing Service and finally to the crowd. This approach also includes to transform parameters of a Crowdsourcing Task as well as specific operations. The implementation of the transformation is exemplified by the operation Writing.

The framework uses a numeric scale which represents the quality of results in the range of 0 to 100. The value 0 represents the lowest quality, the value 100 the best possible quality. The component CrowdflowerAdapter respective the component MechanicalTurkAdapter map this scale to the scale of the component Crowdflower respective MechanicalTurk. The API of the Crowdsourcing Service CrowdFlower provides three levels of performance[1] of contributors for a task: „Level 1 Contributors“, „Level 2 Contributors“, and „Level 3 Contributors“. The higher the level, the higher the performance of the contributors. The component Crowdflower maps the scale of the Crowdsourcing Engine to these three levels. The value 0 to 32 represents Level 1 Contributors, the values 33 to 66 Level 2 Contributors, the values 67 to 100 Level 3 Contributors. The API of the Crowdsourcing Service Amazon Mechanical Turk provides a mechanism to decide if a worker is eligable for contribution by defining a minimum percentage of HIT's which a specific worker has done and have been accepted by a requester. The mechanism uses a scale identical to the scale of the framework, thus no transformation is necessary.

As stated before the operation Writing is implemented by the Crowdsourcing Service Amazon Mechanical Turk. The API of Amazon Mechanical Turk defines the data format „QuestionForm“. The format is based on XML and provides a specific structure for the definition of a HIT. The component MechanicalTurkAdapter translates the definitions delivered by the component Crowdsourcing Engine into the data format QuestionForm through the function „buildQuestionForm“. This function builds the QuestionForm in general by settings basic parameters of the HIT (e.g. title, description) as well as „Questions“ which reflect the instructions of a HIT. Questions are used to define which operations have to be processed by the crowd. Therefor

[1] Performance represents the accuracy of a contributor, i.e. relation of appropriate and inappropriate contributions to a task

each operation is transformed with the function „translateOperation“. This function maps the parameters of an operation to the data structure „Question“. This structure requires to specify an identifier, a question and a type of answer, i.e. text or selection. In case of the operation Writing the type for the answer is text.

After transforming all operations the components CrowdflowerAdapter respective the component MechanicalTurkAdapter use the component Crowdflower respective MechanicalTurk to create a HIT respective Job, receiving a specific id. Finally the id is returned to the component CrowdsourcingEngine which stores the id's in relation to the Crowdsourcing Task.

Retrieval of Results for a Crowdsourcing Task

The concept describes the principle of aggregation of results from different Crowdsourcing Services. Therefor the Crowdsourcing Engine instantiates the assigned Crowdsourcing Adapters of sub-task of a Crowdsourcing Task and invokes them to retrieve results from a specific Crowdsourcing Service. The Crowdsourcing Adapter retrieves results in a specific format and structure, and transforms them into a standardized format and structure defined by the Crowdsourcing Engine. The transformed data is handed to the Crowdsourcing Engine which combines results from different Crowdsourcing Services and transmits them to the Crowdsourcing Task.

The Crowdsourcing Service CrowdFlower provides „Judgements“ for a specific Crowdsourcing Task, reflecting the results delivered by the crowd. The structure of the delivered results is similar to the structure defined by the component CrowdsourcingEngine. Thus the component CrowdflowerAdapter can map these results identically. The results of the component MechanicalTurk only reflect all results, thus the component MechanicalTurkAdapter can only provide all results to the component CrowdsourcingEngine. After aggregating the results of different Crowdsourcing Adapters the component CrowdsourcingEngine hands them over to the component Contoller. Finally the component controller returns the results to a requesting application.

4.1.5 Conclusion

The implemented components and functions demonstrate the applicability of the concept for the components Crowdsourcing Engine, Crowdsourcing Adapter, and Crowdsourcing Service. The following section describes the implementation of the component Crowdsourcing Task.

4.2 Crowdsourcing Task

The prototype extends the ACMS by the component Crowdsourcing Task, enabling Knowledge Workers the use of Crowdsourcing in an ACMS. The design, architecture and components of the Crowdsourcing Task are introduced in the following sections.

4.2.1 Design

As stated in 3, the Crowdsourcing Task is responsible to enable Knowledge Workers to define and announce a task to, and to retrieve and display results from the Crowdsourcing Engine. The implemented Crowdsourcing Task divides the interface into two sections: „Definitions“ and „Results“.

The section Definitions allows Knowledge Workers to set parameters of a specific Crowdsourcing Task (e.g. title, description, quality, amount) and to announce the Crowdsourcing Task. Further, the component allows to add and modify defined operations. The parameter quality of results is visualized through a three-part scale, mapped to the scale defined by the implemented Multiplatform Crowdsourcing Framework.

The section Results allows Knowledge Workers to retrieve results of a specific Crowdsourcing Task. Further this section visualizes the results of the Crowdsourcing Task, listed by their operations and highlighting best results in bold face.

4.2.2 Architecture

As mentioned before the ACMS VSRCM implements a MVC pattern. Thus, the Crowdsourcing Task follows this principle by implementing an appropriate structure of components. The implemented Crowdsourcing Task consists of the following components:

- View
- CrowdsourcingController
- CrowdsourcingProvider

Figure 4.3 visualizes the architecture of the Crowdsourcing Task in the VSRCM.

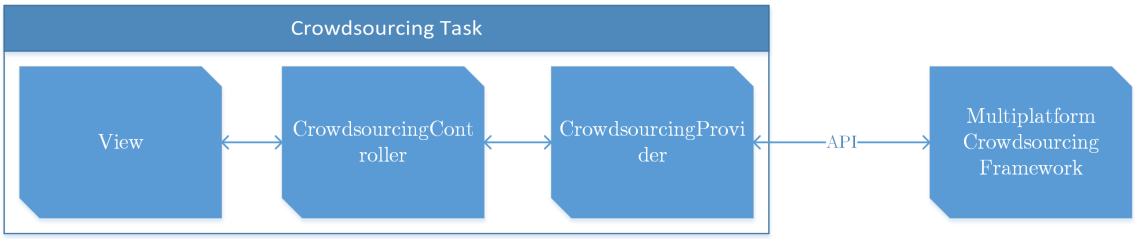

Figure 4.3: Architecture of the implemented Crowdsourcing Task

4.2.3 Components

View

The component View visualizes the sections Definitions and Results of the Crowdsourcing Task in the web browser through HTML and CSS. Further the component uses AJAX to call functions of the component CrowdsourcingController. This functions include to announce and to retrieve Results for the Crowdsourcing Task as well as retrieving definitions of a Crowdsourcing Task.

CrowdsourcingController

The component CrowdsourcingController routes functions and data received by the component View and hands them over to the component CrowdsourcingProvider. These functions include CreateTask, GetResults, and ReadTask.

CrowdsourcingProvider

The component CrowdsourcingController is responsible for transactions with the API of the implemented Multiplatform Crowdsourcing Framework, using the library „RestSharp“ to build HTTP requests. The component implements the functions CreateTask, GetResults, and ReadTask.

4.2.4 Processes

Announcing of a Crowdsourcing Task to a Crowdsourcing Service

The component View calls the function CreateTask of the CrowdsourcingController, delivering the id and the defined parameters of the Crowdsourcing Task. The component CrowdsourcingController then instantitates the component CrowdsourcingProvider and calls the function CreateTask. The component CrowdsourcingProvider on his side builds a HTTP post request with the parameters func, taskid and data.

The parameter func defines the function that has to be processed by the Multiplatform Crowdsourcing Framework to create the Crowdsourcing Task. The Parameters taskid and data represent the id and the defined parameters of the Crowdsourcing Task. The component CrowdsourcingProvider then executes the request and receives a response by the Multiplatform Crowdsourcing Framework. The response then is handed back to the component CrowdsourcingController and finally to the component View, that processes the response.

Retrieval of Results for a Crowdsourcing Task

The component View calls the function GetResults of the CrowdsourcingController, delivering the id of the specific Crowdsourcing Task. The component CrowdsourcingController then instantitates the component CrowdsourcingProvider and calls the function GetResults. The component CrowdsourcingProvider on his side builds a HTTP post request with the parameters func and taskid. The parameter func defines the function that has to be processed by the Multiplatform Crowdsourcing Framework to retrieve results for the Crowdsourcing Task. The Parameter taskid represents the id of the Crowdsourcing Task. The component CrowdsourcingProvider then executes the request and receives a response by the Multiplatform Crowdsourcing Framework which reflects the results of the Crowdsourcing Task. The response then is handed back to the component CrowdsourcingController and finally to the component View, that processes the response.

4.2.5 Conclusion

The implemented components and functions demonstrate the applicability of the concept for the component Crowdsourcing Task.

4.3 Summary

The introduced VSRCM and the implemented Multiplatform Crowdsourcing Framework as well as the implemented Crowdsourcing Task demonstrate the practicability of the concept. The following chapter evaluates whether the implemented prototype can solve the outlined problem, as stated in 1.2.

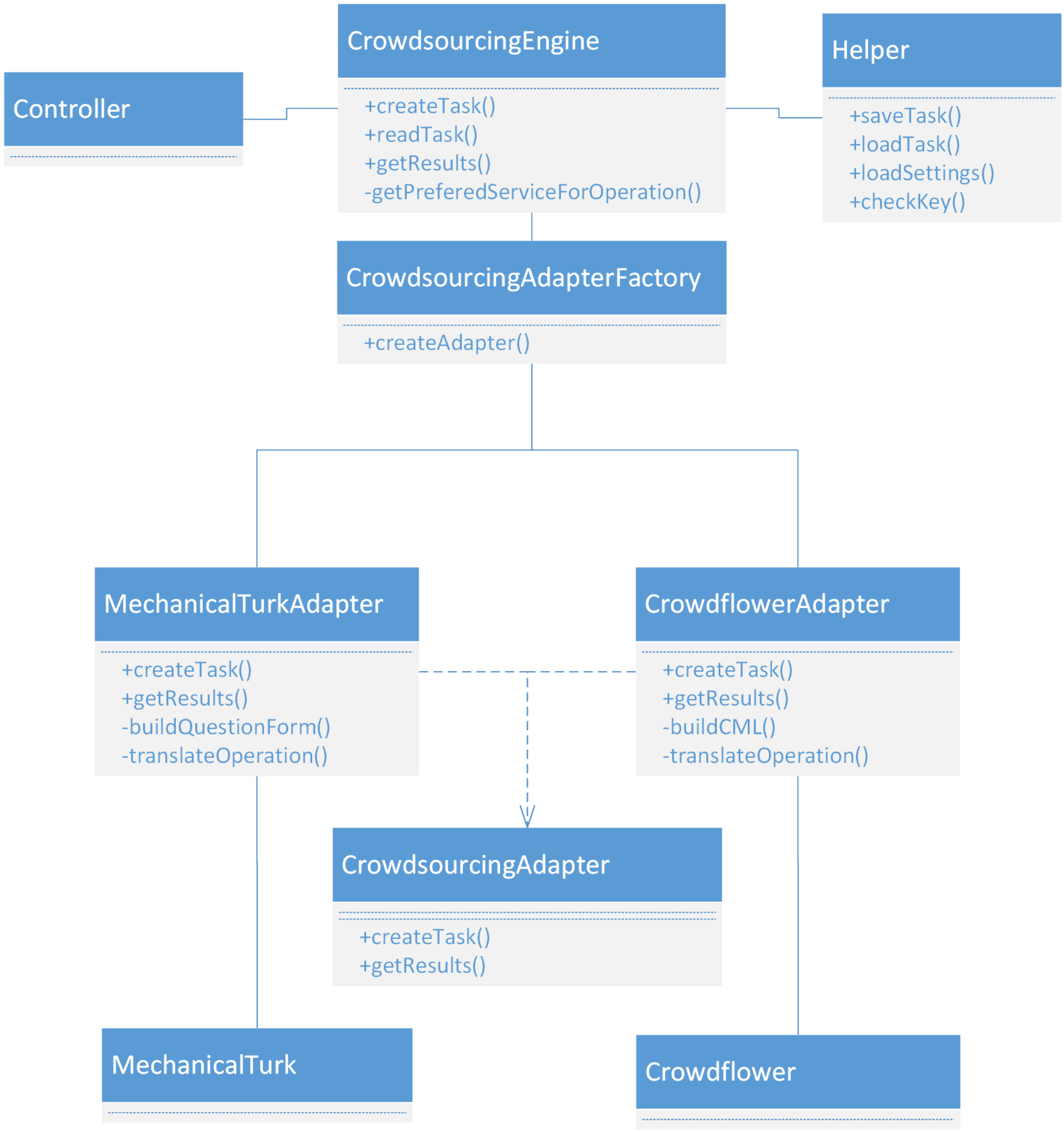

Figure 4.2: Architecture of the Multiplatform Crowdsourcing Framework

5 Evaluation

This chapter evaluates the implementation from chapter 4 based on the requirements introduced in chapter 2.1. The evaluation is used to show that the elaborated concept, as described in 3, provides a suitable solution to the stated problem in 1.2. The evaluation is summarized in table 5.1.

Knowledge Work

Knowledge Work is a creative and complex type of work that confronts Knowledge Workers with unpredictable work processes. Thus, Knowledge Workers need solutions that enable them to manage processes in an adaptive way. Further, they need an environment that supports them in management and distribution of knowledge and information.

The examinations have shown that data-centric Knowledge Work support approaches fulfil this requirements to a satisfying degree. Especially ACM is applicable to provide a suitable solution to Knowledge Workers.

The prototype uses VSRCM, a system based on the principles of ACM. VSRCM allows Knowledge Workers to manage processes through goals and tasks in an adaptive way. Content is stored in relation to a case which is shared between different Knowledge Workers.

Because VSRCM meets all standards, as defined in 2.1.1 (i.e. enable adaptive process organization, support knowledge management, support collaboration), the prototype is rated ●.

Crowdsourcing

Crowdsourcing provides valuable improvements to Knowledge Work. Thus, it is required that a solution allows Knowledge Workers to use Crowdsourcing, as defined in 2.1.2.

The concept introduced the components Crowdsourcing Task and Crowdsourcing Goal as possible interfaces which allow Knowledge Workers to use Crowdsourcing in an ACMS. The interfaces use several parameters (i.e. title,description,recompense,quality) for the definition of specific Crowdsourcing Tasks and Crowdsourcing Goals. Further, the Crowdsourcing Task introduced operations to reflect applications of Crowdsourcing for several types of tasks.

The prototype demonstrates the implementation of a Crowdsourcing Task in VSRCM which enables Knowledge Workers to announce a Crowdsourcing Task to the crowd and retrieve results from the crowd.

The argumentation indicates that the prototype can fulfil the requirement Crowdsourcing to a satisfying degree, thus the prototype is rated ●.

Multiplatform

Crowdsourcing is implemented by various Crowdsourcing platforms. The requirement defined in 2.1.3 demands that a solution has to enable the use of various and arbitrary, existing and future Crowdsourcing platforms.

The examinations have shown that intermediary Crowdsourcing approaches fulfil this requirements to a satisfying degree. However there exists no approach that can be combined with an ACMS. Thus, the concept introduced the component Crowdsourcing Engine and Crowdsourcing Adapter which enable transactions with various Crowdsourcing platforms. Further, the combination of these components allows to add and modify Crowdsourcing platforms to a ACMS as needed.

The prototype couples the components Crowdsourcing Engine and Crowdsourcing Adapter to an external framework called Multiplatform Crowdsourcing Framework. The interaction with the Multiplatform Crowdsourcing Framework is realized through an API. This approach allows the use of various and arbitrary, existing and future Crowdsourcing platforms in an ACMS and improves the reusability of the Multiplatform Crowdsourcing Framework for further implementations.

The argumentation indicates that the prototype can fulfil this requirement to a satisfying degree, thus the prototype is rated ●.

Quality

Crowdsourcing enables Knowledge Workers to retrieve various results for a task. Thus, a suitable solution has to provide mechanisms that enable the evaluation of the quality of the results delivered by the crowd, see 2.1.4.

The examinations have shown that dedicated Crowdsourcing approaches as well as intermediary Crowdsourcing approaches can be used to implement appropriate mechanisms. Thus, the concept uses multiple components for this purpose: either the component Crowdsourcing Service or the component Crowdsourcing Adapter provide functions to implement appropriate mechanisms.

The prototype demonstrates the use of Crowdsourcing Services to fulfil the requirement Quality. The Crowdsourcing Task defines a parameter that allows Knowledge Workers to set the quality of results in a three-part scale. The parameter is transformed by the Crowdsourcing Adapter in relation to the scale of a specific Crowdsourcing Service. This approach enables to define parameters of mechanisms which are used to evaluate the quality of results in relation to a specific Crowdsourcing Service, e.g. mechanisms based on the relation of appropriate and inappropriate results delivered by a specific contributor.

The prototype demonstrates the use of mechanisms of the component Crowdsourcing Service to evaluate the quality of the results delivered by the crowd. Thus, the prototype is rated ●.

Applicability

One goal of the thesis is to improve the productivity of Knowledge Workers. Thus, it is important that a solution minimizes unnecessary efforts. This means that Knowledge Workers get to stay in their domain of expertise and that a solution is provided that requires no context switches. Thus, a combination of solutions of the scopes of Knowledge Work and Crowdsourcing is necessary.

The examinations have shown that dedicated Knowledge Work support approaches can fulfil this requirement to a limited degree. The concept introduced the combination of dedicated Knowledge Work support approach, intermediary Crowdsourcing approach and dedicated Crowdsourcing approaches. This enables the integration of Crowdsourcing in the process of Knowledge Work without forcing Knowledge Workers to leave their domain of expertise or to switch context between different applications.

The prototype demonstrates the concept by extending VSRCM through the component Crowdsourcing Task. However, defining a Crowdsourcing Task requires Knowledge Workers to understand the parameters that are needed in context with Crowdsourcing. This requires Knowledge Workers to understand the principle of Crowdsourcing and forces them to leave their domain of expertise. Thus, the prototype is rated ◐.

Summary

Table 5.1 summarizes the evaluation of the elaborated solution. The rating scale is derived from the different requirements defined in chapter 2.1. In general, the scale is divided into the ratings ○, ◐ and ●.

Implementations which are evaluated with the rating ● fulfil a requirement completely. The rating ◐ illustrates that a approach fulfils a requirement to a limited degree. If an approach is rated with ○ the corresponding requirement is not fulfilled.

Requirement	Rating
Knowledge Work	●
Crowdsourcing	●
Multiplatform	●
Quality	●
Applicability	◐

Table 5.1: Summarized evaluation of the proposed solution

The evaluation suggests that the implemented prototype can solve the stated problem in a holistic way. The solution combines the strengths of data-centric Knowledge Work support approaches, intermediary Crowdsourcing approaches, and dedicated Crowdsourcing approaches to fulfil all requirements to a satisfying degree. However, the solution requires Knowledge Workers to understand the principle of Crowdsourcing at a minimum level. Thus, the solution can fulfil the requirement Applicability only to a limited degree.

The following chapter summarizes the thesis and provides future prospects on possible research objects.

6 Summary

This thesis introduced a concept that allows to integrate Crowdsourcing in the process of Knowledge Work, answering the question how Knowledge Work can be enhanced through Crowdsourcing.

The introduction has shown that Knowledge Work is an important part of our modern society and that Crowdsourcing can be used to improve the productivity of Knowledge Workers in various ways. Further, the introduction explained problems that exists in relation to the application of Knowledge Work and Crowdsourcing.

Based on the stated problems, several requirement have been identified. These requirements were used to evaluate promising approaches to the scopes of Knowledge Work and Crowdsourcing. The evaluation has shown that there is no solution that can meet all defined requirements equally well. However, each requirement can be fulfilled by a different group of approaches, i.e. data-centric Knowledge Work approaches, intermediary Crowdsourcing approaches, and dedicated Crowdsourcing approaches.

A concept has been developed, which combines the strengths of these approaches to define a holistic solution for the outlined problem. Thus, the concept describes an architecture that combines several components which meet the requirements.

A system based on the principles of Adaptive Case Management is used to provide Knowledge Workers an environment that is in line with the characteristics of Knowledge Work, i.e. Adaptive Case Management System. The Adaptive Case Management System is extended by the component Crowdsourcing Task and Crowdsourcing Goal which enable Knowledge Workers to use Crowdsourcing for several types of task without forcing them to leave their domain of expertise or to switch between different applications. Further, the concept introduces the component Crowdsourcing Engine, Crowdsourcing Adapter, and Crowdsourcing Service which enable the use of various and arbitray Crowdsourcing platforms and which provide options to add and modify existing and future platforms as needed. To evaluate the quality of results delivered by the crowd, the concept describes appropriate mechanisms in the components Crowdsourcing Adapter and Crowdsourcing Service.

The implementation of a prototype demonstrates the applicability of the concept. Following the concept, the Adaptive Case Management System VSRCM is extended by the component Crowdsourcing Task, and combined with the components Crowdsourcing Engine, Crowdsourcing Adapter and Crowdsourcing Service in an applicable way. Further, the documentation of the prototype discusses general aspects of the implementation in context of the concept as well as aspects in regard of the implemented prototype, e.g. the implementation of a Multiplatform Crowdsourcing Framework.

Based on the defined requirements, the evaluation of the prototype has shown that the concept provides a solution which meets all requirements to a satisfying degree.

While the concept provides a solution to the outlined problem, the thesis has not considered other topics in context of Crowdsourcing, e.g. motivation of workers, Cost-Efficient Crowdsourcing, Real-Time Crowdsourcing or Reactive Crowdsourcing [6, 38, 22, 8]. These topics overextend the scope of this thesis and can be used to implement additional features of the elaborated concept.

In context of the thesis there is potential for future research objects.

As shown in the concept it is necessary to transform transactions of different Crowdsourcing platforms. To the best of my knowledge there is no approach which implements the standardization of these transactions. Thus, it is reasonable to research an appropriate solution. Possible candidate for this purpose are Human Tasks [41] as used in [48].

The thesis has shown that Knowledge Workers benefit by employing human intelligence for specific tasks. However, some tasks can be solved efficiently by artificial intelligence. The combination of artificial intelligence and human intelligence promises a powerful tool, implementing a hybrid crowd-machine solution, that allows Knowledge Workers to get assistance in a wide field of applications. [11, 34] Thus, analyzing how these approaches can be combined to an holistic solution is worthy of additional research.

Bibliography

[1] Aalst, WilM.P. van der, ArthurH.M. ter Hofstede and Mathias Weske: *Business Process Management: A Survey.* In Aalst, WilM.P. van der and Mathias Weske (editors): *Business Process Management*, volume 2678 of *Lecture Notes in Computer Science*, pages 1–12. Springer Berlin Heidelberg, 2003.

[2] al., Zhu et: *Advanced case management with IBM Case Manager.* IMB Corp., International Technical Support Organization, Poughkeepsie, NY, 2014.

[3] Alliance, The Citizen Science: *Galaxy Zoo*, 2014. [Online; accessed 8-October-2014].

[4] Amazon: *Amazon Mechanical Turk*, 2014. [Online; accessed 13-October-2014].

[5] Bell, Daniel: *The coming of post-industrial society : a venture in social forecasting.* Basic Books, New York, 1976.

[6] Bernstein, Michael S., David R. Karger, Robert C. Miller and Joel Brandt: *Analytic Methods for Optimizing Realtime Crowdsourcing.* CoRR, abs/1204.2995, 2012.

[7] Bozzon, Alessandro, Marco Brambilla and Stefano Ceri: *Answering Search Queries with CrowdSearcher.* In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 1009–1018, New York, NY, USA, 2012. ACM.

[8] Bozzon, Alessandro, Marco Brambilla, Stefano Ceri and Andrea Mauri: *Reactive crowdsourcing.* In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 153–164, 2013.

[9] Brabham, Daren C.: *Crowdsourcing as a Model for Problem Solving.* Convergence: The International Journal of Research into New Media Technologies, 14(1):75–90, February 2008.

[10] Campbell, Donald J: *Task complexity: A review and analysis.* Academy of management review, 13(1):40–52, 1988.

[11] Cheng, Justin and Michael S. Bernstein: *Flock: Hybrid Crowd-Machine Learning Classifiers.* 2012.

[12] CrowdFlower: *CrowdFlower*, 2014. [Online; accessed 13-October-2014].

[13] CrowdFlower: *CrowdFlower - API Documentation (API Version 2014-07-31)*, 2014. [Online; accessed 20-November-2014].

[14] crowdsourcing.org: *Directory of Sites*, 2014. [Online; accessed 20-November-2014].

[15] Crowdspring: *Crowdspring*, 2014. [Online; accessed 13-October-2014].

[16] Davenport, T.H.: *Thinking for a Living: How to Get Better Performances And Results from Knowledge Workers.* Harvard Business Review Press, 2013.

[17] Dawson, R. and S. Bynghall: *Getting Results from Crowds: Second Edition: The Definitive Guide to Using Crowdsourcing to Grow Your Business.* Advanced Human Technologies, Incorporated, 2012.

[18] Demartini, Gianluca, Beth Trushkowsky, Tim Kraska and Michael J. Franklin: *CrowdQ: Crowdsourced Query Understanding.* In *CIDR*, 2013.

[19] Drucker, Peter: *Landmarks of tomorrow a report on the new 'post-modern' world.* Transaction Publishers, New Brunswick, N.J., U.S.A, 1996.

[20] Drucker, P.F.: *The Landmarks of Tomorrow.* Harper & Row, 1959.

[21] Drucker, P.F.: *Post-capitalist society.* HarperBusiness, 1993.

[22] Eickhoff, Carsten, Christopher G Harris, Arjen P de Vries and Padmini Srinivasan: *Quality through flow and immersion: gamifying crowdsourced relevance assessments.* In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 871–880. ACM, 2012.

[23] Estellés-Arolas, Enrique and Fernando González-Ladrón-De-Guevara: *Towards an Integrated Crowdsourcing Definition.* J. Inf. Sci., 38(2):189–200, April 2012.

[24] Franklin, Michael J., Donald Kossmann, Tim Kraska, Sukriti Ramesh and Reynold Xin: *CrowdDB: Answering Queries with Crowdsourcing.* In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, SIGMOD '11, pages 61–72, New York, NY, USA, 2011.

ACM.

[25] Google: *ReCaptcha*, 2014. [Online; accessed 8-October-2014].

[26] Grossman, Gene M. and Elhanan Helpman: *Outsourcing in a Global Economy.* The Review of Economic Studies, 72(1):135–159, 2005.

[27] Group, Object Management: *Business Process Model and Notation*, 2014. [Online; accessed 13-October-2014].

[28] Heil, Sebastian, Stefan Wild and Martin Gaedke: *Collaborative Adaptive Case Management with Linked Data.* In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, WWW Companion '14, pages 99–102, Republic and Canton of Geneva, Switzerland, 2014. International World Wide Web Conferences Steering Committee.

[29] Hetmank, Lars: *A Synopsis of Enterprise Crowdsourcing literature.* In *22st European Conference on Information Systems, ECIS 2014, Tel Aviv, Israel, June 9-11, 2014*, 2014.

[30] Howe, Jeff: *The Rise of Crowdsourcing.* (accessed July 20, 2010), June 2006.

[31] Hube, Gerhard: *Beitrag zur Beschreibung und Analyse von Wissensarbeit.* Jost-Jetter, Heimsheim, 2005.

[32] InnoCentive: *InnoCentive*, 2014. [Online; accessed 13-October-2014].

[33] Jeston, John and Johan Nelis: *Business process management.* Routledge, 2014.

[34] Kamar, Ece, Severin Hacker and Eric Horvitz: *Combining Human and Machine Intelligence in Large-scale Crowdsourcing.* In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, pages 467–474, Richland, SC, 2012. International Foundation for Autonomous Agents and Multiagent Systems.

[35] Kittur, Aniket, Boris Smus, Susheel Khamkar and Robert E. Kraut: *CrowdForge: Crowdsourcing Complex Work.* In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 43–52, New York, NY, USA, 2011. ACM.

[36] La Vecchia, Gioacchino and Antonio Cisternino: *Collaborative Workforce, Business Process Crowdsourcing as an Alternative of BPO.* In Daniel, Florian and FedericoMichele Facca (editors): *Current Trends in Web Engineering*, volume 6385 of *Lecture Notes in Computer Science*, pages 425–

430. Springer Berlin Heidelberg, 2010.

[37] Machlup, Fritz: *The production and distribution of knowledge in the United States.* Princeton University Press, Princeton, N.J, 1962.

[38] Mashhadi, Afra J. and Licia Capra: *Quality Control for Real-time Ubiquitous Crowdsourcing.* In *Proceedings of the 2Nd International Workshop on Ubiquitous Crowdsouring*, UbiCrowd '11, pages 5–8, New York, NY, USA, 2011. ACM.

[39] Matthias, John T.: *Technology for Case Management.* In *Mastering the unpredictable : how adaptive case management will revolutionize the way that knowledge workers get things done*, pages 63–89. Meghan-Kiffer Press, 2010.

[40] Mundbrod, Nicolas, Jens Kolb and Manfred Reichert: *Towards a System Support of Collaborative Knowledge Work.* In Rosa, Marcello La and Pnina Soffer (editors): *Business Process Management Workshops*, volume 132 of *Lecture Notes in Business Information Processing*, pages 31–42. Springer, 2012.

[41] OASIS: *Web Services - Human Task; (WS-HumanTask) Specification Version 1.1*, 2014. [Online; accessed 20-November-2014].

[42] OECD: *OECD Science, Technology and Industry Scoreboard 2013.*

[43] O'reilly, Tim: *What is web 2.0.* O'Reilly Media, Inc., 2009.

[44] Osterhage, Wolfgang: *ERP-Kompendium Eine Evaluierung von Enterprise Resource Planning Systemen.* Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[45] Pont, B. and P. Werquin: *Competencies for the Knowledge Economy.* Education Policy Analysis, 2001.

[46] Pucher, Max J.: *The Elements of Adaptive Case Management.* In *Mastering the unpredictable : how adaptive case management will revolutionize the way that knowledge workers get things done*, pages 89–135. Meghan-Kiffer Press, 2010.

[47] Riggert, Wolfgang: *ECM - Enterprise Content Management Konzepte und Techniken rund um Dokumente.* Vieweg+Teubner Verlag / GWV Fachverlage GmbH, Wiesbaden, Wiesbaden, 2009.

[48] Schall, Daniel: *Service-Oriented Crowdsourcing Architecture, Protocols and Algorithms.* Springer New York Imprint Springer, New York, NY, 2012.

[49] Schenk, Eric and Claude Guittard: *Towards a characterization of crowdsourcing practices.* Journal of Innovation Economics & Management, (1):93–107, 2011.

[50] Surowiecki, James: *The wisdom of crowds : why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations.* Doubleday, New York, 2004.

[51] Swenson, Keith: *Mastering the unpredictable : how adaptive case management will revolutionize the way that knowledge workers get things done.* Meghan-Kiffer Press, 2010.

[52] Swenson, Keith: *Case Management: Contrasting Production vs. Adaptive*, 2014. [Online; accessed 13-October-2014].

[53] Swenson, KeithD.: *Position: BPMN Is Incompatible with ACM.* In La Rosa, Marcello and Pnina Soffer (editors): *Business Process Management Workshops*, volume 132 of *Lecture Notes in Business Information Processing*, pages 55–58. Springer Berlin Heidelberg, 2013.

[54] Troia, Project: *Troia*, 2014. [Online; accessed 20-November-2014].

[55] Turk, Amazon Mechanical: *Amazon Mechanical Turk - API Reference (API Version 2014-08-15)*, 2014. [Online; accessed 20-November-2014].

[56] Wikipedia: *Clickworkers*, 2014. [Online; accessed 8-October-2014].

[57] Zaidan, Omar F. and Chris Callison-Burch: *Crowdsourcing Translation: Professional Quality from Non-professionals.* In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1220–1229, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.