



---

**Title:** Plankalkül (PK) und Verwaltung  
**Author(s):** Konrad Zuse  
**Date:** 1973  
**Published by:** Konrad Zuse Internet Archive  
**Source:** Document - ZIA ID: 0302

---

The Konrad Zuse Internet Archive preserves and offers free access to the digitized original documents of Konrad Zuse's private papers and to other related sources.

The Konrad Zuse Internet Archive is a nonprofit service that helps scholars, researchers, students and other interested parties discover, use and build upon a wide range of content in a digital archive. For more information about the Konrad Zuse Internet Archive, please contact [zusearchive@zib.de](mailto:zusearchive@zib.de).

---

Your use of the Konrad Zuse Internet Archive indicates your acceptance of the Terms & Conditions of Use (<http://zuse.zib.de/tou>) including the following license agreement. If you do not accept the Terms & Conditions of Use you are not permitted to use the material.

This work by Konrad Zuse Internet Archive is licensed under a  
Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License  
(<http://creativecommons.org/licenses/by-nc-sa/3.0/>).  
Based on a work at <http://zuse.zib.de>



**Attribution (BY)** - You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work). Attribute with "Konrad Zuse Internet Archive (<http://zuse.zib.de>)".

**Noncommercial (NC)** - You may not use this work for commercial purposes.

**Share Alike (SA)** - If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

The usage of this document requires the consideration of possible third party copyrights, and might necessitate obtaining the consent of the copyright holder. The Konrad Zuse Internet Archive assumes no liability with respect to the rights of third parties. The Konrad Zuse Internet Archive is not responsible for the claims of any third party resulting from any infringement of copyright laws.

Professor K. ZUSE

Hünfeld, den 20. September 1973

Plankalkül (PK) und Verwaltung

Vorschlag für eine Untersuchung der praktischen Anwendungsmöglichkeiten des PK, insbesondere bei der Verwaltung als Ergänzung der laufenden GMD-Studie.

I. Allgemeines

Der PK wurde vom Verfasser im Jahre 1945 angeregt durch die Entwicklung von Computern (Z1 bis Z4, 1936 bis 1945) aufgesetzt mit der Zielsetzung, eine allgemein algorithmische Sprache zu schaffen, mit deren Hilfe sämtliche Rechenvorschriften im allgemeinsten Sinn exakt und eindeutig formuliert werden können. Das geschah unabhängig und losgelöst von praktischen Einsatzmöglichkeiten lediglich am Schreibtisch. Die für Deutschland ungünstigen Nachkriegsverhältnisse brachten es mit sich, dass die Arbeit zunächst unbeachtet in der Schublade liegen blieb.

Die weiteren Computerentwicklungen nach dem Kriege erfolgten wesentlich unter amerikanischem Einfluss. Es entstanden in den Jahren zwischen 1950 und 1960 mehrere Programmiersprachen wie z.B. COBOL, FORTRAN und ALGOL. Sie waren auf Tagesprobleme, in erster Linie also numerische Rechnungen, ausgerichtet und wurden unabhängig von den im PK geleisteten Vorarbeiten geschaffen. Sie dienten dem unmittelbaren Einsatz entsprechend dem damaligen Stand der Computertechnik.

Die weiteren Entwicklungen führten zu einer Art Evolution des Begriffes Rechnen über das reine Zahlenrechnen hinaus zu einem allgemeinen Rechnen mit Bedingungen, Begriffen, logischen Operationen usw. Das entsprach zwar dem Grundkonzept des PK; jedoch war der Abstand zu den inzwischen eingeführten Programmiersprachen und dem PK noch immer zu gross, um Anreiz zu geben, die in ihm enthaltenen Möglichkeiten genauer zu untersuchen.

Mit zunehmender Verbreiterung des Spektrums der Art der durchzuführen-  
den Rechnungen zeigte sich jedoch immer deutlicher, dass die gebräuch-  
lichen Programmiersprachen den Ansprüchen nicht mehr gewachsen waren  
und man erkannte die Notwendigkeit, höhere Programmiersprachen zu ent-  
wickeln. Die bekanntesten Beispiele hierfür sind PL/1 und ALGOL 68.  
Sie bauen im wesentlichen auf den bewährten algorithmischen Sprachen  
COBOL, FORTRAN und ALGOL 60 auf. Dadurch ist es erklärlich, dass sie,  
im Gegensatz zum PK, in ihrem Grundkonzept nicht von vornherein als  
logische universelle Sprachen angelegt sind, sondern Erweiterungen der  
bisherigen vornehmlich numerischen Aufgaben dienenden Sprachen dar-  
stellen. Andererseits enthalten sie eine Reihe von Kennzeichen, wel-  
che auf den inzwischen mit den Vorgängersprachen gemachten Erfahrun-  
gen aufbauen, und die, wie man heute sagt, im wesentlichen der "Imple-  
mentierung" dienen, d.h. dass sie unter dem Gesichtspunkt der prak-  
tischen Einsatzmöglichkeiten aufgesetzt wurden.

Es ist kein Geheimnis, dass diese Versuche bis jetzt noch nicht zu be-  
friedigenden Ergebnissen geführt haben. Die so geschaffenen höheren  
Programmierersprachen sind leider sehr kompliziert. Sie erfordern ent-  
sprechende komplexe Compiler mit langen Übersetzungszeiten, wodurch  
viele Vorteile dieser Sprachen wieder kompensiert werden. In dieser  
kritischen Situation erscheint es angemessen, den PK wieder hervorzu-  
holen und zu untersuchen, wie weit die darin enthaltenen Prinzipien  
uns heute weiter helfen können.

## II. Die laufende Untersuchung des PK

Mit Unterstützung der deutschen Forschungsgemeinschaft (DFG) und der  
Gesellschaft für Mathematik und Datenverarbeitung in Birlinghoven  
(GMD) wurde daher unter der Leitung des Unterzeichneten eine Unter-  
suchung mit folgender Aufgabenstellung in Angriff genommen:

"Wie gut ist der Plankalkül und inwiefern reicht er über algorithmi-  
sche Sprachen hinaus?"

An dieser Untersuchung arbeitet zunächst ausser dem Unterzeichneten  
ein junger Diplom-<sup>Intensiv</sup>ingenieur (Herr Hohmann) bei der Zweigstelle der

GMD in Darmstadt. Da die Untersuchung erst gerade begonnen hat, können detaillierte Ergebnisse noch nicht vorgelegt werden. Trotzdem kann bereits heute folgendes gesagt werden:

- 1) Verschiedene Bepfahrungen mit Fachleuten haben gezeigt, dass ähnliche allgemeine Untersuchungen algorithmischer Sprachen z.Zt. an anderer Stelle, insbesondere in Westdeutschland kaum durchgeführt werden.
- 2) Der Begriff "Algorithmus" bzw. "algorithmische Sprachen" wird anscheinend verschieden definiert. Klare Abgrenzungen sind noch nicht allgemein anerkannt. Demgegenüber kann die Auffassung vertreten werden, dass der PK eine Grundform einer universellen algorithmischen Sprache darstellt.
- 3) Beschränkt man den Begriff "algorithmische Sprachen" auf die eingeführten Programmiersprachen, so weist der PK einige wesentliche Züge auf, die darüberhinaus gehen. Er scheint insbesondere geeignet zu sein, als Basissprache zur Formulierung höherer moderner Programmiersprachen zu dienen. Die Grundkonzeption des PK erlaubt es, die wesentlichen Charakteristika einer algorithmischen Sprache in elementarer Weise zu formulieren und darauf aufbauend, Probleme der Implementierung zu simulieren.

Allgemein kann gesagt werden, dass der von der heutigen Informatik eingeschlagene Weg oft zu kompliziert ist, um den Forderungen der Praxis zu entsprechen. Was dringend not tut, ist ein Zurückgehen auf einfache elementare Formen, welche aber trotzdem die nötige Universalität aufweisen. Da der PK von vornherein nach diesen Gesichtspunkten aufgebaut wurde, bietet er die ideale Basis auch für zukünftige Entwicklungen.

In der Anlage sind die Gesichtspunkte zusammengestellt, nach denen die laufende Untersuchung des PK weiterhin durchgeführt werden soll (siehe Anlage 1).

### III. Vorschlag zur Ergänzung der Studie

In Ergänzung zu dieser laufenden Studie erscheint es insbesondere im Hinblick auf die kritische allgemeine Situation auf dem Gebiet der Software dringend erforderlich, bereits jetzt gewisse Voruntersuchungen im Hinblick auf eine praktische Verwertbarkeit des PK durchzuführen. Hierfür ist das Aufgabengebiet der Verwaltung besonders geeignet.

Dies sei am Beispiel der Informationsstruktur demonstriert. Das Thema ist z.Zt. Gegenstand intensiver Untersuchungen z.B. im Zusammenhang mit sogenannten Datenbanken. Es handelt sich dabei darum, umfangreiches Informationsmaterial so zu ordnen, dass es leicht wieder aufgefunden werden kann (information retrieval). Diese Datenbanken bedürfen selbstverständlich einer laufenden Pflege. Sie müssen geändert, ergänzt und auch teilweise gelöscht werden können.

Die heute eingeführten Programmiersprachen arbeiten mit einem breiten Spektrum von Strukturen, wie z.B. String, Array, Record, File, List, Baumstrukturen verschiedener Variation usw., deren Bedeutung nicht immer einheitlich ist, und welche oft durch Compiler schwer zu handhaben sind.

Demgegenüber arbeitet der PK mit Kombinationen von Elementarstrukturen, beginnend mit dem einfachen Ja-Nein-Wert (Bit) bis hin zu den kompliziertesten überhaupt denkbaren Datenstrukturen, welche alle auf ein einfaches Grundgesetz zurückgeführt werden können.

Eine Reihe von Formulierungen, welche in anderen Programmiersprachen besondere Vereinbarungen bzw. Erweiterungen der Sprache erfordern, können im PK als elementare Programme in der Basisform dargestellt werden. (Vergleiche hierzu Kapitel 2 des PK). Hierher gehören z.B. die heute oft benutzten verschiedenen Variationen des List-processing.

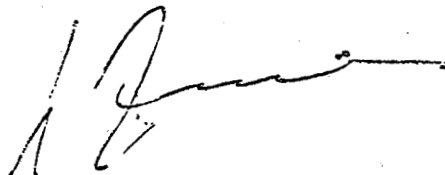
Weiterhin bietet der PK die Möglichkeit, Ein- und Ausgabeprogramme für spezielle Formate in der Basissprache zu schreiben. Es können

somit mit Hilfe einer dem PK angepassten Basissprache solche Programme simuliert werden, die in anderen Sprachen besonderer Spracherweiterungen bedürfen (z.B. die verschiedenen Formatanweisungen). Es besteht gute Aussicht, dass es auf diese Weise möglich sein wird, auch das Compilerproblem zu vereinfachen, so dass zunächst nur ein Compiler für die Basissprache entwickelt zu werden braucht.

All das zeichnet sich bereits heute ab und eine diesbezügliche Untersuchung könnte daher auch sofort unabhängig von der bereits laufenden Studie, jedoch als Ergänzung hierzu, durchgeführt werden. Um die Arbeiten einigermaßen zügig voran treiben zu können, wären 1 bis 2 Informatiker als zusätzliche Hilfskräfte für die Zeit von etwa 2 Jahren erforderlich.

Um die kritische Software-Situation noch zu unterstreichen, werden Kopien zweier Artikel, aus der Zeitschrift Data Exchange vom September 1973 (Herausgeber Firma Diebold) beigelegt (Dijkstra, The Humble Programmer; David Nelson, Some Comments on General Purpose Programming Languages).

Anlagen

A handwritten signature in dark ink, appearing to be 'A. B.', is written on the page.