

Konzeption und Evaluation eines domänenbasierten Explorers als VS- Code Extension für SAP CAP Anwendungen

Projektskizze Bachelorarbeit

aus dem Studiengang Wirtschaftsinformatik Software Engineering

an der Dualen Hochschule Baden-Württemberg Mannheim

von

Max Christian Meinel

25.02.2026

| | |
|---------------------------------|--------------------------|
| Matrikelnummer, Kurs: | 7864687, WWI23SEB |
| Unternehmen: | SAP SE, 69190, Walldorf |
| Abteilung: | CAP Tools & MTX |
| Leiter des Studiengangs: | Prof. Dr. Henning Pagnia |
| Betreuer im Unternehmen: | Christian Fuchs |
| Betreuer an der DHBW: | Ulrich Wolf |

Selbstständigkeitserklärung

Gemäß Ziffer 1.1.13 der Anlage 1 zu §§ 3, 4 und 5 der Studien- und Prprüfungsordnung für die Bachelorstudiengänge im Studienbereich Technik der Dualen Hochschule Baden-Württemberg vom 29.09.2017. Ich versichere hiermit, dass ich meine Arbeit mit dem Thema:

Konzeption und Evaluation eines domänenbasierten Explorers als VS-Code Extension für SAP CAP Anwendungen

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich versichere zudem, dass alle eingereichten Fassungen übereinstimmen.

Walldorf 25.02.2026

Max Christian Meinel

Abstract

Abstract hier

Inhaltsverzeichnis

| | |
|---|----------|
| 1 Einleitung | 1 |
| 1.1 Motivation | 1 |
| 1.2 Problemstellung | 1 |
| 1.3 Zielsetzung der Arbeit | 1 |
| 1.4 Forschungsfragen | 1 |
| 1.5 Aufbau der Arbeit | 1 |
| 2 Grundlagen | 2 |
| 2.1 SAP Cloud Application Programming Model (CAP) | 2 |
| 2.2 Language Server und Language Server Protocol | 2 |
| 2.3 VS Code Extension Architektur | 2 |
| 3 Methodik | 3 |
| 3.1 Design Science Research | 3 |
| 3.2 Einordnung der Arbeit in den DSR-Prozess | 3 |
| Literatur | a |

1 Einleitung

1.1 Motivation

1.2 Problemstellung

1.3 Zielsetzung der Arbeit

1.4 Forschungsfragen

1.5 Aufbau der Arbeit

2 Grundlagen

2.1 SAP Cloud Application Programming Model (CAP)

2.1.1 CDS und CDL

2.1.2 Core Schema Notation (CSN)

2.2 Language Server und Language Server Protocol

2.2.1 Architektur des Language Servers

2.2.2 Relevante LSP-Funktionalitäten

2.3 VS Code Extension Architektur

2.3.1 Tree View API

2.3.2 Integration von Language Servern

3 Methodik

3.1 Design Science Research

3.2 Einordnung der Arbeit in den DSR-Prozess

3.2.1 Problemidentifikation

3.2.2 Zieldefinition

3.2.3 Artefaktentwurf

3.2.4 Demonstration

3.2.5 Evaluation

Für diese Arbeit wird der Design-Science-Research-Ansatz (DSR) als methodische Grundlage verwendet. Dieser Ansatz eignet sich insbesondere für Arbeiten, bei denen ein konkretes Artefakt entwickelt wird, das ein praktisches Problem adressiert. In dieser Arbeit stellt die entwickelte VS-Code Extension sowie das zugrunde liegende Strukturmodell (domänenbasierter Explorer) dieses Artefakt dar.

Der Design-Science-Research-Ansatz gliedert sich in mehrere Phasen. Zunächst wird das zugrundeliegende Problem präzise analysiert und abgegrenzt. Dabei wird untersucht, welche strukturellen Herausforderungen in größeren SAP CAP Projekten entstehen und weshalb bestehende Werkzeuge nur eingeschränkt eine domänenorientierte Übersicht ermöglichen.

Auf Basis dieser Problemdefinition werden funktionale sowie nicht-funktionale Anforderungen an die Erweiterung definiert. Dazu gehören unter anderem Anforderungen an die Strukturierung der Services, die Filter- und Sortiermöglichkeiten sowie Aspekte der Usability.

Im Anschluss erfolgt die Konzeption des Artefakts. Dabei werden unterschiedliche Modellierungsansätze zur Strukturierung des Repositories betrachtet. Insbesondere werden das Core Schema Notation (CSN) als statische Strukturquelle sowie der Language Server als Quelle für positions- und dateibezogene Informationen analysiert. Ziel ist es, ein geeignetes Modell zu entwerfen, das beide Informationsquellen sinnvoll kombiniert.

Parallel zur konzeptionellen Modellierung werden verschiedene grafische Mock-ups erstellt. Diese dienen dazu, unterschiedliche Ansätze der Visualisierung und Interaktion zu vergleichen. Für den Vergleich werden zuvor definierte Kriterien herangezogen, beispielsweise Übersichtlichkeit, Nachvollziehbarkeit der Domänenstruktur und Erweiterbarkeit.

Auf Grundlage des ausgewählten Konzepts wird anschließend ein prototypischer Explorer als VS-Code Extension implementiert. Die Implementierung dient dabei nicht primär als Produktivlösung, sondern als Demonstrator des entworfenen Modells.

Abschließend erfolgt eine Evaluation des entwickelten Artefakts. Dabei werden sowohl die strukturelle Modellierung (Kombination von CSN und Language Server) als auch die gewählte grafische Darstellung anhand der zuvor definierten Kriterien bewertet. Ziel der Evaluation ist es, die Eignung des entwickelten Konzepts zur Unterstützung von Entwicklern in größeren CAP-Projekten zu überprüfen.

Literatur