

# Aufgabenstellung für «C++ Concept Refactorings»

## Jeremy Stucki / Vina Zahnd

---

### 1. Betreuer

Diese Studienarbeit wird an der OST, Ostschweizer Fachhochschule, im Herbstsemester 2023 durchgeführt. Sie wird von Thomas Corbat ([thomas.corbat@ost.ch](mailto:thomas.corbat@ost.ch)) betreut.

### 2. Studierende

Das Projekt wird im Rahmen des Moduls «Studienarbeit Informatik» im Departement Informatik von folgenden Studierenden durchgeführt:

- Jeremy Stucki ([jeremy.stucki@ost.ch](mailto:jeremy.stucki@ost.ch))
- Vina Zahnd ([vina.zahnd@ost.ch](mailto:vina.zahnd@ost.ch))

### 3. Einführung

Die Programmiersprache C++ wird durch das Standard-Komitee ständig weiterentwickelt. Eine signifikante Neuerung in C++20 sind Template Parameter Constraints, welche es erlauben die erwartete Funktionalität von Template-Parametern zu spezifizieren. Die sogenannten Concepts ermöglichen es, solche Prädikate für Template-Argumente zu benennen.

Refactoring ist in der Software-Entwicklung eine etablierte Technik, um Code und Design Smells zu eliminieren, beziehungsweise allgemein das interne Design zu verbessern, ohne das externe Verhalten zu verändern. Dazu gib es umfangreiche Sammlungen von Schritt-für-Schritt-Anleitungen, wie dabei vorzugehen ist. Automatisierte Tests stellen sicher, dass die korrekte Funktionsfähigkeit erhalten bleibt. Um diese effektiv anzuwenden, ist die Unterstützung der eingesetzten Programmierwerkzeuge essenziell.

Früher implementierten integrierte Entwicklungsumgebungen (IDE) Unterstützung bei der Analyse und Werkzeugen, wie automatisierten Refactorings, für die Zielsprache jeweils selbst. Mittlerweise ist solche Funktionalität oft in separate Software-Komponenten ausgelagert. Sogenannte Language Server implementieren diese Funktionalität und können über schlanke Plug-ins, welche über das Language Server Protocol kommunizieren, einfach in beliebige IDEs eingebunden werden.

Die neuen Konstrukte der C++20 Concepts bieten Potenzial, etablierte Refactorings anzuwenden. Zudem können möglicherweise neue Refactorings entwickelt werden.

## 4. Ziele des Projektes

In dieser Studienarbeit sollen mehrere Aspekte von möglichen C++20-Concept Refactorings durch die Studierenden bearbeitet werden:

- Allgemein bekannte Refactorings können möglicherweise auf Concepts angewendet werden. Dies ist möglicherweise bereits in aktuell verfügbarem Tooling implementiert.
- Die Concepts bieten Potenzial für neue Refactorings. Falls solche identifiziert werden können, sind diese zu definieren. Hier besteht Potenzial für ein wissenschaftliches Paper.
- Der clang Language Server soll um automatisierte Code Transformationsfunktionalität erweitert werden, um als Proof-of-Concept der vorgängig identifizierten Concept-Refactorings zu dienen.

Da die ersten beiden Punkte explorativen Charakter haben, können die genauen Ziele der Arbeit vorgängig nicht exakt definiert werden. Es ist keine Pflichtanforderung, dass die implementierten Features für clang auch in das Produkt übernommen werden, da dies teilweise ausserhalb des Einflussbereichs der Studierenden liegt.

## 5. Dokumentation

Das Projekt muss entsprechend der Richtlinien des Departments Informatik dokumentiert werden [1]. Dies umfasst alle Analyse, Design, Implementation und Projekt Management, usw. Kapitel. Die Dokumentation sollte vorzugsweise auf Englisch verfasst werden. Ein Projektplan muss am Anfang ausgearbeitet und entsprechend dem effektiven Fortschritt angepasst werden. Schlussendlich müssen alle Resultate komplett auf den Archiv-Server der OST hochgeladen werden.

Eine gedruckte Ausgabe der Dokumentation muss dem Betreuer abgegeben werden (farbig und doppelseitig gedruckt, gebunden).

## 6. Wichtige Daten

\* Die URLs könnten noch angepasst werden.

<b>18.09.2023</b>	<b>Semesterstart und Beginn des Projektes</b>
<b>Bis 21.12.2023</b>	<b>Erfassen des Abstracts im Online-Tool und Überprüfung durch den Betreuer [2]</b>
<b>22.12.2023, 17:00 Uhr</b>	<b>Finale Abgabe über den Archiv-Server [3]</b>

## 7. Bewertung

Bei erfolgreichem Bestehen der Studienarbeit erhalten die Studierenden 8 ECTS-Punkte. Der geschätzte Aufwand für ein ECTS beträgt 30 Arbeitsstunden (vgl. Modulbeschreibung [4]). Der Betreuer ist verantwortlich für die Bewertung der geleisteten Arbeit.

Kriterium	Gewicht
1. Organisation, Ausführung	1/5
2. Bericht (Abstract, Management Summary, technischer und persönlicher Bericht) sowie Struktur, Visualisierung und Sprache der ganzen Dokumentation.	1/5
3. Inhalt der Arbeit	3/5

Weiter gelten die generellen Regularien für Studienarbeiten im Departement Informatik.

## 8. Referenzen

- [1] <https://ostch.sharepoint.com/:b:/r/teams/TS-StudiengangInformatik/Freigegebene%20Dokumente/Studieninformationen/Studien-%20und%20Bachelorarbeiten/Informationen/Leitfaden%20BA%20SA%20v1.0.pdf?csf=1&web=1&e=1qAL6o>
- [2] <https://abstract.rj.ost.ch>
- [3] <https://avt.i.ost.ch>
- [4] [https://studien.rj.ost.ch/allModules/24386\\_M\\_SAI14.html](https://studien.rj.ost.ch/allModules/24386_M_SAI14.html)

Rapperswil, 17. September 2023

Thomas Corbat

Lehrbeauftragter

OST – Ostschweizer Fachhochschule