

# ACDK als Unternehmensstandard

## Abstrakt

ACDK ist nicht nur als eine Bibliothek unter vielen entworfen worden, sondern als eine Plattform, die die Entwickler auch im Unternehmenskontext unterstützt.

## Autor

Roger René Kommer

## Version

1.0

## Datum

2000-11-02

## Seiten

3

## Kontakt

Ingenieurbüro Kommer, artefaktur  
Dörnbergstrasse 11  
34119 Kassel  
Tel.: 0561-780-150  
Fax.: 0561-780-135  
<http://www.artefaktur.com>  
<mailto:kommer@artefaktur.com>

## **ACDK als Unternehmensstandard**

### **ACDK als Plattform**

ACDK ist nicht als eine Bibliothek unter vielen entworfen worden, sondern als eine Plattform, die die Entwickler auch im Unternehmenskontext unterstützt.

### **ACDK und Java**

Java™ hat sich in den letzten Jahren als ‚lingua franca‘ der IT-Branche für plattformunabhängige, multithreaded und netzwerkbezogene Problemfelder entwickelt.

ACDK als C++-Bibliothek orientiert sich in seinem Design explizit an der Java™ 2.0 Spezifikation, so dass Entwickler mit Vorkenntnissen in Java einen leichten Zugang zu ACDK haben.

### **Orientierung an etablierten Standards**

C/C++ macht es Entwicklern durch die unüberschaubare Vielzahl verschiedener, teilweise paralleler oder gar widersprechender Konzepte sehr schwer, zuverlässige, wiederverwendbare und skalierbare Anwendungen zu entwickeln. Die im C++-Standard enthaltene Standard Template Library (STL) bildet nur wenige Teilaspekte ab. Viele Aspekte moderner Anwendungen (Multithreading, Speichermanagement, IPC, RPC, etc.) werden in ihr gar nicht berücksichtigt.

Für diese ganzen Problemfelder bietet ACDK vorgefertigte Lösungen, die weitgehend der Java™ 2.0-Spezifikation entsprechen und somit einem einheitlichen Konzept folgen.

### **Unternehmenskomponenten in Java und C++**

Da ACDK in Semantik, Syntax und Idiomatik stark an Java angelehnt ist, können wiederverwendbare Unternehmenskomponenten parallel für Java und C++ entwickelt werden. Für die Dokumentation von Java und C++ kann der JavaDoc-Standard verwendet werden.

### **ACDK erweitern**

Im Gegensatz zu der STL und vielen anderen C++-Bibliotheken ist ACDK dazu geeignet, eigene Erweiterungen zu schreiben. Um eigene ACDK-Objekte zu erstellen, braucht es keine ausgeprägten Expertenkenntnisse. Durch ein durchgängiges Namespace- und Packagekonzept kann eine übersichtliche Gliederung der Projekte vorgenommen werden.

## Hilfe beim objektorientierten Design

Unternehmensstandards und -lösungen können als wiederverwendbare Pakete und Komponenten entwickelt werden. Durch klare Leitbilder und Design Pattern beim Objektdesign können Fehler schon in der Konzeption vermieden werden.

## ACDK für Anwendungen im Serverkontext

ACDK konzentriert sich auf die Aufgabenstellungen im Serverbereich und ist auf eine Kooperation mit verteilten Java-Anwendungen im Clientbereich eingestellt.

## Ökonomischer Umgang mit Entwicklerressourcen

Bei dem Einsatz von Java und C++ für Client- und Serverbereich kann ein einheitlicher Programmier- und Dokumentationsstandard verwendet werden. Personalressourcen können flexibler zwischen verschiedenen Abteilungen verteilt werden, da die Grundlagentechnologien verwandt sind. Auch neue Mitarbeiter können sich mit einer flachen Lernkurve und einer kleinen Frustrationsamplitude in ACDK einarbeiten.

