

2.1 Operating system objectives and functions

Thursday, March 26, 2020 4:52 PM

Et operativsystem (OS) er et program som kontrollerer execution av applikasjonsprogrammer, og fungerer som et grensesnitt mellom applikasjoner og computerens hardware. OS-et har tre hovedmål:

1. **Bekvemmelighet:** Et OS gjør computeren mer lettvinnt å bruke.
2. **Effektivitet:** Et OS tillater computersystemets ressurser å bli utnyttet på en mest mulig effektiv måte.
3. **Mulighet til å utvikle:** Et OS bør konstrueres slik at det kan effektivt videreutvikles, testes, og introduseres til nye systemfunksjoner uten at det krasjer med tjenesten.

OS: som et bruker/computer-grensesnitt

Slutbrukeren som ønsker å bruke systemet har typisk ikke innsikt i detaljene til hardware, og ser som regel på et computer system som et sett med applikasjoner. Bak kulissene foregår det mye som lar sluttbruker kunne benytte seg av applikasjonene, og OS-et har som funksjon å gjøre denne opplevelsen så smud og transparent som mulig.

OS-et tilbyr tjenester på følgende måte:

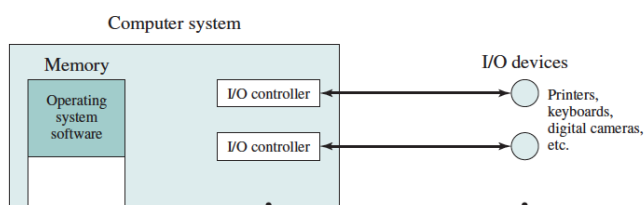
- **Programutvikling:** OS-et tilbyr ulike tjenester som for eksempel integrerte utviklingsmiljøer (IDE) med debugging for å hjelpe programmerer til å kunne lage programmer. Dette er strengt tatt ikke en del av kjernen til OS-et, men et supplement (*application program development tools*).
- **Program execution:** Det er flere steg som må gjennomføres for å utføre et program. Instruksjoner og data må lastes inn til hovedminnet, I/O-enheter og filer må initialiseres, og andre ressurser må forberedes. OS-et håndterer planleggingen av disse stegene.
- **Adgang til I/O-enheter:** Hver I/O-enhet krever et særegnet sett med instruksjoner eller kontrollsignaler for operasjoner. OS-et tilbyr et uniformt grensesnitt som skjuler disse detaljene så programmerere kan aksessere slike enheter med enkle read- og write-operasjoner.
- **Kontrollert adgang til filer:** For systemer med flere brukere kan OS-et tilby beskyttelsesmekanismer for kontrollaksesser til ulike filer. Ved fil-aksess må OS-et ta hensyn til ulike faktorer som hva slag I/O-enhet det er (e.g. disk, tape), men også datastrukturen inneholdt i filene i lagringsmediumet.
- **Systemaksess:** For delte eller offentlige systemer, så kan OS-et kontrollere adgang til systemet som en helhet og til spesifikke systemressurser. Aksessfunksjonen må tilby beskyttelse av ressurser og data fra uautentiserte brukere, og må løse konflikter for ressurs-vanskeligheter (?).
- **Feildetektering og respons:** Det kan oppstå flere errorer når et computersystem kjører. Både interne og eksterne errorer. I hvert tilfelle må OS-et håndtere en respons som klargjør error-tilstanden med minst skadeomfang på kjørende applikasjoner. Responsen kan variere fra å avslutte programmet som forårsaket erroren, eller prøve operasjonen på nytt, eller eksempelvis å rapportere om feilen til applikasjonen.
- **Bokføring:** Et bra OS samler på bruker-statistikk for ulike ressurser og følger med på ytelseparametere som responstid. Dette er viktig for å kunne forbedre systemet i fremtiden.
- **Instruction set architecture (ISA):** ISA definerer repertoaret av maskinspråkets instruksjoner som prosessoren kan utføre. Dette grensesnittet er grensen mellom hardware og software.
- **Application binary interface (ABI):** ABI definerer en standard for binær mobilitet (bærbarhet) mellom programmer. ABI definerer systemets kallings-grensesnitt til OS-et, og hardware ressursene og tjenestene tilgjengelig i et system gjennom bruker-ISA.
- **Application programming interface (API):** API-et gir et program tilgang til hardware ressurser og tjenester tilgjengelig i et system gjennom bruker-ISA-et (supplementert med høynivå-språk bibliotek-kallinger). Bruken av et API muliggjør applikasjon-software til å enkelt bli portert, uten rekompilering, til andre systemer som støtter samme API.

OS: som en ressursfordeler

Et OS er ansvarlig for å kontrollere bruken av computerens ressurser, sånn som I/O, hoved- og sekundærminnet, og prosessor execution-tid. OS-et skiller seg som kontrollmekanisme i form av at

- OS-funksjonene fungerer på samme måte som ordinær computer software, det vil si at det er et program som utføres av prosessorer.
- OS-et frasier seg hele tiden kontroll, og er dermed avhengig av at prosessoren gir den tillatelse til å gjenoppta kontroll igjen.

Figuren under viser de hovedressursene som håndteres av OS-et.



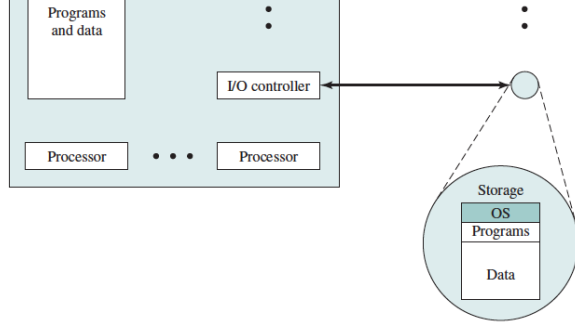


Figure 2.2 The Operating System as Resource Manager

En del av OS-et ligger i hovedminnet. Dette inkluderer kjernen (kernel), som inneholder de mest brukte funksjonene i OS-et, i tillegg til andre deler av OS-et som allerede er i bruk.

OS-et bestemmer når en I/O-enhet kan brukes av et program i execution, og kontrollerer aksess til og bruk av filer.

OS: Evolusjonen

Behovet for å hele tiden måtte oppdatere operativsystemet for å imøtekomme bransje-standarder og brukerbehov gjør at et OS må designes på en viss måte som muliggjør tilpasninger i fremtiden. Derfor bør systemet være modulbasert i konstruksjonen, med veldefinerte grensesnitt mellom modulene, og så bør systemet være veldokumentert.