

# Evaluierung einer Experience Factory bei der COMPRA GmbH

Lennard Brunke

October 29, 2022

## Zusammenfassung

In der vorliegenden Projektarbeit soll evaluiert werden, ob die DISER Methodologie geeignet ist, um bei der COMPRA GmbH eine neue Experience Factory zu entwickeln. Dieses System soll im Bereich der Softwareentwicklung des ERP Systems eEvolution und anderer Projekte der COMPRA GmbH Entscheidungsunterstützung bieten. (Tautz, 2001) Es wird die aktuelle Situation des Experience-Managements bei der COMPRA GmbH analysiert und im weiteren behandelt werden, wie die DISER Methodologie grundsätzlich aufgebaut ist und wie sie auf die Anforderungen der COMPRA GmbH sowie eine moderne agile Arbeitsweise angepasst werden kann. Das Ziel dieser und weiterführender Arbeit ist es, eine neue Experience Base bei der COMPRA GmbH aufzubauen.

## Einleitung

Warum ist Erfahrungsmanagement für Unternehmen, insbesondere im Bereich der Softwareentwicklung, so wichtig? Reicht es aus, dass Wissen und Erfahrung im persönlichen Gespräch von Mitarbeiter zu Mitarbeiter weitergegeben oder sie sporadisch und in unterschiedlichsten Formen in einem Netzwerk, in einer Quellcodeverwaltung oder einem DMS abgelegt werden?

Wie kann sichergestellt werden, dass das Wissen des eines langjährigen Mitarbeiters nicht gänzlich verloren geht, wenn dieser das Unternehmen verlässt? Diese einleitenden Fragen geben einen guten ersten Eindruck, warum Erfahrungsmanagement im Bereich Softwareentwicklung eine zentrale Rolle für die Produktivität der Mitarbeiter und die Qualität der entwickelten Produkte spielt.

(Tautz, 2001) beschreibt in der Einleitung, dass der größte Vorteil im Wettbewerb von Unternehmen in der Softwareentwicklung darin besteht, bessere Software mit mehr Features schneller und mit geringeren Kosten entwickeln zu können. Dieses Know-How kann durch die Konkurrenz nicht imitiert werden, da es nicht von dem Softwareprodukt ableitbar ist. Daraus lässt sich annehmen, dass Experience Management im Software Engineering ein entscheidender Vorteil ist, um sich von der Konkurrenz abzusetzen.

## Experience Factory

In diesem Kapitel soll das Konzept von QIP und der Experience Factory erläutert werden.

Eine Experience Factory (Althoff and Birk, 1997) (Tautz, 2001) beschreibt eine Einheit in einem Softwareunternehmen, welche kontinuierliche und inkrementelle Verbesserung der Softwareentwicklung im Unternehmen erzeugen soll. Dies wird erreicht durch die Verarbeitung und Analyse aller Arten von Wissen und Erfahrung, welche im Kontext der Softwareentwicklung relevant sind. Sie dient als ein Repository (Experience Base) für dieses Wissen und entspricht einer Fallbasis im Sinne von Case-Based-Reasoning. Weiterhin ist die Experience Factory zuständig für die Administration der Experience Base.

In Kapitel 1.1 beschreibt Tautz das Konzept der Experience Factory durch einen Vergleich des CBR Zyklus mit dem Paradigma QIP.

Quality Improvement Paradigm [Basili001] ist ein früherer Ansatz für iterative Qualitätsverbesserung von Softwareprojekten. QIP zielt darauf ab, schon in der Startphase eines Softwareprojektes festzulegen, welche Erfahrungen daraus gewonnen werden können und wie dies konkret erreicht werden kann. QIP ist als eine Feedback Loop aufgebaut, welche durch die selbst auferlegten Ziele gesteuert wird. Im letzten Schritt von QIP wird die gewonnenen Erfahrungen in eine wieder verwendbare Form gebracht und aufbewahrt. Im Kontrast dazu ist der CBR Zyklus mehr technischer Natur, aber grundsätzlich zielen beide Ansätze auf das Gleiche ab: Ein Experience Model welches im Kontext der Softwareentwicklung wiederverwendet werden kann.

Beide dieser Konzepte bilden die Basis der Experience Factory: > The experience factory is a logical and/or physical organization that > supports project developments by analyzing and synthesizing all > kinds of experience, acting as a repository for such experience, and > supplying that experience to various projects on demand. It packages experience by building > informal,

formal or schematized, and > productized models and measures of various software processes, > products, and other forms of knowledge via people, documents, > and automated support.

Die genannten Konzepte wurden in einer Zeit entwickelt, in welcher das übliche Vorgehen in Softwareprojekten einer Form des Wasserfallmodells entsprach. Heutzutage, und auch bei der COMPRA GmbH, wird ein agiler Entwicklungsprozess bevorzugt. Der Fakt, dass CBR und QIP schon damals einer iterativen Funktionsweise zugrunde lagen zeigt, dass ein agiler, iterativer Entwicklungsansatz diese Arten von Prozessen noch viel besser einsetzen und einfacher Wert daraus gewinnen kann. Im Vergleich zum Gebrauch in einem Wasserfallmodell werden die Erfahrungsinkremente bei einem agilen Prozess kleiner und fokussierter sein.

Dies lässt grundsätzlich vermuten, dass der Aufbau und letztendlich auch der laufende Betrieb einer Experience Factory sich gut in den aktuellen agilen Entwicklungsprozess der COMPRA GmbH integrieren lässt.

## **Erfahrungs-Repositories**

In diesem Kapitel soll ein Überblick über die verschiedenen Ansätze, welche im Erfahrungsmanagement in der Softwareentwicklung genutzt werden, gegeben werden. Außerdem sollten hier die Techniken beschrieben werden, die bisher bei der COMPRA GmbH für das Thema Erfahrungsmanagement verwendet werden.

Bei der COMPRA GmbH werden verschiedene Softwaresysteme geführt, welche neben ihrer hauptsächlichen Funktion auch als Repository für Wissens- und Erfahrungsartefakte verwendet werden. Dies geschieht nicht zwangsläufig mit der Intention des Erfahrungsmanagement, sondern ist häufig eine Konsequenz aus Kommunikation und Kollaboration. Nachfolgend werden alle Softwaresysteme genannt, welche in irgendeiner Form als Repository für Erfahrungsmanagement genutzt werden:

- MS Team Foundation Server:  
Source Control Interne sowie externe Kollaborationsplattform zur Planung, Kommunikation und Dokumentation der Softwareentwicklung in allen Teams und Projekten Projekt- oder Teambezogene Wikis
- ELO Document Management System

Interne Dokumente (Personalmanagement, Gehaltsabrechnungen etc.)  
Dokumentation von internen Prozessen und Vorgängen (Sprint Retrospektiven, Protokolle)

- Windows File-Server:  
Ablage von verschiedensten Dateien und Dokumenten Installationsdateien Projektbezogene Dateien und Dokumente
- ERP System eEvolution und MS SQL Server:  
Erstellung von Rechnungen Dokumentation von Arbeitszeiten Verwaltung von Mitarbeitern und Kunden CRM
- [eEvolution Wiki](#)  
Offizielles, zentrales Wiki für das ERP System eEvolution
- MS Teams:  
Interne und externe Kommunikation und Kollaboration (Chats, Meetings).
- MS Sharepoint:  
Teamübergreifende Kalender, Dokumentation, Verschiedenes.
- Yammer:  
Internes soziales Netzwerk
- Email und IP-Telefon

Vergleichen wir diese Systeme mit den von Tautz aufgeführten vorhandenen Ansätzen für Erfahrungs-Repositories wird deutlich, dass eine ganze Reihe bei der COMPRA GmbH im Einsatz sind:

- Controlled / Uncontrolled Keyword System [Tautz001 5.1.4]
- Full Text Search auf vorhandene Repositories [Tautz001 5.1.5]
- Hypertext [Tautz001 5.2]
- Relational and Object-Oriented DBMS [Tautz001 5.3.1 5.3.2]

Alle vorhandenen Repositories verfügen über die Möglichkeit über Full Text Search durchsucht zu werden. Der Team Foundation Server verwendet ein unkontrolliertes Keyword System, wobei jegliche Art von Workitem mit beliebigen Keywords versehen werden kann. Das eEvolution Wiki ist eine

Website, verwendet also Hypertext. eEvolution ist ein ERP System, welches mit einer relationalen Datenbank betrieben wird.

Auffallend ist hier, dass nur eher simple Ansätze verwendet werden, welche ohne weiteren größeren Aufwand innerhalb der genannten Softwaresysteme betrieben werden können. Es gibt kein System, welches logikbasiert, wissensbasiert oder mit irgendeiner Form von KI arbeitet.

## **Erfahrungsmanagement Ist-Situation**

Zuvor wurden Erfahrungs-Repositories bei der COMPRA GmbH genannt. In diesem Kapitel soll im Detail behandelt werden, wie die Repositories im Prozess der Softwareentwicklung eingebunden sind, welche Vor- und Nachteile dies beinhaltet und welche Maßnahmen nötig sind, um die Softwareentwicklung durch gezieltes Erfahrungsmanagement zu verbessern.

Das am häufigsten und am erfolgreichsten verwendete System bei der COMPRA GmbH ist der MS Team Foundation Server (TFS). Er wird als Source Control, Product Backlog sowie Kommunikations- und Kollaborationsplattform (intern und extern) verwendet. Die COMPRA GmbH arbeitet nach einem agilen Manifest mit Scrum ([Schwaber and Sutherland, n.d.](#)). Im TFS wird das Product Backlog zu jedem Team bzw. Kunden oder Projekt in Form von Features, User Stories und Issues gepflegt. An diesem Product Backlog sind alle relevanten Stakeholder beteiligt.

Nehmen wir nun an, ein Team steht vor einer neuen Softwareanforderung und möchte frühere Entwicklungen und Erfahrungen nutzen, welche bereits für andere Projekte in diesem Bereich gemacht wurden, um den Entwicklungsprozess in den Aspekten Kosten, Geschwindigkeit und Qualität zu verbessern.

Das bisherige übliche Vorgehen ist hierbei, zunächst den TFS über eine Full Text Search, Keywords oder gefiltert nach Attributen des Product Backlogs zu durchsuchen. Dies kann in manchen Fällen schnell relevante Product Backlog Items finden, jedoch nur, wenn der Suchende bereits Kenntnis über die Existenz der relevanten Backlog Items hat. Ist keine Kenntnis darüber vorhanden, ist die Suche meist langwierig und ohne Erfolg. Aus diesem Grund wird vor dem Schritt des Durchsuchens des Backlogs im persönlichen Gespräch mit anderen Mitarbeitern versucht, bereits zu filtern ob ähnliche Anforderungen in der Vergangenheit bereits existierten, für welches Projekt bzw. Kunden diese entwickelt wurden und welche Mitarbeiter darin involviert waren. Mit diesen Informationen können die richtigen Kollegen angesprochen werden, um letztendlich die relevanten Backlog Items zu finden, sofern sie existieren. Es ist vorgesehen, dass Backlog Items mit sämtlichen

relevanten Informationen verknüpft sind. Dies beinhaltet Code Changesets, Dokumentationen jeglicher Art sowie sämtliche Kommunikation zum Backlog Item im Bezug auf Requirement-Engineering. In der Praxis ist dies leider nicht immer der Fall und auch hier sind wichtige Informationen oft nur über die involvierten Mitarbeiter zu finden. Ein Problem dabei ist, dass wie oben beschrieben noch viele weitere Systeme als Experience-Repository verwendet werden und nicht alle Informationen im relevanten TFS Backlog Item gesammelt werden. Gibt es beispielsweise eine Installationsanleitung mit “lessons learned” aus vorherigen Projekten, welche aber in einem Projektdossier auf einem File Server abgelegt ist, so muss der Mitarbeiter erst auf diese Information hingewiesen werden.

Dies führt zu einer großen Belastung von gewissen Senior Mitarbeitern, welche den größten Erfahrungsschatz besitzen. Das Problem und die Auslastung wird noch viel größer, wenn der Mitarbeiter wiederholt aus verschiedenen Quellen nach den selben Informationen gefragt wird.

Um dieses Problem zu lösen, sollte ein einheitlicher Prozess für eine Experience Factory entwickelt werden, damit Artefakte in genau einem Repository zu finden sind und Mitarbeiter wissen, wie diese Artefakte dort abgelegt werden sollen. Weiterhin muss die Experience Factory in die Entwicklungsprozesse integriert werden, damit die Experience Base kontinuierlich wächst und verbessert wird. Wenn ein Mitarbeiter nach einer Information sucht muss klar sein wie und wo diese Information zu finden ist.

## **Agiler Entwicklungsprozess bei der COMPRA GmbH**

In diesem Kapitel wird der agile Entwicklungsprozess bei der COMPRA GmbH aufgezeigt und weiterhin ermittelt, wie der Betrieb einer Experience Factory in diesen integriert werden kann.

Die COMPRA GmbH setzt für die Komposition der Teams auf Full-Stack Teams. Das bedeutet, dass die einzelnen Teams zwar auf gewisse Bereiche spezialisiert sind, weil in diesen Bereichen einfach mehr Erfahrung vorhanden ist, grundsätzlich aber ein einzelnes Team alle Arten von Aufgaben oder Prozessen durchführen kann, welche in der Softwareentwicklung der COMPRA GmbH benötigt werden, also von der Akquirierung eines Projektes über Requirements-Engineering bis zum fertigen Softwareinkrement. Dabei arbeiten die Teams nach einem agilen Manifest mit Scrum (**Schwaber and Sutherland, n.d.**), wobei hier bei allen Teams gewisse Details unterschiedlich gehandhabt werden, der Kernprozess aber der Gleiche ist.

Als Beispiel betrachten wir das Allstars Team mit dem Projekt Rossmann Central Europe. Das Allstars Team entwickelt und betreut den Betrieb einer

kundenspezifische Version des ERP Systems eEvolution 10 für Rossmann Ungarn und Rossmann Tschechien. Das Team besteht aus einem Scrum Master aus einem anderen Team bei der COMPRA GmbH, einem Product Owner von Rossmann Ungarn, einem Product Owner von Rossmann Tschechien und 5 Entwicklern sowie einem Auszubildenden. Das Product Backlog im Team Foundation Server ist eine Liste von Features, User Stories und Issues und stellt die zentrale Quelle an Arbeit für das Team dar. Ein Feature ist hier gleichzusetzen mit einem Product Goal und beschreibt den Zielzustand eines Teiles des Produktes. Um diesen Zielzustand zu erreichen werden User Stories unterhalb eines Features definiert, welche umgesetzt werden müssen um das Ziel des Features zu erreichen. Dabei sollten User Stories im Umfang idealerweise möglichst klein sein, damit sie innerhalb eines Sprints abgeschlossen werden können und der Aufwand für dieses Workitem besser abschätzbar ist. Issues sind für ungeplante Arbeit, also Probleme welche während eines Sprints auftauchen und schnell gelöst werden müssen. Dies kann alle Arten von Problemen beinhalten welche durch das Scrum Team gelöst werden. (Bugs in laufender Software, Support für User etc.) Sprints werden grundsätzlich für eine Dauer von zwei Wochen geplant. Zum Ende eines Sprints werden sowohl ein Sprint Review, eine Retrospektive als auch ein Planning für den nächsten Sprint durchgeführt.

Ausgehend von der oben beschriebenen Situation kann festgestellt werden, wie eine Experience Factory in den Prozess integriert werden kann. Der erste Schritt zur Integration sollte sein, die Definition of Done eines Features, einer User Story oder eines Issues darum zu erweitern, das Wissen und die Erfahrung aus diesem Workitem in den Prozess der Experience Factory einfließen zu lassen. Weiterhin gilt es zu entscheiden, welche Mitarbeiter aktiv für den Aufbau und den Betrieb der Experience Factory zuständig sind und wie sich diese als Einheit oder Team in den Rest des Unternehmens einbindet.

In (Tautz, 2001) und (Althoff and Birk, 1997) wird die Experience Factory eher als eine getrennte Organisationsstruktur beschrieben (siehe Abbildung 1).

Für die COMPRA GmbH als agiles Unternehmen mit Full-Stack Teams käme es nicht in Frage, Experience Engineers in einer gesonderten Organisationsstruktur oder einem eigenen Team zu beschäftigen. Stattdessen sollte es pro Team einige wenige Mitarbeiter geben, welche als zusätzliche Aufgabe aktiv für den Prozess der Experience Factory verantwortlich sind. Im Sinne von Agile würde dies im Unternehmen eher als eine Guild oder ein Chapter beschrieben werden. Eine Gilde ist eine Gruppe von Mitarbeitern aus verschiedenen Teams, welche gemeinsam an einem gewissen Thema oder einer

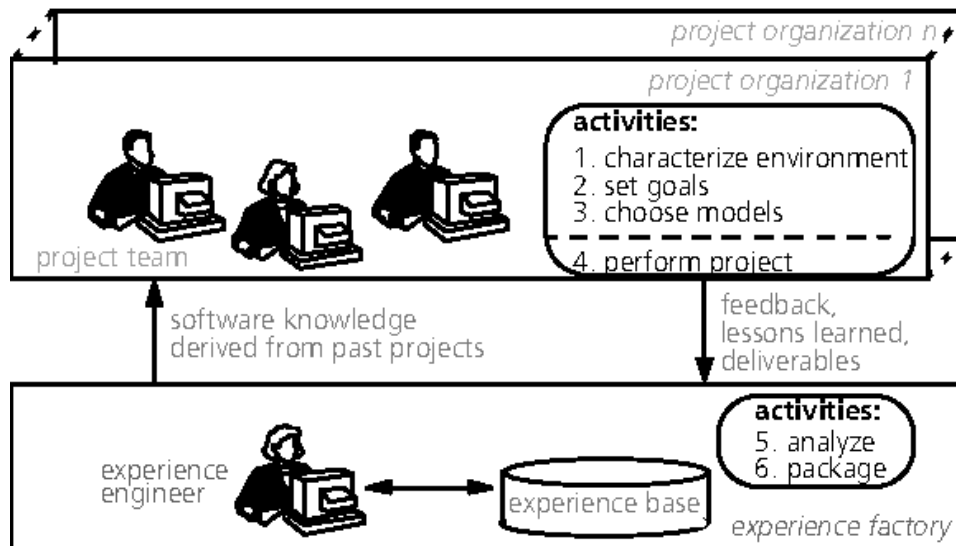


Figure 1: Experience Factory

Kompetenz arbeiten und sich regelmäßig dazu austauschen. Dieses Konzept von Guilds, Chapters und Tribes wurde erstmalig von Spotify eingesetzt, um Agile in großen Unternehmen besser skalierbar zu machen. (Wardt, 2022) Ziel einer Gilde ist es, Erfahrung auszutauschen und das Unternehmen als ganzes zu verbessern. Dies deckt sich mit den Zielen der Experience Factory.

Die einzelnen Mitglieder der Gilde sind also dafür verantwortlich die Erfahrung aus der Arbeit des eigenen Teams in die Experience Factory zu bringen, um dann gemeinsam mit der Gilde diese Artefakte zu analysieren und zu verpacken, damit das Wissen im Unternehmen wieder verwendbar ist.

## DISER

In diesem Kapitel soll die Methodologie DISER zusammengefasst werden.

### Repräsentation von Software Engineering Erfahrungen

In diesem Kapitel soll (Tautz, 2001) Kapitel 6 zusammengefasst werden.

Eine der elementarsten Fragen für den Aufbau einer neuen Experience



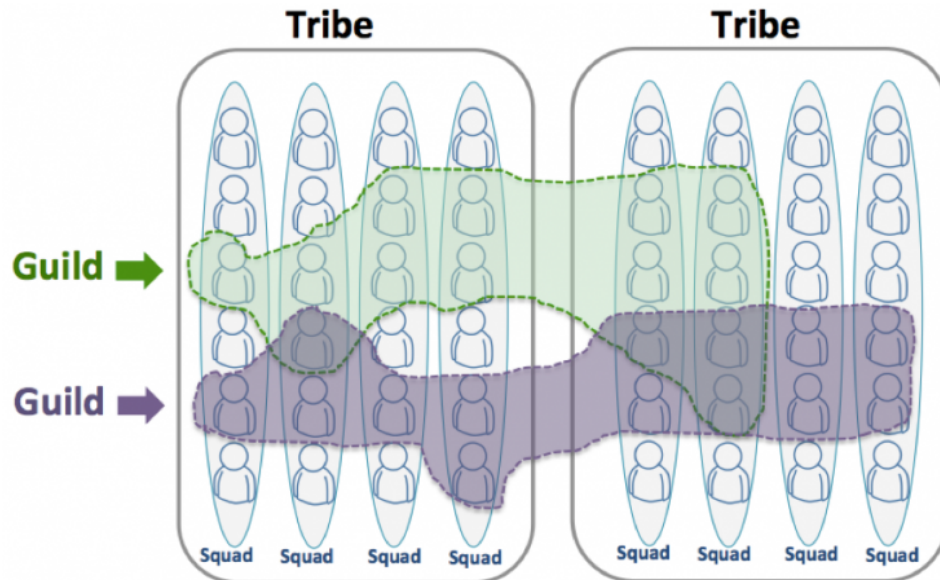


Figure 2: Agile und Scrum bei Spotify

Base ist, nach welchem Schema das Wissen und die Erfahrung strukturiert sein soll. Dies ist wichtig, um einerseits das Erfassen von Artefakten einfach und intuitiv zu halten und sie in der alltäglichen Kommunikation verwenden zu können, aber auch um die spezifischen Anforderungen des Unternehmens abzudecken. Weiterhin bestimmt der Repräsentationsformalismus, wie die Erfahrungsartefakte tatsächlich im Kontext der Softwareentwicklung genutzt werden können.

## REFSENO

REFSENO (Tautz and Gresse von Wangenheim, 1998) beschreibt wie die Informationen in einer Experience Base formuliert und strukturiert werden können um die Möglichkeiten in der Verwendung in einer Experience Factory zu maximieren. [Tautz001 6.1] REFSENO's zentrales Element wird durch "concepts" definiert, welche Software Engineering Artefakte modellieren. Concepts setzen sich durch verschiedene Attribute zusammen, welche sich in terminal und nonterminal attributes unterscheiden.

**GOODSEE (goal-oriented ontology development for software engineering experience)**

**Technische Infrastruktur**

**DISER Task Framework**

**DISER Anforderungen**

## **Anforderungen COMPRA**

In diesem Kapitel soll beschrieben werden, welche Anforderungen von COMPRA kommen und ob DISER diese erfüllen kann. Hier sollten die Anforderungen von Tautz durchgegangen und evaluiert werden, welche für die COMPRA relevant sind. Weiterhin wird beschrieben, was angepasst werden muss.

**Wissensquellen**

**Fazit**

## **Relevante Kapitel aus [Tau001]**

T32 LEARN Decomposition: record (T33, page 107), analyze SEEMS (T50, page 116), forget (T54, page 119), package (T55, page 120)

## **Literaturverzeichnis**

Althoff, K.-D. and Birk, A., 1997. The experience factory approach: Realizing learning from experience in software development organizations.

Schwaber, K. and Sutherland, J., n.d. *The 2020 scrum GUIDE. Scrum Guide / Scrum Guides*. Available at: <<https://scrumguides.org/scrum-guide.html>>.

Tautz, C., 2001. *Customizing software engineering experience management systems to organizational needs*. PhD thesis.

Tautz, C. and Gresse von Wangenheim, C., 1998. *REFSENO: A representation formalism for software engineering ontologies*.

Wardt, R. van der, 2022. *Das spotify modell: Agile und scrum für große organisationen*. Agile Scrum Group. Available at: <<https://agilescrumgroup.de/spotify-modell/>>.