

# Bättre dataprogram kräver nära samarbete mellan läkare och tekniker

## Några tankar om dagens rudimentära datateknik i primärvården

Många program för datorstött patientarbete är dåligt anpassade till primärvårdsläkarens behov. För att åstadkomma ett bra dataprogram krävs ett nära samarbete mellan systemutvecklare och användare, där användaren ansvarar för idéerna och systemutvecklaren för dess förverkligande. Läkarna bör värna om det datorstödda patientarbetet och se till att få ett övergripande ansvar för utvecklingen av datorverktyget.

TED GOLIGER

ST-läkare, Nässjö vårdcentral

JÖRN FRANK NIELSEN

distriktsläkare, Nässjö vårdcentral

HENNING PEDERSEN

distriktsläkare, Eksjö vårdcentral

TOM WICKSTRÖM

leg läkare, administration Högländssjukhuset, Eksjö

■ På så gott som alla vårdcentraler används numera datorer som arbetsredskap och det finns en mycket stor potential i detta. Oberoende av vad man anser om datorisering kommer framtiden att erbjuda direkt datakommunikation mellan olika vårdinstanser, centrala dataregister, länkning till rådgivande kunskapsystem och mycket annat som vi idag inte kan blicka över.

### Datorstött patientarbete

Datorer har funnits i primärvården i många år och har intagit en viktig plats i vårt arbete. Vi använder oss av olika dataprogram som i själva verket utgör integrerade verktyg för styrning och uppföljning av hela arbetsprocessen på vårdcentraler. Man kan utan överdrift tala om datorstött patientarbete där datorbaserade journalanteckningar, medicinlistor, tidbokning och provsvar utgör några av hörnstenarna, Figur 1. Olika yrkeskategorier jobbar i dessa system och närmar sig tekniken från var sitt håll. För sekreterare handlar det om ordbehandlingsprogram fyllda med mallar och standardformulär, för sköterskor dokumentations- och informationssystem, för laboratoriet registerprogram med prov och provsvar osv.

Det finns en yrkeskategori som använder detta redskap i hela dess bredd. Varje primärvårdsläkare skriver inom loppet av någon timme recept för en pa-

tient, kontrollerar provsvar för en annan, bokar in telefotid för en tredje och läser tidigare anteckningar om en fjärde. Utifrån vårt arbetssätt behövs ett platt uppbyggt datasystem där man lätt och snabbt kan förflytta sig inom hela programmet med hjälp av ett fåtal robusta kommandon, som självklart ska gälla tvärs över hela programpaketet. I stället får vi arbeta i ganska klumpiga konglomerat, där vi med hjälp av invecklade kommandosekvenser tvingas klättra i hierarkiska strukturer och moduler med sinsemellan inkonsekvent kommandosyntax. Det går naturligtvis att förlika sig med så länge man jobbar inom systemets enskilda funktioner men är ett gissel när man ska röra sig inom hela systemet.

### I skuggan av pappersarkivet

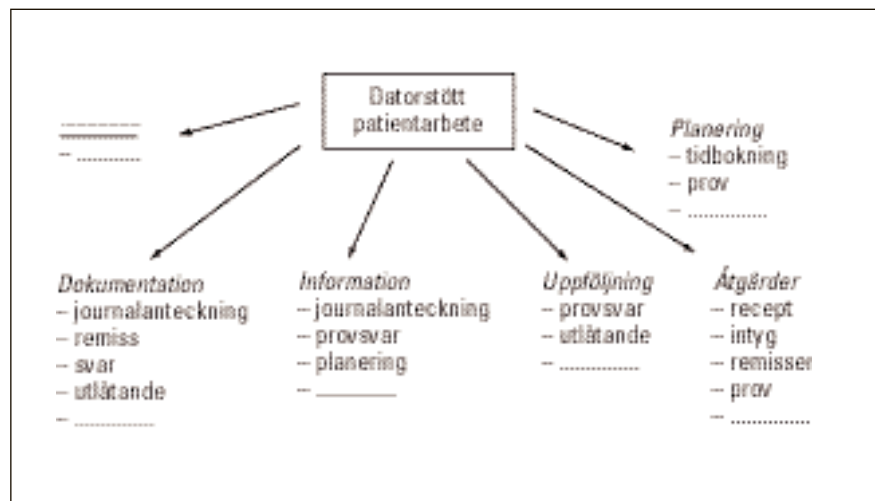
Det finns en orsak och en förklaring till att det kunde bli på det här sättet. Förklaringen ligger i att man i detta moderna verktyg har kopierat in gamla arbetsme-

toder och traditionellt tänkande. Våra datasystem är i själva verket kopior av pappersarkiv, dvs hierarkiskt uppbyggda strukturer med hyllor, mappar och flikar. Denna uppbyggnad utgör pappersarkivets innersta väsen och är en förutsättning för att kunna lagra och söka pappersburen information, Figur 2.

### Fotbollsplan med journalsidor

Datorer lagrar information på ett helt annat sätt. Man kan tänka sig databasen som en fotbollsplan där alla papper ligger utlagda efter ett visst mönster och dataprogrammet som en optisk sökare som svävar över planen och blixtnsnabbt kan förflytta sig till önskat ställe för att blicka över ett valt dokument, Figur 3.

För varje flytt räcker naturligtvis ett enda kommando och det är en självklarhet att alla kommandon ser likadana ut över hela planen, alltså en enhetlig kommandosyntax. Men det är också exakt så vi vill och måste arbeta för att ha nytta av datorer. Vi behöver snabbt och lätt kunna röra oss i systemet för att t ex ögna igenom journalanteckning på en patient, en stund senare skriva recept för en annan, därefter kontrollera alla inkomna provsvar och sedan gå tillbaka till den första patienten. Inte konstigt att det känns otympligt när vi tvingas in i pappersarkivets hierarkiska strukturer, inte heller konstigt att datorn blir slö när den tvingas in på omvägar där den måste in-



Figur 1. Arbetet på vårdcentral innefattar en mängd olika, inbördes sammanflätade aktiviteter. Redan idag används integrerade dataprogram för styrning och uppföljning av hela arbetsprocessen. Man kan utan överdrift tala om datorstött patientarbete.

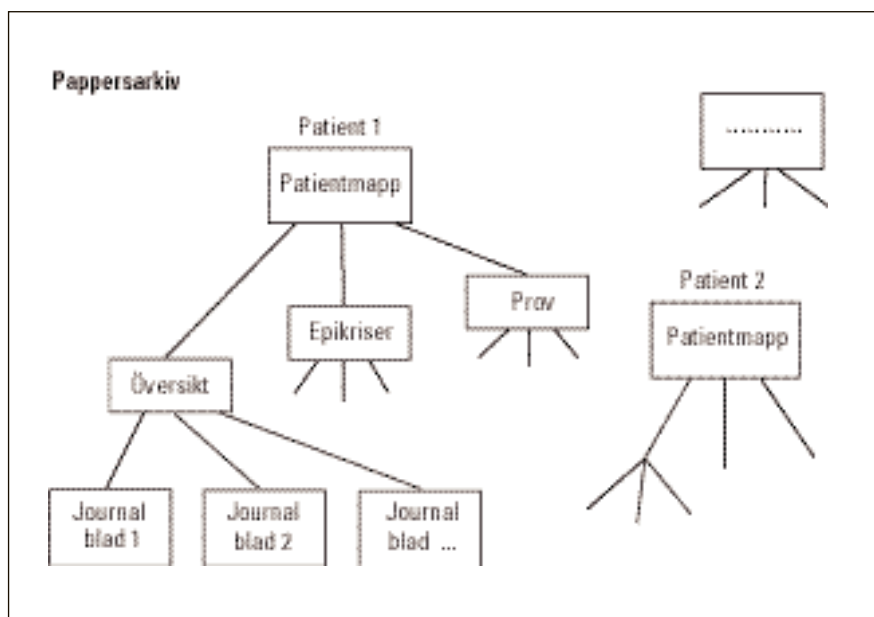
vänta våra mänskligt långsamma tangenttryckningar.

### Användarens ansvar och rättighet

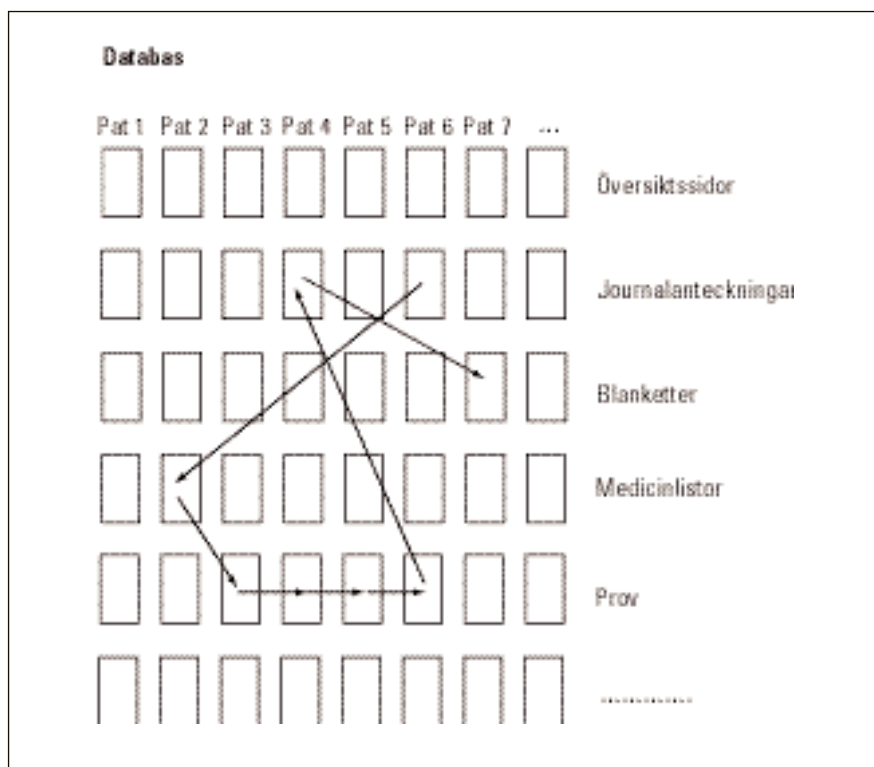
Orsaken till att systemen inte är bättre anpassade till primärvårdsläkarens arbetssätt och behov kan sökas i att vi på läkarsidan inte har varit med och i tillräcklig omfattning påverkat utveckling av programvaran. För att åstadkomma ett bra dataprogram krävs ett intimt samarbete mellan systemutvecklare och användare, där användaren ansvarar för idéerna och systemutvecklaren för dess förverkligande. Självklart måste man utgå ifrån den bredaste och mest heltäckande användningen. Det är därför vår uppgift som primärvårdsläkare att värna om konceptet med datorstött patientarbete och vår rättighet att ha ett övergripande ansvar för och kontroll över utveckling av de datorverktyg som vi arbetar med.

En andra generations program har lämnat provbänkarna och en besvikelse är ett faktum. Det är hög tid att tänka igenom situationen och agera. Bakom hörnet lurar en fara i form av oss själva. Efter en tids datortrassel lär sig fingrarna vilka 3–4–5 knapptryckningar som behövs för varje enskild operation och börjar omedvetet ursäktas dessa magnifika rudiment till datasystem medan pappersarkivets innersta väsen småler och gnuggar händer.

Ytterst handlar alltsammans om en enkel fråga. Ska några tusen primärvårdsläkare anpassa sig till några dataprogram eller tvärtom? •



Figur 2. Pappersarkiv är alltid hierarkiskt uppbyggda. Sådan uppbyggnad är en förutsättning för att man skall kunna söka pappersburen information.



Figur 3. Datasystem kräver inte någon hierarkisk uppbyggnad. Det räcker att informationen lagras på ett strukturerat sätt. Man kan jämföra en databas med en fotbollsplan där alla papper ligger utlagda efter ett visst mönster.