



Doctoral Thesis

Vamos

Entwurf und Realisierung eines erweiterbaren Betriebssystems für Arbeitsplatzrechner

Author(s):

Peschel, Frank

Publication Date:

1988

Permanent Link:

<https://doi.org/10.3929/ethz-a-000507796> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Vamos–
Entwurf und Realisierung eines erweiterbaren
Betriebssystems für Arbeitsplatzrechner

Abhandlung zur Erlangung des Titels eines
DOKTORS DER TECHNISCHEN WISSENSCHAFTEN
der
EIDGENOESSISCHEN TECHNISCHEN HOCHSCHULE ZUERICH

vorgelegt von
Frank Volker Peschel
Diplom Informatiker
Technische Universität Berlin
geboren am 2.10.1959
von BRD

Angenommen auf Antrag von:
Prof. Dr. N. Wirth, Referent
Dr. S.E. Knudsen, Korreferent

Kurzfassung

Vamos ist ein Einbenutzer-Betriebssystem für Arbeitsplatzrechner, das die gleichzeitige Ausführung mehrerer Programme unterstützt. Es gehört zur Kategorie der offenen Betriebssysteme, d.h. sowohl der Zugriff auf systeminterne Schnittstellen als auch die transparente Erweiterung des Systems sind möglich.

Obwohl sich alle Programme in einem einzigen globalen Adressraum befinden, können mit Unterstützung geeigneter Hardware deren Daten vor gegenseitigem, unbeabsichtigtem Überschreiben geschützt werden. Grundlage hierfür bildet die Definition von Sichtbarkeitsbereichen innerhalb des globalen Adressraums, die als *Domänen* bezeichnet werden. Zusätzlich können sie zur logischen Gruppierung von Modulen benutzt werden. Domänen sind die Elemente, mit denen die statische Struktur des laufenden Systems, also die zur Verfügung stehende Funktionalität, verändert werden kann.

Diese Arbeit beschreibt die Kriterien und grundlegenden Konzepte, die beim Entwurf von Vamos ausschlaggebend waren. Dabei werden richtungsweisende oder häufig verwendete Betriebssysteme für Arbeitsplatzrechner vorgestellt. Es zeigt sich, dass bei der Entwicklung dieser Systeme deren Offenheit und Speicherschutz als sich gegenseitig ausschliessende Alternativen angesehen wurden. In Vamos sind diese beiden Konzepte voneinander unabhängig. Die Realisierung von Speicherschutzmassnahmen hat keinen Einfluss auf die Offenheit des Systems. Aus diesem Grund unterscheiden sich die beiden Implementierungen von Vamos (mit und ohne Speicherschutzmassnahmen) nur in 10 Zeilen Programmcode innerhalb eines Moduls.

Die dynamische Strukturierung eines Programmes und des gesamten Systems erfolgt mit Hilfe von Prozessen, die wiederum jeweils in mehrere Ausnahme- und Fehlerbehandlungsebenen eingeteilt werden können. Die von Vamos zur Verfügung gestellten Synchronisationsverfahren von Prozessen sind mit der Ausnahme- und Fehlerbehandlung gekoppelt, sodass die Realisierung von auch im Fehlerfall verklemmungsfreien Systemen möglich ist. Die dafür benötigten Basisoperationen stehen auch Benutzerprogrammen zur Verfügung, damit weitere Verfahren realisiert werden können.

Die Implementierung erfolgte, mit Ausnahme eines sehr kleinen Kerns, vollständig in Modula-2 und unterstützt nur die Ausführung von in dieser Sprache definierten Programmen. Es zeigt sich, dass Vamos eine sehr mächtige und flexible Umgebung für die Entwicklung und Ausführung von Programmen darstellt. Die Klarheit und Verständlichkeit des Systems aus Benutzersicht beruht auf der klaren Trennung der angebotenen Konzepte und drückt sich auch in der geringen Grösse der Implementierung aus.

Abstract

Vamos is a single-user operating system for workstation computers. It supports the parallel execution of multiple programs. Vamos is an "open" operating system, i.e. the invocation of operations internal to the system is possible as well as the transparent extension of the system's functionality.

All running programs reside in a single global address space. In the presence of an adequate hardware, the program's code and data are protected against mutual erroneous overwriting. This is achieved by the definition of areas of restricted visibility in the global address space, which are called *domains*. Additionally, domains are used to build logical groups of modules. Domains are the building blocks which are used for the definition of the running system's static structure.

This thesis describes the criteria and basic concepts that has led to the operating system Vamos. Both current trends and a few modern operating systems for workstation computers are presented in an exemplary manner to illustrate the discussed concepts. It turns out that the designers of those systems considered the concepts of openness and memory-protection as mutually exclusive alternatives. In Vamos both concepts are independent, i.e. the implementation of memory-protection mechanisms has no influence on the system's openness. Due to this fact both implementations of Vamos (with and without memory-protection) differ only in 10 lines of one single module.

The dynamic structure of a program as well as that of the complete system is represented by processes, which can be subdivided into multiple exception levels. The process-synchronization policies offered by Vamos are connected with the exception-handling facilities. Therefore it is possible to implement systems which are deadlock-free even in the presence of exceptions. The basic operations needed for such mechanisms can be used by some user-defined synchronization schemes.

The complete system is written in Modula-2 with the exception of a small kernel. It supports only the execution of programs written in this language. It turns out that Vamos is a powerful and flexible environment for the development and execution of programs. The system's clarity and comprehensibility from the user's point of view rely on the strict separation of the offered concepts, and it is reflected by the modest size of the actual implementation.