(12) **SØKNAD**

(19) NO (21) **20171308** (13) **A1**

NORGE

(51) Int Cl.
G06F 21/55 (2013.01)
G06F 21/57 (2013.01)
G06F 15/16 (2006.01)

Patentstyret

(21)	Søknadsnr	20171308	(86)	Int.inng.dag og søknadsnr	2007.11.30 PCT/US2007/086195
(22)	Inng.dag	2017.08.04	(85)	Videreføringsdag	2017.08.04
(24)	Løpedag	2007.11.30	(30)	Prioritet	2006.12.01, US, 11/566,170
(41)	Alm.tilgj	2009.08.31			
(62)	Avdelt fra	20092482, med inndato 2009.07.01			

(71) Innehaver: Microsoft Technology Licensing, LLC, One Microsoft Way, US-WA98052 REDMOND, USA

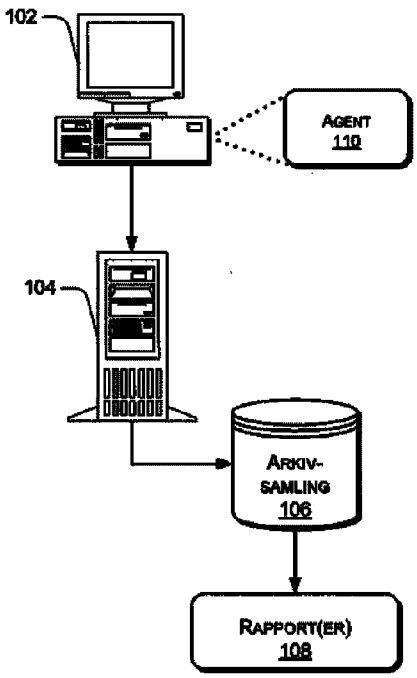
(72) Oppfinner:
Chad Verbowski, c/o Microsoft Corporation, International Patents, One Microsoft Way, US-WA98052 REDMOND, USA
Junhan Lee, c/o Microsoft Corporation, International Patents, One Microsoft Way, US-WA98052 REDMOND, USA
Xiaogang Liu, c/o Microsoft Corporation, International Patents, One Microsoft Way, US-WA98052 REDMOND, USA
Roussi Roussev, c/o Microsoft Corporation, International Patents, One Microsoft Way, US-WA98052 REDMOND, USA
Yi-min Wang, c/o Microsoft Corporation, International Patents, One Microsoft Way, US-WA98052 REDMOND, USA

(74) Fullmektig: Bryn Aarflot AS, Postboks 449 Sentrum, 0104 OSLO, Norge

(54) Benevnelse **Systemanalyse og håndtering**

(57) Sammendrag

Systemer og fremgangsmåter for å utøve systemforvaltning som er basert på undersøkelse av vekselvirkningene mellom ett eller flere programmer og den vedholdte tilstanden de gjerne representerer. Systemet muliggjør deteksjon av modifikasjoner som finner sted innenfor et system, verifikasjon av hvorvidt modifikasjonene er godkjente eller ikke og generering av varslinger av detekterte ukjente modifikasjoner.



BAKGRUNN

[0001] En hovedutfordring ved oppbygging av et pålitelig og sikkert datasystem er forvaltning av en vedholdt tilstand (PS – Persistent State) for systemet, som omfatter alle eksekverbare filer, konfigurasjonsinnstillinger og andre data som styrer hvordan systemet fungerer. Uriktige konfigurasjoner og andre PS-relaterte problemer er blant hovedårsakene til sammenbrudd og sikkerhetssårbarheter i en rekke forskjellige systemer fra enkeltstående, stasjonære maskiner til storskala Internett-tjenester. PS-relaterte problemer, sammen med problemer forårsaket av svikt i systemelementer så som maskinvarekomponenter og programlogikk, kan ha en ødeleggende innvirkning på hele systemet.

[0002] Kostnaden ved ikke å forvalte et systems PS-tilstand på en effektiv måte er høy. For eksempel kan PS-relaterte problemer gjenoppstå etter en omstart av systemet eller omstart av en applikasjon. Videre vil PS-tilstand endre seg under kjøring som følge av endringer så som patcher og applikasjonsrelaterte oppdateringer. Det finnes i dag ingen effektiv måte å lukke sløyfen av endringer som finner sted i systemet. I et slikt scenario, dersom kjent problemidentifisering slår feil og dersom en påfølgende omstart av systemet/en applikasjon ikke løser PS-problemet, kan en bli nødt til å undersøke systemet manuelt for å identifisere opphavet til PS-problemene.

[0003] Manuell undersøkelse av et system for å identifisere opphavet til PS-problemer er vanskelig og kostbart som følge av det store antallet mulige problemer. For eksempel er det potensielle settet av tilstander som kan påvirke en applikasjon med problemer enormt, og følgelig kan en liste av mulige problemårsaker omfatte et fullstendig tilstandssett på systemet. I tillegg vil situasjonen kunne forverre seg ytterligere dersom en også tar hensyn til enhver mulig kombinasjon av sett, spesielt i tilfeller der det ikke er ett enkelt opphav til PS-problemer.

OPPSUMMERING

[0004] Denne oppsummeringen er gitt for å introdusere forenklede konsepter ved modellbasert lisenstelling, som er beskrevet nærmere nedenfor i den detaljerte beskrivelsen. Denne oppsummeringen er ikke ment å identifisere avgjørende trekk ved oppfinnelsen det kreves beskyttelse for, og den er heller ikke ment for bruk til å

bestemme rammen til oppfinnelsen det kreves beskyttelse for.

[0005] I en utførelsesform blir programmer i en databasert anordning katalogisert og listeført, tidspunktet for siste innlasting av programmer registrert på den databaserte anordningen blir innhentet og tidspunktet for siste modifikasjon av filer tilknyttet programmene som er registrert hos den databaserte anordningen blir sammenliknet med tidspunktet for siste innlasting.

KORT BESKRIVELSE AV FIGURENE

[0006] Den detaljerte beskrivelsen er gitt under henvisning til de vedlagte figurene. I figurene identifiserer det eller de første sifrene i et referansenummer figuren hvor referansenummeret forekommer første gang. De samme referansenumre er anvendt i alle figurene for å henvise til like trekk og komponenter.

Figur 1 illustrerer et eksempel på arkitektur for systemforvaltning.

Figur 2 illustrerer et eksempel på samletjener.

Figur 3 illustrerer et eksempel på grafisk grensesnitt som viser en generert varsling.

Figur 4 illustrerer et eksempel på grafisk grensesnitt som viser avhengigheten til kjøringen av ett program av kjøringen av et første program.

Figur 5 illustrerer ett eller flere eksempler på fremgangsmåter for å fange opp data knyttet til modifikasjon av PS for et system.

Figur 6 illustrerer ett eller flere eksempler på fremgangsmåter for å klassifisere varslede (noted) endringer.

Figur 7 illustrerer ett eller flere eksempler på fremgangsmåter for å hindre gjennomføring av uautoriserte vekselvirkninger.

Figur 8 illustrerer ett eller flere eksempler på fremgangsmåter for å detektere ett eller flere utvidelsespunkter.

Figur 9 illustrerer ett eller flere eksempler på fremgangsmåter for å detektere lekkede (leaked) elementer.

Figur 10 illustrerer ett eller flere eksempler på fremgangsmåter for å detektere vanlige feilkonfigurasjoner eller foreldede filer.

Figur 11 illustrerer et eksempel på datamaskinmiljø.

DETALJERT BESKRIVELSE

[0007] For å realisere det ovennevnte omfatter systemet ett eller flere dataprogrammer eller agenter som rapporterer data knyttet til modifikasjonene som finner sted innenfor et system. Dataene omfatter informasjon knyttet til alle vekselvirkninger med filer og/eller innstillinger. Slike typer veekselvirkninger omfatter aktiviteter som lese- og skriveaksess til registerelementer, filer, samt vekselvirkninger med binære moduler, så som innlasting, og annet. Agentene rapporterer de innsamlede dataene til en underliggende tjeneste som behandler den rapporterte informasjonen for aktiviteter som generering av web-rapporter, varslinger eller integrasjon med andre tjenester for utøvelse av systemforvaltning. Videre kan behandlingen også skje på én enkelt maskin der dataene er samlet inn. Dette omfatter generering av rapporter, varslinger, etc. Spesielt blir en vedholdt tilstand (PS – Persistent State) for systemet behandlet, der PS-tilstanden omfatter alle eksekverbare filer, konfigurasjonsinnstillinger og andre data som styrer hvordan systemet fungerer. Selv om vedholdt tilstand er beskrevet, må det forstås at de beskrevne teknikkene og fremgangsmåtene vil kunne anvendes med andre typer tilstander.

[0008] Dataene som rapporteres kan bli anvendt for flere formål. For eksempel kan data bli undersøkt for å verifisere at vekselvirkninger som fremkalles er i overensstemmelse med et fastsatt regelsett eller er knyttet til en autorisert vekselvirkning.

[0009] Selv om aspekter ved beskrevne systemer og fremgangsmåter for systemforvaltning kan realiseres i en lang rekke forskjellige datasystemer, -miljøer og/eller -anordninger, er utførelsesformer av analyse og forvaltning i systemer beskrevet i forbindelse med følgende eksempler på systemarkitekturer.

Eksempel på system

[0010] Figur 1 illustrerer et eksempel på datasystem 100 der informasjonen knyttet til vekselvirkninger mellom ett eller flere programmer kan bli samlet inn og analysert. Systemet 100 omfatter en databasert anordning 102 der ett eller flere programmer kjører eller er installert, en samletjener 104, en arkivsamling 106 og én eller flere rapporter 108.

[0011] Informasjonen knyttet til vekselvirkningene mellom ett eller flere programmer og/eller filsystemer eller innstillinger er representativ for endringene i den vedholdte tilstanden (PS) som kan finne sted innenfor systemet 100. Den

databaserte anordningen 102 kan omfatte et hvilket som helst antall databaserte anordninger 102. For eksempel kan systemet 100 i én utførelse også omfatte et bedriftsnettverk, omfattende tusenvis av personlige datamaskiner (PC-er), forskjellige tjenere og andre databaserte anordninger spredt over flere land, som alle fungerer som databaserte anordninger 102. Alternativt kan systemet 100 i en annen mulig utførelse omfatte et hjemmenettverk med et begrenset antall PC-er. De databaserte anordningene 102 kan være koblet til hverandre i forskjellige kombinasjoner over et kabelbasert og/eller trådløst nettverk, omfattende et LAN, WAN eller en hvilken som helst annen nettverksteknologi kjent for fagmannen.

[0012] De databaserte anordningene 102 kan omfatte en agent 110 som er i stand til å instrumentere funksjoner i systemet 100 for å fange opp informasjon knyttet til vekselvirkninger mellom én eller flere databaserte anordninger 102 og/eller filsystemer og innstillinger. I én utførelse kan agenten 110 være en tråddata-registrerer (TDR) i stand til å modifisere, legge til og/eller slette datamaskinlesbare instruksjoner i funksjonen for å fange opp tråder som anroper funksjonen. I en annen mulig utførelse omfatter instrumentering av funksjoner også modifisering, tillegging og/eller sletting av datamaskinlesbare instruksjoner i en funksjon for å sørge for at en tråd eksekverer datamaskinlesbare instruksjoner i funksjonen som muliggjør oppfanging av data knyttet til tråden. I nok en annen utførelse omfatter data tilknyttet tråden informasjon vedrørende et program som tråden et tilknyttet, én eller flere vekselvirkninger forbundet med tråden og informasjon vedrørende brukeren av programmet som tråden tilhører. Selv om en TDR er beskrevet, må det forstås at oppfanging ikke nødvendigvis er påkrevet for alle algoritmer, og derfor trenger de beskrevne teknikkene og fremgangsmåtene ikke nødvendigvis være bundet til TDR-basert datainnsamling. Videre kan virtuell maskin-(VM)-basert instrumentering skille seg fra TDR-basert instrumentering der kode blir lagt til dynamisk. I en virtuell maskin kan det være en hardkodet funksjon internt i den virtuelle maskinen for å gjøre denne typen innsamling.

[0013] De instrumenterte funksjonene kan omfatte funksjoner som kan bli anropt av et program/en prosess. I én utførelse kan de instrumenterte funksjonene omfatte lavnivå tilknytningspunkt-funksjoner, så som filsystemdrivere, registerfunksjoner, funksjoner som oppretter nye prosesser og/eller tjenester, etc.

[0014] Data fanget opp fra tråder av tråddata-registratoren kan bli lagret og/eller behandlet for å styre oppførselen til systemet 100, og for å undersøke et forhold eller den vedholdte tilstanden til systemet 100. Typer av data som kan bli fanget opp fra tråder av tråddata-registrereren og tråddata-registrererens virkemåte er beskrevet nærmere i U.S.-patentsøknaden med tittel "Thread Interception and Analysis" av Verbowski m.fl., søknadsnummer _____, innlevert _____, som med dette inntas som referanse.

[0015] Samletjeneren 104 har ansvar for å samle inn informasjon vedrørende endringer som kan ha funnet sted i systemet 100. I én utførelse lagrer agenten 110 informasjonen knyttet til vekselvirkningene i samletjeneren 104 som komprimerte logger. I nok en annen utførelse kan informasjonen knyttet til vekselvirkningene også bli lastet opp til en arkivsamling 106. Analysen av informasjonen samlet i samletjeneren 104 eller arkivsamlingen 106 blir anvendt for å generere én eller flere rapporter 108. Rapporten(e) 108 generert som et resultat av analyse utført på informasjon samlet på samletjeneren 104 eller arkivsamlingen 106 gir kunnskap om vekselvirkninger eller modifikasjoner som kan finne sted innenfor én eller flere databaserte anordninger 102. I en annen utførelse kan rapporten(e) 108 bli generert gjennom et grafisk grensesnitt. I nok en annen utførelse kan det grafiske grensesnittet være implementert gjennom en nettleser for fremhenting og fremvisning av tidligere opprettede og/eller bufrede rapporter, for gjennomføring av programmerbar dataaksess av informasjonen lagret i samletjeneren 104 eller arkivsamlingen 106. Samletjeneren 104 og arkivsamlingen 106 kan befinne seg på eller være en del av én enkelt anordning som fungerer enten som samletjener 104 eller arkivsamling 106.

[0016] Som angitt over kan informasjonen samlet inn av agenten 110 og lagret i samletjeneren 104 eller arkivsamlingen 106 bli analysert for å frembringe kunnskap om virkemåten til systemet 100. Analysen som blir utført kan omfatte deteksjon av avvik, håndtering av endringer, håndtering av unormal systemaktivitet, identifisering av sikkerhetssårbarheter, identifisering av uautoriserte applikasjoner, gjennomføring av overholdelseskontroll, osv.

[0017] Figur 2 illustrerer et eksempel på samletjener 106 innrettet for å lagre, behandle og/eller analysere data fra agenten 110. Samletjeneren 106 omfatter én eller flere prosessorer 202 og et minne 204. Prosessoren(e) 202 omfatter for

eksempel mikroprosessorer, mikrodatamaskiner, mikrokontrollere, digitale signalprosessorer, sentralprosesseringsenheter, tilstandsmaskiner, logiske kretser og/eller hvilke som helst anordninger som behandler signaler basert på kjøreinstruksjoner. Blant andre funksjoner er prosessoren(e) 202 innrettet for å hente frem og eksekvere datamaskinlesbare instruksjoner lagret i minnet 204.

[0018] Minnet 204 kan være et hvilket som helst datamaskinlesbart medium kjent for fagmannen, for eksempel volatilt minne (f.eks. RAM) og/eller ikke-volatilt minne (f.eks. ROM, flash og annet). Minnet 204 kan også inneholde ett eller flere programmer 206 og data 208. Programmet/programmene 206 kan utføre, blant andre operasjoner, spørringsrelaterte prosesser på data knyttet til vekselvirkninger mellom programmer som kjører på én eller flere databaserte anordninger 102 og filsystemer og/eller innstillinger. Programmet/programmene 206 omfatter videre for eksempel en spørremodul 210, en varslingsmodul 212, et operativsystem 214 og andre én eller flere andre applikasjoner 216. Operativsystemet 214 tilveiebringer et kjøremiljø for funksjonene til én eller flere av modulene i programmet/programmene 206.

[0019] Spørremodulen 210 utfører spørringsbaserte operasjoner på informasjon samlet inn av agenten 110, så som informasjon omfattet i et logglager 218. Informasjon samlet inn av agenten 110 kan også være tilgjengelig fra arkivsamlingen 106. Spørringen(e) 220 omfatter flere spørringer, for eksempel forhåndsdefinerte spørringer. Slike forhåndsdefinerte spørringer kan vedrøre forhold vedrørende én eller flere regeldefinisjoner, så som sikkerhetsregeldefinisjoner, som kan være fastsatt for systemet 100. I et slikt scenario kan enhver eller all analyse som kan bli utført av spørremodulen 210 være i overensstemmelse med slike forhåndsdefinerte regeldefinisjoner eller forhåndsdefinerte spørringer.

[0020] Spørremodulen 210 kan begrense spørringen(e) 220 til én eller flere attributter. Slike attributter kan omfatte filnavn, applikasjonstype, eksekveringstid og annet. Når den kjører basert på en begrenset spørring, søker spørremodulen 210 gjennom all informasjon som er lagret i logglageret 218 og/eller arkivsamlingen 106 etter verdier som angir tilstedeværelse av attributten. Dersom for eksempel en person ønsker å søke i arkivsamlingen 106 etter data knyttet til en bestemt applikasjon, for eksempel en tekstbehandler, søker spørremodulen 210 etter

elementer eller hendelser knyttet til vekselvirkningene som har blitt innledet og påvirket av tekstbehandleren.

[0021] Spørringen(e) 220 kan omfatte spørringer matet inn eller programmert av én eller flere personer eller entiteter, så som en systemadministrator. For eksempel kan spørringen(e) 220 omfatte instruksjoner for å detektere alle vekselvirkninger knyttet til en gitt bruker-ID. Videre kan spørringen(e) 220 omfatte instruksjoner for å detektere alle vekselvirkninger i tilknytning til en applikasjon som kjører på én eller flere av de databaserte anordningene 102.

[0022] Med henvisning tilbake til samletjeneren 104 skal analysen av informasjonen knyttet til vekselvirkninger av én eller flere databaserte anordninger 102 med filsystemer og/eller innstillinger utføres for å bestemme virkemåten og/eller den vedholdte tilstanden til systemet 100. Spørremodulen 210 kan anvendes for å utføre analyse på informasjon som er samlet inn av agenten 110 og lagret i logglageret 218 og/eller arkivsamlingen 106. Spørremodulen 210 kan gjøre dette ved å gjøre et søk i logglageret 218 og/eller arkivsamlingen 106 med bruk av én eller flere spørringer som spesifisert i spørringen(e) 220. Resultater generert ved gjennomføring av spørringen(e) 220 viser vekselvirkningene mellom én eller flere av de databaserte anordningene 102 og filsystemene og/eller innstillingene.

[0023] Spørremodulen 210 kan instruere varslingsmodulen 212 til å sende ut en varsling for resultatene generert som et resultat av gjennomføring av spørringen(e) 220. Varslingen generert av varslingsmodulen 212 kan bli lagret i én eller flere varslinger 222 i dataene 208. Varslingene sendt ut av varslingsmodulen 212 kan også bli lagret i en ekstern database, for eksempel en ekstern lagringsanordning. Varslingsmodulen 212 kan også bli instruert av spørremodulen 210 til å kommunisere varslingene generert som et resultat av gjennomføring av spørringen(e) 220.

[0024] Spørremodulen 210 kan søke i logglageret 218 og/eller arkivsamlingen 106 for å detektere avvik i informasjon knyttet til vekselvirkninger mellom én eller flere databaserte anordninger 102 med hensyn til spørringen(e) 220 som gjennomføres av spørremodulen 210. I et slikt scenario kan deteksjon av slike avvik i forbindelse med bestemte vekselvirkninger også bli varslet av varslingsmodulen 212, og én eller flere tilhørende varslinger 222 kan bli kommunisert til personer, så som

systemadministratorer, eller datasystemer for lagring av varslinger for fremtidig henvisning.

[0025] Varslingsmodulen 212 er også i stand til å frembringe kontekstinformasjon for varslingen(e) 222. Kontekstinformasjonen kan i tillegg spesifisere innstillingene som kan være knyttet til den motsvarende vekselvirkningen. Kontekstinformasjon kan bli annotert på den eller de relevante varslingene 222 i ett eller flere trinn. For eksempel gir ett nivå statistisk informasjon vedrørende antallet maskiner som kan ha et program installert, den mest vanlige versjonen av filer, og annet. Et annet annoteringsnivå angir sammenlikning av hash-verdier for de installerte filene med et datasett som lager en fortegnelse over attributter, for eksempel programnavn, versjonsinformasjon og annet. Nok et annet annoteringsnivå kan forefinnes som kan gi kommentarer eller eventuell tilleggsinformasjon, vedrørende kjente problemer, leverandører og annet. Ytterligere annoteringsnivåer kan innlemmes som spesifiserer ytterligere attributter vedrørende én eller flere tilhørende varslinger 222. Varslingen(e) 222 kan også bli vist gjennom et grafisk grensesnitt slik at personer, for eksempel systemadministratorer, kan gå gjennom varslingen(e) 222 og iverksette passende tiltak i nødvendige tilfeller.

[0026] Spørremodulen 210 kan anvendes for deteksjon av varslede endringer som finner sted innenfor systemet 100 som følge av vekselvirkning mellom programmer som kjører på én eller flere databaserte anordninger 102 og filsystemer og/eller innstillinger. Varslede endringer omfatter endringer eller modifikasjoner av PS-tilstanden til et system som kan være et resultat av uforventet kjøring av et program, operativsystemet, programmer som anvendes for å utføre bestemte forretningsoppgaver som bokføring, og andre programmer. Slike varslede endringer av PS-tilstanden til et system blir godkjent og regulert for å hindre uønskede situasjoner så som redusert systemytelse, sikkerhetsproblemer og annet. Det skal også bemerkes at alle endringer som finner sted i PS-tilstanden ikke er varslede endringer.

[0027] Varslede endringer kan bli annotert med en identifikator og klassifisert i henhold til de tildelte annoteringene. Annotering av varslede endringer kan skje gjennom spesifisering av en klassifiseringsregel av spørremodulen 210. Basert på parametrene spesifisert i klassifiseringsregelen blir passende parametere vedrørende bestemte attributter knyttet til den varslede endringen. For eksempel

knytter spørremodulen 210 hvert sammenfall av en delstreng inneholdt i en klassifiseringsregel til navnet eller typen modifikasjon inneholdt i hver varslede endring. Klassifiseringer for de varslede endringene kan bli tildelt på grunnlag av en prioritetsverdi. I et slikt scenario vil for eksempel sammenfall av en klassifisering-delstreng med høyere prioritet gå foran klassifisering-delstrenger med lavere prioritet. Klassifisering-delstrenger med høyere prioritet bestemmer da den relevante klassifiseringen for den varslede endringen.

[0028] Varslede endringer kan klassifiseres ved å merke endringene med minst én eller flere av følgende klassifiseringer:

- **Problem**: Angir et kjent problem eller fremkommer som følge av eksistens eller fjerning av den gjeldende PS-tilstanden.
- **Install**: Angir endring i PS-tilstand som følge av en installasjon eller oppgradering.
- **Setting**: Angir endringer gjort av konfigurasjonsinnstillinger eller konfigurasjons-PS.
- **Content**: Angir nettsider, bilder, tekst og brukerdata.
- **Management Change**: Angir installasjon, patching eller konfigurasjonsendringer gjort av programmer med ansvar for systemforvaltning som kjører på systemet.
- **Unauthorized**: Angir installasjon av uautoriserte eller forbudte applikasjoner, eller konfigurasjonsendringer som omfatter forbudte verdier.
- **User Activity**: Angir endringer i PS som følge av at brukere logger seg inn eller kjører vindusapplikasjoner.
- **Noise**: Angir midlertidig eller bufret PS.
- **Unknown**: Angir uklassifisert PS.

[0029] Ytterligere annoteringer kan bli gitt for ytterligere å klassifisere varslede endringer og gjøre det mulig å skille dem fra andre endringer.

[0030] Spørremodulen 210 kan også anvendes for å fastslå status for et program som kjører på systemet 100 som autorisert eller uautorisert. Dette er basert på kravet om at bare godkjente prosesser eller programmer skal kjøre på systemet 100. Spørremodulen 210 fastslår status for et program som kjører på systemet 100 som autorisert eller uautorisert ved å sammenlikne attributter som spesifisert i spørringen(e) 220 og attributtene som definerer en spesifikk endring eller

modifikasjon i PS-tilstanden til et system. For eksempel kan spørremodulen 210 gjennomføre én eller flere spørringer 220 som spesifiserer en applikasjonstype som et uautorisert program. Resultater oppnådd etter gjennomføring av spørringen(e) 220 inneholder informasjon vedrørende endringer i PS som har funnet sted som reaksjon på kjøring av den spesifiserte applikasjonstypen. Spørremodulen 210, når den mottar resultatene, merker disse resultatene som endringer forårsaket av kjøring eller virkningen av et uautorisert program.

[0031] Spørremodulen 210 kan sammenlikne attributtene spesifisert i en forhåndsdefinert liste av godkjente og/eller ikke godkjente programmer med attributter som definerer en spesifikk endring eller modifikasjon i PS-tilstanden til et system. Listen kan i dette tilfellet inneholde et bestemt antall godkjente eller ikke godkjente programmer. Programmer som kjører på systemet 100 som er tilsvarende programmene identifisert som ikke godkjente i den forhåndsdefinerte listen, blir merket som uautoriserte programmer.

[0032] Godkjente og/eller ikke godkjente programmer spesifisert i den forhåndsdefinerte listen kan også inneholde ytterligere informasjon, så som en etikett, som beskriver karakteren til og/eller forskjellige trekk ved programmene. Eksempler på slik ytterligere informasjon omfatter etiketter, så som "godkjent", "type", "kategori", "funksjon", "produktinformasjon", "tilvirkerinformasjon" og "produktbeskrivelse". For eksempel blir programmer som er merket som "godkjente" betraktet som autoriserte programmer for kjøring på én eller flere av de databaserte anordningene 102 i systemet 100; og en "kategori"-etikett spesifiserer den tiltenkte bruken av programmet.

[0033] Endringer eller modifikasjoner utført av et program for første gang er som default ikke godkjent og merket som "uautorisert". For eksempel, når endringer eller modifikasjoner av et program blir detektert for første gang, merker spørremodulen 210 programmet og og dets tilhørende vekselvirkninger som "uautoriserte". Programmer som har blitt merket som "uautoriserte" kan bli varslet av varslingsmodulen 212 for undersøkelse, for eksempel av en systemadministrator, for gjennomføring av diagnose om nødvendig eller for en forestående godkjennelse. Dersom godkjennelse blir gitt, blir det godkjente programmet videre gitt en passende etikett som beskriver programmet, og kan også bli lagt til i den forhåndsdefinerte listen som inneholder de godkjente og/eller ikke godkjente programmene.

[0034] Spørremodulen 210 kan også detektere utvidelsespunkter (EP – Extensibility Points). Utvidelsespunkter er vekselvirkninger som angir dynamisk lasting og eksekvering av instruksjoner i et program eller et operativsystem som kjører på én eller flere av de databaserte anordningene 102. Når for eksempel et første program, så som en tekstbehandler, en regnearkapplikasjon eller annet som kjører på én eller flere av de databaserte anordningene 102 starter opp, kan det første programmet også trigge instruksjoner i andre programmer, så som "add-on"-programmer, som tilveiebringer ytterligere funksjonalitet for det første programmet. På denne måten kan kjøringen av det første programmet generere forskjellige vekselvirkninger, omfattende vekselvirkninger mellom det første programmet og filsystemet, og vekselvirkninger mellom de andre programmene gir ytterligere funksjonalitet i kjøringen av det første programmet og filsystemet. Slik informasjon kan gi innsikt i virkemåten til systemet der det første programmet er installert og også gjøre det mulig å vurdere hvilken innvirkning slike installasjoner vil kunne ha i systemet.

[0035] Informasjonen knyttet til forskjellige vekselvirkninger generert som et resultat av kjøringen av det første programmet kan bli snappet opp og kopiert for eksempel av agenten 110. Hendelsesinformasjon forbundet med de forskjellige vekselvirkningene kan bli lagret som komprimerte logger i logglageret 218 og/eller arkivsamlingen 106. Selv om hendelsesinformasjonen blir lagret i komprimert lager, må det forstås at komprimert lagring ikke nødvendigvis blir anvendt; imidlertid vil bruk av komprimert lagring gjøre systemet mer skalerbart ved at lagringen krever mindre plass. Den lagrede hendelsesinformasjonen kan bli undersøkt av entiteter, så som en systemadministrator, eller av spørremodulen 210 for å detektere vekselvirkninger i forbindelse med det første programmet og de andre programmene med filsystemet. På denne måten kan de andre programmene bli detektert dersom de er knyttet til kjøringen av det første programmet.

[0036] Spørremodulen 210 kan også anvendes for å detektere direkte EP-er for det første programmet. For eksempel kan spørremodulen 210 detektere direkte EP-er ved å isolere vekselvirkninger som både (1) vedrører forskjellige programmer lastet i systemminne for kjøring før kjøring av det første programmet, og (2) refererer til det første programmet eller er knyttet til kjøringen av det første programmet.

[0037] I ett utførelseseksempel kan spørremodulen 210 identifisere mulige

direkte EP-er for det første programmet ved å spørre logglageret 218 og/eller arkivsamlingen 106 om vekselvirkninger vedrørende forskjellige programmer lastet i systemminne for kjøring før kjøring av det første programmet. For eksempel kan spørremodulen 210 spørre etter vekselvirkninger vedrørende forskjellige programmer lastet i systemminne for kjøring innenfor en gitt tidsperiode, for eksempel 1 sekund, før kjøring av det første programmet. Spørremodulen 210 kan identifisere direkte EP-er for det første programmet fra de mulige EP-ene ved å spørre de mulige EP-ene om vekselvirkninger som refererer til det første programmet eller er knyttet til kjøringen av det første programmet. Direkte EP-er kan bli lagret i andre data 224.

[0038] Spørremodulen 210 kan også anvendes for å detektere indirekte EP-er. For eksempel, tilbake til eksempelet med det første programmet over, kan spørremodulen 210 spørre logglageret 218 og/eller arkivsamlingen 106 om vekselvirkninger som refererer til, eller som er knyttet til, de direkte EP-ene. Slike vekselvirkninger kan kalles indirekte EP-er. Indirekte EP-er kan bli lagret i andre data 224.

[0039] Spørremodulen 210 kan også anvendes for å detektere tilstedeværelse av ondsinnede programvareapplikasjoner gjennom overvåkning av direkte EP-er. Ondsinnede programvareapplikasjoner kan omfatte "spyware", "Trojanske hester", "ormer", "viruser", etc. som under normale omstendigheter ikke vil være knyttet til noe program. For eksempel kan spørremodulen 210 sammenlikne EP-er for et program som kjører på én eller flere av de databaserte anordningene 102 mot kontroll-EP-er for det samme programmet funnet når programmet kjørte på de databaserte anordningene 102 i fravær av ondsinnet programvare. Forskjeller mellom EP-ene og kontroll-EP-ene kan bli undersøkt av entiteter så som spørremodulen 210 og/eller en systemadministrator for å fastslå om forskjellene antyder tilstedeværelse av ondsinnet programvare som kjører i forbindelse med programmet. Ondsinnet programvare funnet ved hjelp av EP-er kan bli fjernet fra den berørte databaserte anordningen 102 av spørremodulen 210, systemadministratoren eller andre.

[0040] I en annen utførelse kan spørremodulen 210 anvendes for å generere én eller flere varslinger 222 ved varslingsmodulen 212 som reaksjon på deteksjon av EP-er. I nok en annen utførelse kan varslingen(e) 222 som blir generert videre bli vist

eller hentet frem gjennom et grafisk grensesnitt som letter gjennomgang av varslingen(e) 222 av en person, for eksempel en systemadministrator, for nærmere analyse eller for gjennomføring av nødvendig diagnose.

[0041] Figur 3 illustrerer et eksempel på grafisk grensesnitt 300 som viser den eller de genererte varslingene 222 i én av de mulige utførelsene. I den illustrerte utførelsen viser det grafiske grensesnittet 300 nedlastinger som er utført av et første program (f.eks. en web-nettleser) under dets kjøring. Det grafiske grensesnittet 300 viser i feltene 302 og 304 en liste av programmer som ble lastet ned av det første programmet (i det viste eksempelet i figur 3, spesielt "MSN Search Toolbar" og "Winamp Media Player") under dets kjøring. I illustrasjonen kan det i feltet 306 sees at nedlastingen av programmene som vist i feltene 302, 304 også resulterer i opprettelse av programfiler svarende til andre programmer enn det første programmet og programmene illustrert i feltene 302, 304. En visuell representasjon i form av det grafiske grensesnittet 300 gir således en liste av programmer som utilsiktet blir installert på én eller flere av de databaserte anordningene 102 i systemet 100 under kjøring, nedlasting og/eller installasjon av det første programmet.

[0042] Feltet 306 kan også angi innvirkningene eller modifikasjonene i PS-tilstanden til systemet 100 som følge av installasjon eller kjøring av andre programmer enn det første programmet. Videre kjøring av ett eller flere andre programmer enn det første programmet kan være avhengig av kjøringen av det første programmet. For eksempel kan som illustrert feltet "MSN Search Toolbar" bli aktivert ved kjøring av programfiler tilknyttet det første programmet. Bestemmelse av denne effekten kan oppnås ved å detektere EP-ene svarende til det første programmet. Ved å overvåke EP-ene til det første programmet kan instanser av programkjøring av andre programmer som avhenger av kjøringen av det første programmet bli detektert, og korrigeringstiltak kan bli iverksatt om nødvendig.

[0043] Instanser av kjøring av andre programmer som er avhengig av kjøringen av det første programmet kan også bli vist gjennom et annet grafisk grensesnitt 400, som illustrert i figur 4. Figur 4 viser kjøring av et første program vist som segment 402, for eksempel "iexplorer.exe", trigger kjøring av Winamp vist som segmentet 404, som igjen kjører "emusic.exe" vist som segment 406. Fra denne illustrasjonen kan deteksjon av EP-er knyttet til det første programmet realiseres i detalj og gjennom

visuelle innretninger.

[0044] Spørremodulen 210 kan anvendes for å detektere lekket PS. Lekkede filer omfatter filer eller registerinnstillinger som er igjen på et system, så som systemet 100, etter at et program som opprettet filene eller registerinnstillingene har blitt avinstallert. De kan også omfatte filer eller innstillinger som kan ha blitt opprettet som et resultat av en installasjon, for eksempel midlertidige filer, men ikke har blitt slettet etter avslutning av installasjonsprosessen. I tillegg finnes det en annen klasse av lekket PS-tilstand, for eksempel PS-tilstand som blir generert ved kjøretid for programmet (dvs. etter installasjon). Eksempler på disse er tilstand som kan være generert ved første bruk eller utvidelser av programmet som er installert separat etter den innledende installasjonen.

[0045] For å detektere lekkede filer katalogiserer spørremodulen 210 installasjonsfiler og innstillingsendringer i forbindelse med hvert program lastet inn på systemet 100, som kan spores ved hjelp av programmer og ved innledende installasjon. Senere, dersom programmet blir avinstallert, kan en tilhørende katalog over installasjonsfiler og konfigurasjons- eller registerinnstillinger for programmet bli hentet frem, og systemet 100 kan bli sjekket for å sikre at alle installasjonsfiler og registerinnstillinger har blitt fjernet eller tilbakestilt. For å detektere lekkede filer på en databasert anordning 102 katalogiserer spørremodulen 210 installasjonsfiler ved å kjøre et søk gjennom én eller flere databaserte anordninger 102 for å detektere alle programmer, for eksempel applikasjoner, som er installert på de databaserte anordningene 102.

[0046] Spørremodulen 210 kan også frembringe en liste over alle programmer registrert i en installer-database i operativsystemer på den ene eller de flere databaserte anordningene 102. Eksempler på installer-databaser omfatter komponenter som genererer en besatt liste av programmer installert på en databasert anordning som betraktes.

[0047] Spørremodulen 210 spør logglageret 218 og/eller arkivsamlingen 106 om informasjon om registerkonfigurasjoner eller innstillinger og for å danne en liste av programmer registrert hos operativsystemene på de databaserte anordningene 102. Spørremodulen 210 kan da søke i logglageret 218 og/eller arkivsamlingen 106 for å listeføre filene og registeroppføringene, som kan bli generalisert til all PS-tilstand, for alle programmer installert på den databaserte anordningen 102. For å

listeføre filene og registeroppføringene kan spørremodulen 210 spørre etter alle filer og registeroppføringer som svarer til én eller flere attributter, for eksempel programidentifikatorer, for programmene installert på den databaserte anordningen 102.

[0048] Dersom en fil eller innstilling på den databaserte anordningen 102 ikke er omfattet i filene og registeroppføringene svarende til programidentifikatorene for programmene installert på den databaserte anordningen 102, kan spørremodulen 210 konkludere at filen eller innstillingen er en lekket fil. Lekkede filer kan bli fjernet av spørremodulen 210 eller forskjellige andre programmer, omfattende operativsystemet, en systemadministrator eller annet.

[0049] De lekkede filene (PS) som oppdages kan bli vist gjennom et grafisk grensesnitt slik at personer, for eksempel en systemadministrator, kan gå gjennom de lekkede filene og iverksette passende tiltak for nødvendige tilfeller. Videre kan de viste lekkede filene (PS) og tilhørende informasjon bli lagret på et eksternt lagringssted, for eksempel en ekstern database, for fremtidig henvisning. Listen over lekket PS kan anvendes for automatisk fjerning av den lekkede tilstanden av systemer når hovedapplikasjonen fjernes. Denne listen av lekket PS kan også anvendes for å knytte hver PS i systemet til en eierapplikasjon.

[0050] Spørremodulen 210 kan detektere uaktuelle prosesser som følge av endrede filer, innstillinger eller foreldede moduler, omfattende vanlige feilkonfigurasjoner, gamle programvareversjoner og annet. Uaktuelle prosesser oppstår for eksempel når programvareoppgraderinger ikke er i stand til å starte berørte prosesser på nytt etter utskifting av disklagrede eksekverbare filer, programfiler eller innstillinger. Som følge av dette vil den databaserte anordningen der den uaktuelle prosessen kjører ikke ta hensyn til oppgraderingen og kjøre videre basert på de gamle eksekverbare filene, programfilene eller innstillingene.

[0051] For å oppdage uaktuelle prosesser kan spørremodulen 210 spørre om informasjon knyttet til vekselvirkninger av programmer lagret i logglageret 218 og/eller arkivsamlingen 106. Spørremodulen 210 spør logglageret 218 og/eller arkivsamlingen 106 om tidspunktet for siste innlasting av programmer installert på én eller flere av de databaserte anordningene 102. Spørremodulen 210 kan også spørre logglageret 218 og/eller arkivsamlingen 106 om tidspunktet for siste innlasting av filer eller registerinnstillinger knyttet til den installerte programvaren. I ett

utførelseseksempel spør spørremodulen 210 logglageret 218 og/eller arkivsamlingen 106 om tidspunktet for siste innlasting av filer eller registerinnstillinger knyttet til de tilhørende dynamiske linkbibliotekene (DLL-ene) installert med programvaren. Spørremodulen 210 kan også spørre logglageret 218 og/eller arkivsamlingen 106 om tidspunkt eller dato for siste modifisering av programvaren installert på den databaserte anordningen. Slike modifikasjoner omfatter for eksempel aksess gjort til én eller flere filer eller programinnstillinger tilknyttet den siste kjente versjonen av den installerte programvaren.

[0052] Dersom tidspunktet for siste innlasting av programvaren er senere enn tidspunktet eller datoen for den siste kjente modifikasjonen av programvaren, kan det forekomme inkonsistenser som følge av at programvaren ikke anvender den siste lastede oppdateringen. Slike inkonsistenser, dersom de detekteres av spørremodulen 210, kan bli varslet og korrigert av en person, for eksempel en systemadministrator.

[0053] De detekterte foreldede filene kan bli vist gjennom et grafisk grensesnitt slik at personer, for eksempel systemadministratorer, kan undersøke de foreldede filene og iverksette passende tiltak i nødvendige tilfeller. De viste foreldede filene og tilhørende informasjon kan bli lagret på et eksternt lagringssted, for eksempel en ekstern database, for fremtidig henvisning.

[0054] Spørremodulen 210 kan detektere forekomster av kjente uønskede programmer, omfattende programvareapplikasjoner som "malware, "spyware", "trojanske hester", "viruser", etc. For å gjøre dette kan spørremodulen 210 spørre og søke i logglageret 218 og/eller arkivsamlingen 106 etter programmer lastet for kjøring i minnet på én eller flere av de databaserte anordningene 102. Programmene lastet for kjøring i minnet kan da bli sammenliknet, for eksempel av spørremodulen 210, mot en liste av kjente uønskede programmer.

[0055] For eksempel kan spørremodulen 210 detektere forekomster av programmer lastet for kjøring i minne på en databasert anordning 102 på grunnlag av identifikatorer, så som programidentifikatorer, for programmene. Spørremodulen 210 kan da sammenlikne identifikatorene for programmene lastet for kjøring i minne på den databaserte anordningen 102 mot en liste av identifikatorer, så som programidentifikatorer, for kjente uønskede programmer. Dersom en identifikator for et program lastet for kjøring i minne sammenfaller med en identifikator for et kjent

uønsket program, kan spørremodulen 210 bevirke fjerning av programmet lastet for kjøring i minne fra den databaserte anordningen 102. I én mulig utførelse kan listen av identifikatorer for kjente uønskede programmer være lagt inn, i hvert fall delvis, av en systemadministrator.

[0056] De uønskede programmene detektert av spørremodulen 210 kan bli vist gjennom et grafisk grensesnitt slik at personer, for eksempel en systemadministrator, kan undersøke de uønskede programmene og iverksette passende tiltak for å fjerne dem. De viste uønskede programmene og tilhørende informasjon kan bli lagret på et eksternt lagringssted, for eksempel en ekstern database, for fremtidig henvisning for å detektere de samme eller tilsvarende uønskede programmer.

[0057] Et uidentifisert program på én eller flere databaserte anordninger 102 som ikke har en tilhørenede identifikator kan bli detektert av spørremodulen 210 og rapportert til en systemadministrator for bestemmelse av om det uidentifiserte programmet er et uønsket program eller ikke. Systemadministratoren kan undersøke karakteren til det uidentifiserte programmet ved å gå gjennom listen av uidentifiserte programmer, i form av en rapport. Gjennomgang av en systemadministrator kan omfatte undersøkelse av hensikten med det uidentifiserte programmet, avhengigheten til det uidentifiserte programmet av andre programmer, og bestemmelse av hvorvidt det uidentifiserte programmet er uønsket. I tillegg kan systemadministratoren sjekke tidligere erfaring med programmer med tilsvarende trekk som de til det uidentifiserte programmet for å avgjøre om det uidentifiserte programmet er uønsket.

[0058] Dersom systemadministratoren bestemmer at det uidentifiserte programmet er uønsket, kan systemadministratoren bevirke fjerning av det uidentifiserte programmet fra de databaserte anordningene 102. For eksempel kan systemadministratoren fjerne det uidentifiserte programmet selv, eller systemadministratoren kan instruere elementer i de databaserte anordningene 102 til å fjerne det uidentifiserte programmet.

[0059] I tillegg kan systemadministratoren på grunnlag av en generert rapport eller én eller flere av varslingen(e) 222 tilordne en identifikator, for eksempel en program-ID, til det uidentifiserte programmet, og innlemme identifikatoren i en liste over uønskede programmer. På denne måten, dersom det uidentifiserte programmet igjen dukker opp på en databasert anordning 102, kan det raskt bli identifiseretsom

et uønsket program på grunnlag av den tilhørende identifikatoren. Videre kan fjerningen av det uidentifiserte programmet bli utført av elementer i den databaserte anordningen 102, agenten 110 og andre.

[0060] De uidentifiserte programmene og deres tilhørende prosesser detektert av spørremodulen 210 kan bli vist gjennom et grafisk grensesnitt slik at personer, for eksempel systemadministratorer, kan undersøke de uidentifiserte programmene og iverksette passende tiltak for å fjerne dem. De viste uidentifiserte programmene og tilhørende informasjon kan bli lagret på et eksternt lagringssted, for eksempel en ekstern database, for fremtidig henvisning for å detektere de samme eller tilsvarende uønskede programmer. Videre kan uønskede endringer også bli identifisert og/eller sporet.

[0061] Spørremodulen 210 kan hindre kopiering av filer til nettverksstasjoner eller flyttbare lagringssteder ved å avvise skriveoperasjoner til slike lagringssteder for et program som kjører på én eller flere av de databaserte anordningene 102. Spørremodulen 210 kan også undersøke tidligere slike avvisninger av skriveoperasjoner gjort for revisjonsformål for å hindre slike skriveoperasjoner i fremtiden.

Eksempler på fremgangsmåter

[0062] Eksempler på fremgangsmåter for å fange opp og analysere eksekveringstråder er beskrevet i forbindelse med figurene 1 til 4. Disse fremgangsmåteeksemplene kan beskrives i den generelle sammenhengen datamaskin-eksekverbare instruksjoner. Generelt kan datamaskin-eksekverbare instruksjoner omfatte rutiner, programmer, objekter, komponenter, datastrukturer, prosedyrer, moduler, funksjoner og liknende som utfører bestemte funksjoner eller implementerer bestemte abstrakte datatyper. Fremgangsmåtene kan også praktiseres i et distribuert databehandlingsmiljø der funksjoner blir utført av fjerne prosesseringsanordninger som er sammenkoblet gjennom et kommunikasjonsnettverk. I et distribuert databehandlingsmiljø kan datamaskin-eksekverbare instruksjoner befinne seg i både lokale og fjerne datalagringsmedier, omfattende minnelagringsanordninger.

[0063] Figur 5 illustrerer et eksempel på fremgangsmåte 500 for å fange opp og samle inn informasjon som er knyttet til vekselvirkninger mellom programmer som kjører på én eller flere databaserte anordninger 102 og/eller filsystemer og

innstillinger. Rekkefølgen fremgangsmåtetrinnene er beskrevet er ikke ment som en begrensning, og et hvilket som helst antall av de beskrevne fremgangsmåtetrinnene kan bli kombinert i en hvilken som helst rekkefølge for å gjennomføre fremgangsmåten, eller en alternativ fremgangsmåte. Videre kan enkelttrinn utelates fra fremgangsmåten uten at en fjerner seg fra idéen og rammen til oppfinnelsen beskrevet her. I tillegg kan fremgangsmåten bli utført i en hvilken som helst passende maskinvare, programvare, fastvare eller kombinasjon av dette.

[0064] I trinn 502 blir informasjon eller data knyttet til et program som kjører eller eksekverer på et system fanget opp. I én utførelse blir nevnte informasjon innhentet når et program/en prosess anroper en instrumentert funksjon som inneholder modifisert funksjonskode. For eksempel kan de datamaskinlesbare instruksjonene være modifisert til å instruere den ene eller de flere funksjonene til å fange opp data knyttet til vekselvirkninger mellom programmer som kjører på én eller flere databaserte anordninger 102 og/eller filsystemer og innstillinger. I en utførelse anvender en virtuell maskin datainnsamlingslogikk direkte når den tolker opprinnelig kode som eksekveres. Denne metoden vil ikke kreve modifikasjon av den opprinnelige koden. Likeledes vil dette kunne implementeres direkte i prosessormaskinvare.

[0065] En agent, så som agenten 110, kan instrumentere én eller flere funksjoner i systemet 100. Den ene eller de flere funksjonene kan instrumenteres ved å modifisere datamaskinlesbare instruksjoner knyttet til den ene eller de flere funksjonene.

[0066] En agent 110, så som tråddata-registrereren, kan fange opp tråder som anroper modifiserte funksjoner i systemet 100. Programmer som trådene er tilknyttet kan kjøre i ett av flere kjørelag, så som et programlag, et mellomvarelag, et operativsystemlag og annet. Et filsystem som programmet kan forsøke å vekselvirke med kan omfatte filer (så som datafiler, eksekverbare filer) og innstillingsinformasjon (så som konfigurasjonsinnstillinger eller registerinnstillinger) og annet.

[0067] I trinn 504 blir forskjellig informasjon eller data knyttet til eksekvering av programmer som kjører på én eller flere av de databaserte anordningene 102 samlet inn eller kopiert til et lagringssted. Informasjon knyttet til vekselvirkninger av programmer med filsystemer og/eller innstillinger omfattende vekselvirkninger fremkalt av modifiserte funksjoner, blir kopiert og sendt til et lagringssted. For

eksempel kan agenten 110 kopiere alle eller utvalgte data knyttet til vekselvirkningene og lagre dataene på et lagringssted, så som samletjeneren 104. Data knyttet til vekselvirkningene kan omfatte informasjon vedrørende vekselvirkninger fremkalt av den instrumenterte funksjonen.

[0068] I trinn 506 blir data lagret på lagringsstedet komprimert. I én utførelse kan de komprimerte dataene bli lagret på et annet lagringssted. For eksempel kan de komprimerte dataene bli lagret i logglageret 218 i samletjeneren 104 og/eller i arkivsamlingen 106.

[0069] I trinn 508 blir de komprimerte dataene lastet opp periodisk for analyse. De komprimerte dataene kan bli lastet opp til en samletjener 106 eller til et lagringssted som fungerer som en samletjener 106. Periodisiteten i opplastingen av de komprimerte dataene for analyse kan varieres. I én utførelse blir de komprimerte dataene lastet opp etter spesifiserte tidsintervaller. I en annen utførelse kan de komprimerte dataene bli lastet opp når de komprimerte dataene overstiger en forhåndsdefinert terskellagringsplass.

[0070] Figur 6 illustrerer et eksempel på fremgangsmåte 600 for å klassifisere varslede endringer. Varslede endringer omfatter endringer eller modifikasjoner som kan finne sted som følge av uforventet kjøring av et program, et operativsystem, programmer som anvendes for å utføre spesifikkbestemte forretningoppgaver som bokføring, og andre programmer. Rekkefølgen fremgangsmåtetrinnene er beskrevet er ikke ment å forstås som en begrensning, og et hvilket som helst antall av de beskrevne fremgangsmåtetrinnene kan bli kombinert i en hvilken som helst rekkefølge for å utføre fremgangsmåten, eller en alternativ fremgangsmåte. Videre kan enkelttrinn utelates fra fremgangsmåten uten at en fjerner seg fra idéen og rammen til oppfinnelsen beskrevet her. I tillegg kan fremgangsmåten bli utført i en hvilken som helst passende maskinvare, programvare, fastvare eller kombinasjon av dette.

[0071] I trinn 602 spesifiseres en klassifiseringsregel beskrevet av forskjellige parameterverdier. For eksempel kan klassifiseringsregelen bli spesifisert av spørremodulen 210 sammen med parameterverdiene som definerer klassifiseringsregelen.

[0072] I trinn 604 blir parametere som definerer klassifiseringsregelen knyttet til den ene eller de flere attributtene som definerer karakteren og trekkene til en varslet

endring. Spørremodulen 210 knytter parametrene som definerer klassifiseringsregelen til attributtene som karakteriserer den varslede endringen. Tilknytning av én eller flere parameterverdier til attributtene som karakteriserer den aktuelle varslede endringen resulterer i et sett av sannsynlige klassifiseringer. For eksempel knytter spørremodulen 210 hvert sammenfall av en delstreng inneholdt i en klassifiseringsregel med PS-navn inneholdt i hver varslede endring.

[0073] I trinn 606 blir én eller flere av de sannsynlige klassifiseringene tilordnet en prioritetsverdi. Spørremodulen 210 kan tilordne en prioritetsverdi til én eller flere av de sannsynlige klassifiseringene. For eksempel vil en spesifikk varslet endring som opptrer over en lenger tidsperiode bli tilordnet en høyere prioritetsverdi.

[0074] I trinn 608 blir den sannsynlige klassifiseringen med høyest prioritetsverdi tildelt den aktuelle varslede endringen. I én utførelse bestemmer spørremodulen 210 den høyeste prioritetsverdien tildelt til de sannsynlige klassifiseringene og tilordner denne klassifiseringen til den aktuelle varslede endringen.

[0075] Figur 7 illustrerer et eksempel på fremgangsmåte 700 for å hindre gjennomføring av uautoriserte vekselvirkninger -- som definert av en systemadministrator, for eksempel – på én eller flere databaserte anordninger 102. Eksempler på uautoriserte vekselvirkninger omfatter lese- og/eller skriveoperasjoner utført i et filsystem av en entitet eller et program som ikke har tillatelse til å utføre denne operasjonen.

[0076] Rekkefølgen fremgangsmåtetrinnene er beskrevet er ikke ment å forstås som en begrensning, og et hvilket som helst antall av de beskrevne fremgangsmåtetrinnene kan bli kombinert i en hvilken som helst rekkefølge for å gjennomføre fremgangsmåten, eller en alternativ fremgangsmåte. Videre kan enkelttrinn utelates fra fremgangsmåten uten at en fjerner seg fra idéen og rammen til oppfinnelsen beskrevet her. I tillegg kan fremgangsmåten bli utført i en hvilken som helst passende maskinvare, programvare, fastvare eller kombinasjon av dette.

[0077] I trinn 702 mottas informasjon for et program som kjører på et system. Informasjonen er knyttet til vekselvirkningene mellom programmet og filsystemer og/eller konfigurasjonsinnstillinger. Spørremodulen 210 kan spørre logglageret 218 og/eller arkivsamlingen 106 om informasjon vedrørende vekselvirkningene utført av et program på én eller flere av de databaserte anordningene 102. Informasjonen

oppnådd gjennom spørringen er kjennetegnet ved én eller flere attributter.

[0078] I trinn 704 blir attributter for programmet som kjører på systemet sammenliknet med attributter for flere godkjente og ikke godkjente programmer/prosesser innlemmet i en forhåndsdefinert liste. For eksempel sammenlikner spørremodulen 210 attributtene, for eksempel programtype, for programmet med attributtene for programmene inneholdt i den forhåndsdefinerte listen.

[0079] I trinn 706 blir det bestemt hvorvidt attributtene svarer til de til ikke godkjente programmer/prosesser eller vekselvirkninger. Dersom for eksempel attributten for programmet som kjører på systemet 100 svarer til en attributt forbundet med ikke godkjente vekselvirkninger (dvs. 'ja'-grenen fra trinn 706), tillates ikke vekselvirkningene knyttet til programmet å fortsette (dvs. trinn 708). Alternativt, dersom attributten for programmet som kjører på systemet 100 ikke svarer til en attributt forbundet med ikke godkjente vekselvirkninger (dvs. 'nei'-grenen fra trinn 706), tillates vekselvirkningene knyttet til programmene å fortsette (dvs. trinn 710).

[0080] Figur 8 illustrerer et eksempel på fremgangsmåte 800 for å detektere ett eller flere utvidelsespunkter (EP-er) for et program installert på én eller flere databaserte anordninger 102. En EP omfatter vekselvirkninger som styrer dynamisk lasting og kjøring av en datamaskinapplikasjon. Rekkefølgen fremgangsmåtetrinnene er beskrevet er ikke ment å forstås som en begrensning, og et hvilket som helst antall av de beskrevne fremgangsmåtetrinnene kan bli kombinert i en hvilken som helst rekkefølge for å utføre fremgangsmåten, eller en alternativ fremgangsmåte. Videre kan enkelttrinn utelates fra fremgangsmåten uten at en fjerner seg fra idéen og rammen til oppfinnelsen beskrevet her. I tillegg kan fremgangsmåten bli utført i en hvilken som helst passende maskinvare, programvare, fastvare eller kombinasjon av dette.

[0081] I trinn 802 sjekkes tidligere vekselvirkninger (dvs. vekselvirkninger vedrørende forskjellige programmer lastet for kjøring i et systemminne før kjøring av et første program). For eksempel kan spørremodulen 210 identifisere potensielle direkte utvidelsespunkter for et første program ved å spørre logglageret 218 og/eller arkivsamlingen 106 etter vekselvirkninger vedrørende forskjellige programmer lastet inn i minne på én eller flere av de databaserte anordningene 102 for kjøring før kjøring av det første programmet. Spørremodulen 210 kan spørre etter

vekselvirkninger vedrørende forskjellige programmer lastet i minne for kjøring innenfor en gitt tidsperiode, for eksempel to sekunder, før kjøring av det første programmet.

[0082] I trinn 804 blir det gjort en sjekk for å finne tidligere vekselvirkninger som refererer til filnavnet til det første programmet lastet for kjøring i systemminne på en databasert anordning. For eksempel kan spørremodulen 210 spørre etter vekselvirkninger knyttet til forskjellige programmer som refererer til det første programmet eller som er knyttet til kjøring av det første programmet på de databaserte anordningene 102. Spørremodulen 210 kan spørre etter vekselvirkninger med forskjellige attributter, så som filnavnet til det første programmet, en programidentifikator for det første programmet, og annet.

[0083] I trinn 806 blir de tidligere vekselvirkningene som refererer til filnavnet til det første programmet flagget som direkte EP-er. For eksempel kan spørremodulen 210 identifisere direkte EP-er for det første programmet ved å spørre etter alle tidligere vekselvirkninger som refererer til det første programmet eller som er knyttet til kjøring av det første programmet.

[0084] Figur 9 illustrerer et eksempel på fremgangsmåte 900 for å detektere lekkede elementer som har blitt værende igjen etter avinstallering av et program fra én eller flere databaserte anordninger 102. Rekkefølgen fremgangsmåtetrinnene er beskrevet er ikke ment å forstås som en begrensning, og et hvilket som helst antall av de beskrevne fremgangsmåtetrinnene kan bli kombinert i en hvilken som helst rekkefølge for å utføre fremgangsmåten, eller en alternativ fremgangsmåte. Videre kan enkelttrinn utelates fra fremgangsmåten uten at en fjerner seg fra idéen og rammen til oppfinnelsen beskrevet her. I tillegg kan fremgangsmåten bli utført i en hvilken som helst passende maskinvare, programvare, fastvare eller kombinasjon av dette.

[0085] I trinn 902 blir installasjonsfiler og innstillingsendringer for hvert program lastet inn på en databasert anordning og/eller et system katalogisert og listeført. Listeføring omfatter opprettelse av en liste av programmer registrert på et operativsystem på den databaserte anordningen.

[0086] For eksempel kan systemet 100 bli gjennomsøkt for å detektere alle programmer som er installert på databaserte anordninger i systemet og alle operativsystem-installasjonsfiler knyttet til programmene på de databaserte

anordningene. Alle programmer installert på de databaserte anordningene og/eller alle operativsystem-installasjonsfiler tilknyttet programmene kan bli listeført ved at de legges inn i en liste.

[0087] Spørremodulen 210 kjører et søk gjennom systemet 100 for å detektere alle programmer som er installert på én eller flere databaserte anordninger 102 for å katalogisere og listeføre alle operativsystem-installasjonsfiler på databaserte anordninger 102. For eksempel kan spørremodulen 210 spørre logglageret 218 og/eller arkivsamlingen 106 om alle programmer registrert på operativsystemet på én eller flere av de databaserte anordningene 102 i systemet 100. Programmene som blir funnet kan bli katalogisert og listeført av en rekke forskjellige anordninger, så som spørremodulen 210, agenten 110 og andre. Videre kan alle operativsystem-installasjonsfiler på de databaserte anordningene 102 for programmene funnet bli katalogisert og listeført av en rekke forskjellige anordninger, så som spørremodulen 210, agenten 110 og andre.

[0088] I trinn 904 listeføres vedholdt tilstand (PS) på en databasert anordning og/eller et system, omfattende filer og registerinnstillinger for avinstallerte programmer. Dette kan omfatte søking gjennom minne på den databaserte anordningen og/eller systemet etter filer og registerinnstillinger for alle programmer som har blitt installert på den databaserte anordningen og/eller systemet, omfattende filer og registerinnstillinger for programmer som har blitt avinstallert. For eksempel kan spørremodulen 210 spørre logglageret 218 og/eller arkivsamlingen 106 for å finne alle filer og registerinnstillinger som svarer til identifikatorer, så som programidentifikatorer, for alle programmer som har blitt installert på de databaserte anordningene 102.

[0089] I trinn 906 blir filene og registerinnstillingene for programmer registrert hos operativsystemet sammenliknet med filene og registerinnstillingene for programmer som er installert på de databaserte anordningene 102 og/eller systemet 100. For eksempel kan spørremodulen 210 sammenlikne identifikatorer, så som programidentifikatorer, for de listeførte filene og registerinnstillingene for programmene registrert hos operativsystemet på de databaserte anordningene 102 med identifikatorer for filer eller innstillinger for alle programmer som har blitt installert på de databaserte anordningene 102.

[0090] I trinn 908 kan filer og registerinnstillinger for programmer på begge

listene bli utelatt fra betraktning. De gjenværende filer og registerinnstillinger, som representerer filer og registerinnstillinger svarende til programmer som har blitt avinstallert fra den databaserte anordningen 102 og/eller systemet 100, bli merket som lekkede filer, og kan bli fjernet fra den databaserte anordningen 102 og/eller systemet 100. For eksempel kan spørremodulen 210 korrelere identifikatorer, så som programidentifikatorer, for filer og registerinnstillinger i tilknytning til programmer registrert hos operativsystemer på de databaserte anordningene 102 med identifikatorer for filer og registerinnstillinger for programmer som har blitt installert på de databaserte anordningene 102. Filene og registerinnstillingene knyttet til de ukorrelerte programmene kan betraktes som lekkede filer av spørremodulen 210, og kan bli fjernet fra de databaserte anordningene 102 av elementer så som spørremodulen 210, agenten 110 og andre.

[0091] Figur 10 illustrerer et eksempel på fremgangsmåte 1000 for å detektere foreldede filer omfattende vanlige feilkonfigurasjoner, gamle programvareversjoner, etc., installert på én eller flere databaserte anordninger 102. Foreldede filer oppstår for eksempel når programvareoppgraderinger ikke er i stand til å starte berørte prosesser på nytt etter utskiftning av disklagrede eksekverbare filer. Som følge av dette vil de databaserte anordningene 102 der den foreldede filen befinner seg ikke ta hensyn til oppgraderingen og fortsette å kjøre programmet fra den gamle filen. Rekkefølgen fremgangsmåtetrinnene er beskrevet er ikke ment å forstås som en begrensning, og et hvilket som helst antall av de beskrevne fremgangsmåtetrinnene kan bli kombinert i en hvilken som helst rekkefølge for å utføre fremgangsmåten, eller en alternativ fremgangsmåte. Videre kan enkelttrinn utelates fra fremgangsmåten uten at en fjerner seg fra idéen og rammen til oppfinnelsen beskrevet her. I tillegg kan fremgangsmåten bli utført i en hvilken som helst passende maskinvare, programvare, fastvare eller kombinasjon av dette.

[0092] I trinn 1002 blir programmer lastet inn på en databasert anordning og/eller et system katalogisert og listeført. I én utførelse omfatter listeføring opprettelse av en liste av programmer registrert hos et operativsystem på den databaserte anordningen. For eksempel kan et system bli avsøkt for å detektere alle programmer som er installert på databaserte anordninger i systemet. Alle programmer installert på de databaserte anordningene kan bli listeføres ved at de legges inn i en liste.

[0093] I én mulig utførelse kjører spørremodulen 210 et søk gjennom systemet 100 for å detektere alle programmer som er registrert på én eller flere databaserte anordninger 102 for å katalogisere og listeføre alle programmer registrert hos operativsystemet på de databaserte anordningene 102. For eksempel kan spørremodulen 210 spørre logglageret 218 og/eller arkivsamlingen 106 etter alle programmer registrert i operativsystemet på én eller flere av de databaserte anordningene 102 i systemet 100. Programmene som blir funnet kan bli katalogisert og listeført av en rekke forskjellige anordninger og/eller entiteter, så som spørremodulen 210, agenten 110 og andre.

[0094] I trinn 1004 innhentes tidspunktet for siste innlasting for alle programmer registrert på en databasert anordning og/eller et system, og for filer tilknyttet programmene registrert i den databaserte anordningen og/eller systemet. For eksempel kan spørremodulen 210 spørre logglageret 218 og/eller arkivsamlingen 106 om tidspunktet for siste innlasting for programmer registrert på de databaserte anordningene 102 og/eller tidspunktet for siste lasting av filer, så som dynamiske linkbibliotek-(DLL)-filer, installert med programmene registrert på de databaserte anordningene 102 i systemet 100.

[0095] I trinn 1006 blir tidspunktet for siste modifikasjon av filer eller innstillinger tilknyttet programmer registrert på en databasert anordning og/eller et system samlet inn og sammenliknet med tidspunktet for siste innlasting av programmene. For eksempel kan spørremodulen 210 spørre logglageret 218 og/eller arkivsamlingen 106 om tidspunkt eller dato for siste modifikasjon av et program registrert i operativsystemer på de databaserte anordningene 102. Spørremodulen 210 kan sammenlikne tidspunktet eller datoen for siste modifikasjon med tidspunktet for siste innlasting av programmet.

[0096] I trinn 1008 varsles eventuelle inkonsistenser funnet under sammenlikningen. Dersom for eksempel tidspunktet for siste innlasting av et program registrert i en databasert anordning og/eller et system er senere enn tidspunktet eller datoen for en siste kjent modifikasjon av programmet, er det mulig at programmet ikke har reagert på den siste modifikasjonen. I slike tilfeller kan en feilrapport bli sendt til entiteter, så som en bruker eller en systemadministrator, som rapporterer at programmet ikke reagerer på den siste forsøkte modifikasjonen. Alternativt kan det bli gjort et forsøk på å utføre den siste forsøkte modifikasjonen av

programmet på nytt.

[0097] I ett utførelseseksempel kan spørremodulen 210 spørre etter både tidspunktet for siste innlasting og tidspunktet for siste modifikasjon av et program registrert hos operativsystemene på de databaserte anordningene 102 i systemet 100. Spørremodulen 210 kan sammenlikne tidspunktet for siste innlasting med tidspunktet for siste modifikasjon, og dersom tidspunktet for siste innlasting av programmet er senere enn tidspunktet for siste modifikasjon av programmet, kan spørremodulen 210 sende en feilrapport til entiteter, så som en bruker eller en systemadministrator, som rapporterer at programmet ikke reagerer på den siste forsøkte modifikasjonen. I nok en annen utførelse kan spørremodulen 210 også forsøke å utføre den siste forsøkte modifikasjonen av programmet på nytt.

Eksempel på datamaskinmiljø

[0100] Figur 11 viser et eksempel på generelt datamaskinmiljø 1100 som kan anvendes for å realisere teknologien beskrevet her, og som kan være representativt, helt eller delvis, for elementer beskrevet her. Datamaskinmiljøet 1100 er bare ett eksempel på databehandlingsmiljø, og er ikke ment å antyde noen som helst begrensning når det gjelder bruksområdet eller funksjonaliteten til datamaskin- og nettverksarkitekturer. Heller ikke skal datamaskinmiljøet 1100 tolkes som å ha noen som helst avhengighet eller krav vedrørende noen som helst enkeltkomponent eller kombinasjon av komponenter illustrert i eksempelet på datamaskinmiljø 1100.

[0101] Datamaskinmiljøet 1100 omfatter en generell databasert anordning i form av en datamaskin 1102. Datamaskinen 1102 kan for eksempel være en stasjonær personlig datamaskin, en håndholdt datamaskin, en bærbar datamaskin, en tjenermaskin, en spillkonsoll eller annet. Komponentene i datamaskinen 1102 kan omfatte, men er ikke begrenset til én eller flere prosessorer eller prosesseringsenheter 1104, et systemminne 1106 og en systembuss 1108 som kobler forskjellige systemkomponenter omfattende prosessoren 1104 til systemminnet 1106.

[0102] Systembussen 1108 representerer én eller flere av hvilke som helst av forskjellige typer busstrukturer, omfattende en minnebuss eller minnestyringsenhet, en ekstern buss, en akselerert grafikkport og en prosessor eller lokal buss som anvender en hvilken som helst av en rekke forskjellige

bussarkitekturer. Som et eksempel kan slike arkitekturer omfatte en ISA-(Industry Standard Architecture)-buss, en MCA-(Micro Channel Architecture)-buss, en Enhanced ISA-(EISA)-buss, en lokal VESA-(Video Electronics Standards Association)-buss og en PCI-(Peripheral Component Interconnects)-buss, også kjent som en Mezzanine-buss.

[0103] Datamaskinen 1102 omfatter typisk en rekke forskjellige datamaskinlesbare medier. Slike medier kan være hvilke som helst tilgjengelige medier som er tilgjengelig for datamaskinen 1102, og omfatter både volatile og ikke-volatile medier samt flyttbare og stasjonære medier.

[0104] Systemminnet 1106 omfatter datamaskinlesbare medier i form av volatilt minne, så som direkteaksessminne (RAM) 1110, og/eller ikke-volatilt minne, så som leseminne (ROM) 1112. Et grunnleggende inn/ut-system (BIOS) 1114, som inneholder de grunnleggende rutiner som bidrar til å overføre informasjon mellom elementer innenfor datamaskinen 1102, for eksempel under oppstart, er lagret i ROM 1112. RAM 1110 inneholder typisk data og/eller programmoduler som er umiddelbart tilgjengelig for og/eller jobbes med av prosesseringsenheten 1104.

[0105] Datamaskinen 1102 kan også omfatte andre flyttbare/stasjonære, volatile/ikke-volatile datalagringsmedier. Som et eksempel illustrerer figur 11 et harddiskdrev 1116 for å lese fra og skrive til et stasjonært, ikke-volatilt magnetisk medium (ikke vist), et magnetplatedrev 1118 for å lese fra og skrive til et flyttbart, ikke-volatilt magnetplatelager 1120 (f.eks. en "diskett") og et optisk platedrev 1122 for å lese fra og/eller skrive til et flyttbart, ikke-volatilt optisk platelager 1124, så som et CD-ROM, DVD-ROM eller andre optiske medier. Harddiskdrevet 1116, magnetplatedrevet 1118 og det optiske platedrevet 1122 er alle koblet til systembussen 1108 av ett eller flere datamediagrensesnitt 1126. Alternativt kan harddiskdrevet 1116, magnetplatedrevet 1118 og det optiske platedrevet 1122 være koblet til systembussen 1108 av ett eller flere andre grensesnitt (ikke vist).

[0106] Platedrevene og deres tilhørende datamaskinlesbare medier besørger ikke-volatil lagring av datamaskinlesbare instruksjoner, datastrukturer, programmoduler og andre data for datamaskinen 1102. Selv om eksempelet illustrerer en harddisk 1116, et flyttbart magnetplatelager 1120 og et flyttbart optisk platelager 1124, må det forstås at andre typer datamaskinlesbare medier som kan

lagre data tilgjengelig for en datamaskin, så som magnetkasetter eller andre magnetiske lagringsanordninger, flashminnekort, CD-ROM, DVD eller andre optiske lagre, direkteaksessminne (RAM), leseminne (ROM), elektrisk slettbare programmerbare leseminner (EEPROM) og liknende, også kan anvendes for å realisere det eksemplifiserte datasystemet og miljøet.

[0107] Et hvilket som helst antall programmoduler kan være lagret på harddisken 1116, magnetplatelageret 1120, det optiske platelageret 1124, ROM 1112 og/eller RAM 1110, omfattende, som et eksempel, et operativsystem 1127, ett eller flere applikasjonsprogrammer 1128, andre programmoduler 1130 og programdata 1132. Hvert av nevnte operativsystem 1127, ett eller flere applikasjonsprogrammer 1128, andre programmoduler 1130 og programdata 1132 (eller en kombinasjon av disse) kan implementere alle eller deler av komponentene som støtter det distribuerte filsystemet.

[0108] En bruker kan mate inn kommandoer og informasjon til datamaskinen 1102 gjennom innmatingsanordninger, så som et tastatur 1134 og en pekeranordning 1136 (f.eks. en "mus"). Andre innmatingsanordninger 1138 (ikke vist spesifikt) kan omfatte en mikrofon, styrespak, spillkontroll, parabolantenne, serieport, skanner og/eller liknende. Disse og andre innmatingsanordninger er koblet til prosesseringsenheten 1504 via inn/ut-grensesnittet 1140 som er koblet til systembussen 1108, men kan også være tilkoblet via andre grensesnitt og busstrukturer, så som en parallellport, en spillport eller en USB-port.

[0109] En dataskjerm 1142 eller annen type fremvisningsanordning kan også være koblet til systembussen 1108 via et grensesnitt, så som et videoadapter 1144. I tillegg til dataskjermen 1142 kan andre periferiske utmatingsanordninger omfatte komponenter så som høyttalere (ikke vist) og en skriver 1146 som kan være koblet til datamaskinen 1102 via inn/ut-grensesnittet 1140.

[0110] Datamaskinen 1102 kan kjøre i et nettverksmiljø som anvender logiske forbindelser til én eller flere fjerne datamaskiner, så som en fjern dataanordning 1148. Som et eksempel kan den fjerne dataanordningen 1148 være en personlig datamaskin, en bærbar datamaskin, en tjener, en ruter, et nettverkstilknyttet datamaskin, en peer-anordning eller en annen vanlig nettverksnode, og liknende. Den fjerne dataanordningen 1148 er illustrert som en bærbar datamaskin som kan omfatte mange av eller alle elementene og trekkene beskrevet her i forbindelse

med datamaskinen 1102.

[0111] Logiske forbindelser mellom datamaskinen 1102 og den fjerne datamaskinen 1148 er vist som et lokalt nettverk (LAN) 1150 og et generelt regionalt nettverk (WAN) 1152. Slike nettverksmiljøer er vanlige i kontorer, bedriftsdatanettverk, intranett og på Internett.

[0112] Når den anvendes i et LAN-miljø, er datamaskinen 1102 koblet til et lokalt nettverk 1150 via et nettverksgrensesnitt eller -adapter 1154. Når den anvendes i et WAN-miljø, omfatter datamaskinen 1102 typisk et modem 1156 eller en annen anordning for å etablere kommunikasjon over det regionale nettverket 1152. Modemet 1156, som kan være internt i eller eksternt for datamaskinen 1102, kan være koblet til systembussen 1108 via inn/ut-grensesnittet 1140 eller andre passende mekanismer. Det må forstås at de illustrerte nettverksforbindelsene kun er eksempler, og at andre anordninger for å opprette én eller flere kommunikasjonsforbindelser mellom datamaskinene 1102 og 1148 kan anvendes.

[0113] I et nettverksmiljø, så som det illustrert med databehandlingsmiljøet 1100, kan programmoduler vist i datamaskinen 1102, eller deler av disse, være lagret i en fjern minnelagringsanordning. Som et eksempel er fjerne applikasjonsprogrammer 1158 lagret på en minneanordning i den fjerne datamaskinen 1148. For illustrasjonsformål er applikasjonsprogrammer og andre eksekverbare programkomponenter så som operativsystemet illustrert her som diskrete blokker, selv om en innser at slike programmer og komponenter til forskjellige tider befinner seg i forskjellige lagringskomponenter i den databaserte anordningen 1102, og kjøres av dataprosessoren(e) i datamaskinen.

[0114] Forskjellige moduler og fremgangsmåter kan være beskrevet her i den generelle sammenhengen datamaskin-eksekverbare instruksjoner, så som programmoduler, som blir eksekvert av én eller flere datamaskiner eller andre anordninger. Generelt omfatter programmoduler rutiner, programmer, objekter, komponenter, datastrukturer, etc. som utfører bestemte oppgaver eller implementerer bestemte abstrakte datatyper. Funksjonaliteten til programmodulene kan typisk kombineres eller distribueres som ønsket i forskjellige utførelsesformer.

[0115] En realisering av disse modulene og fremgangsmåtene kan bli lagret

på eller sendt over en eller annen form for datamaskinlesbare medier. Datamaskinlesbare medier kan være hvilke som helst tilgjengelige medier som kan aksesseres av en datamaskin. Som et eksempel, og ikke en begrensning, kan datamaskinlesbare medier omfatte "datalagringsmedier" og "kommunikasjonsmedier."

[0116] "Datalagringsmedier" omfatter volatile og ikke-volatile, flyttbare og stasjonære medier realisert med en hvilken som helst metode eller teknologi for lagring av informasjon, så som datamaskinlesbare instruksjoner, datastrukturer, programmoduler eller andre data. Datalagringsmedier omfatter, men er ikke begrenset til RAM, ROM, EEPROM, flashminne eller annen minneteknologi, CD-ROM, DVD eller andre optiske lagre, magnetkasetter, magnetbånd, magnetplatelagre eller andre magnetiske lagringsanordninger, eller et hvilket som helst annet medium som kan anvendes for å lagre den ønskede informasjonen og som kan aksesseres av en datamaskin.

[0117] Alternativt kan deler av rammeverket realiseres i maskinvare eller en kombinasjon av maskinvare, programvare og/eller fastvare. For eksempel vil én eller flere applikasjonsspesifikke integrerte kretser (ASIC-er) eller programmerbare logikkanordninger (PLD-er) kunne innrettes eller programmeres til å implementere én eller flere deler av rammeverket.

KONKLUSJON

[0118] Selv om utførelsesformer av forvaltning og analyse av systemer har blitt beskrevet med en ordlyd som er spesifikk for oppbygningsmessige trekk og/eller fremgangsmåter, må det forstås at gjenstanden for de vedføyde kravene ikke nødvendigvis er begrenset til de konkrete trekk eller fremgangsmåter som er beskrevet. Tvert imot er de konkrete trekk og fremgangsmåter beskrevet som eksempler på utførelser av forvaltning og analyse av systemer.

PATENTKRAV

1. Fremgangsmåte, omfattende det å:

listeføre installasjonsfiler og innstillingsendringer knyttet til programmer på en databasert anordning;

listeføre filer og registerinnstillinger knyttet til avinstallerte filer på den databaserte anordningen; og

sammenlikne filene og registerinnstillingne for å utpeke lekkede filer.

2. Fremgangsmåte ifølge krav 1, der det å listeføre installasjonsfiler og innstillingsendringer knyttet til programmer på en databasert anordning omfatter det å opprette en liste av programmer registrert hos et operativsystem på den databaserte anordningen.

3. Fremgangsmåte ifølge krav 1, der det å listeføre installasjonsfiler og innstillingsendringer knyttet til programmer på en databasert anordning omfatter det å søke for å detektere de aktuelle programmene.

4. Fremgangsmåte ifølge krav 1, der det å listeføre filer og registerinnstillinger knyttet til avinstallerte filer på den databaserte anordningen omfatter det å søke gjennom minne på den databaserte anordningen.

5. Fremgangsmåte ifølge krav 1, der det å sammenlikne omfatter det å korrelere identifikatorer for filer og registerinnstillinger knyttet til programmer registrert hos et operativsystem på den databaserte anordningen med identifikatorer for filer og registerinnstillinger knyttet til programmer installert på den databaserte anordningen.

6. Fremgangsmåte ifølge krav 1, videre omfattende det å fjerne de lekkede filene.

7. Fremgangsmåte, omfattende det å:

katalogisere programmer lastet inn på en databasert anordning;

innhente tidspunktet for siste innlasting av programmer som er registrert hos den databaserte anordningen; og

sammenlikne tidspunktet for siste modifikasjon av filer tilknyttet programmene som er registrert hos den databaserte anordningen med tidspunktet for siste innlasting.

8. Fremgangsmåte ifølge krav 7, der det å katalogisere omfatter det å opprette en liste av programmer registrert hos et operativsystem på den databaserte anordningen.

9. Fremgangsmåte ifølge krav 7, der det å katalogisere omfatter det å listeføre programmene ved å plassere programmene på en liste.

10. Fremgangsmåte ifølge krav 7, der det å katalogisere omfatter det å søke for å detektere alle programmer registrert hos den databaserte anordningen.

11. Fremgangsmåte ifølge krav 7, der det å innhente omfatter det å spørre en loggfil.

12. Fremgangsmåte ifølge krav 7, der det å sammenlikne omfatter det å spørre en loggfil.

13. Fremgangsmåte ifølge krav 7, videre omfattende det å varsle eventuelle inkonsistenser funnet under sammenlikningen.

14. Fremgangsmåte ifølge krav 7, videre omfattende det å varsle eventuelle inkonsistenser funnet under sammenlikningen.

katalogisere programmer på en databehandlingsanordning;

registrere tidspunktene for siste innlasting av programmene;

innhente tidspunktene for siste modifisering av filer tilknyttet programmene; og

sammenlikne tidspunktene for siste modifisering av filene med tidspunktene for siste innlating av programmene.

15. Fremgangsmåte, omfattende det å:

motta informasjon vedrørende et program som kjører på et system;

sammenlikne attributter for programmet med attributter i en liste over godkjente og ikke godkjente programmer; og

fastslå om attributtene svarer til ikke godkjente eller godkjente programmer.

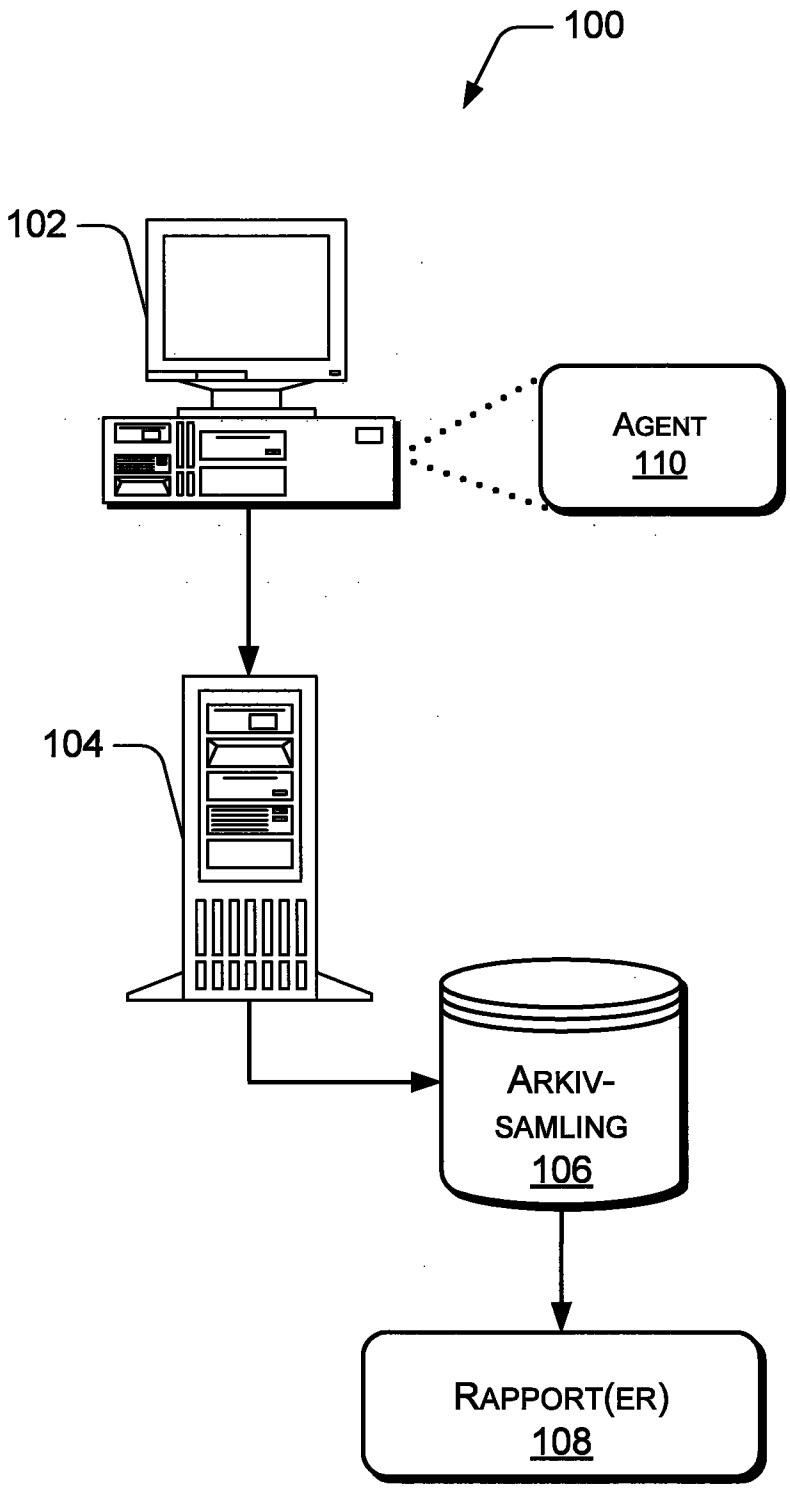


Fig. 1

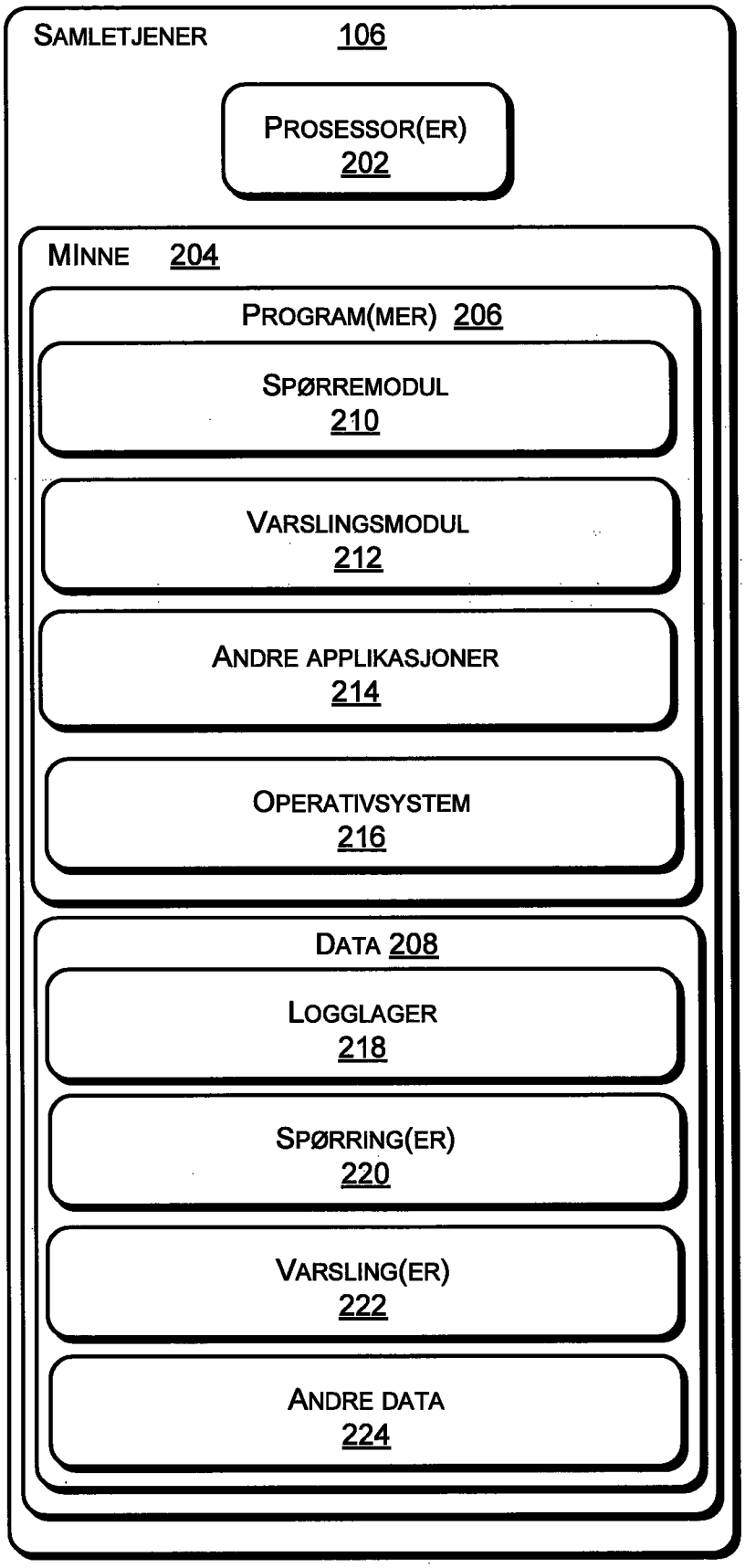

SAMLETJENER 106
PROSESSOR(ER) 202
MINNE 204
PROGRAM(MER) 206
SPØRREMODUL 210
VARSLINGSMODUL 212
ANDRE APPLIKASJONER 214
OPERATIVSYSTEM 216
DATA 208
LOGGLAGER 218
SPØRRING(ER) 220
VARSLING(ER) 222
ANDRE DATA 224

Fig. 2

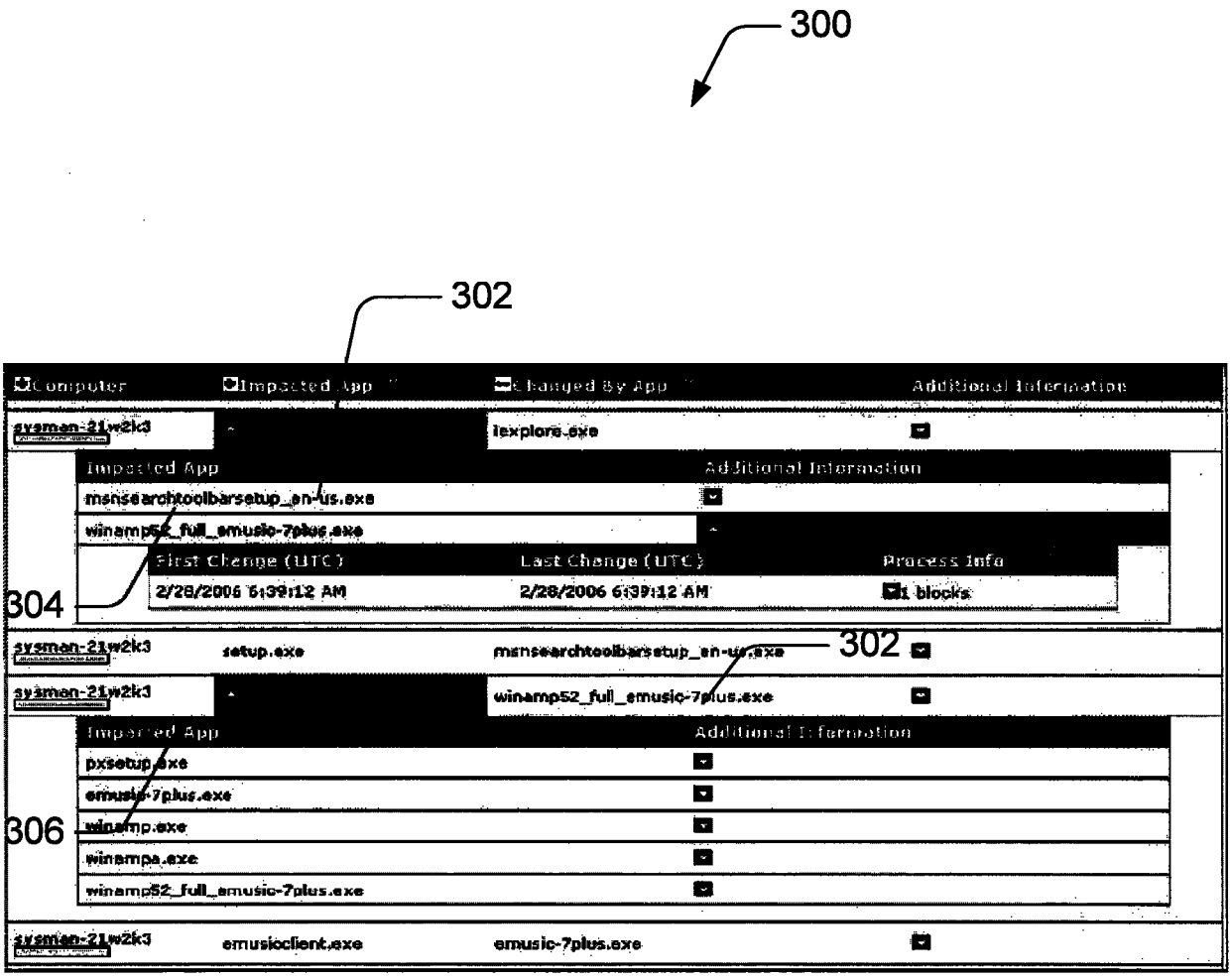

300
302
Computer
Impacted App
Changed By App
Additional Information
sysmon-21w2k3
iexplore.exe
Impacted App
Additional Information
msnsearchtoolbarsetup_en-us.exe
winamp52_full_emusic-7plus.exe
First Change (UTC)
Last Change (UTC)
Process Info
2/28/2006 6:39:12 AM
2/28/2006 6:39:12 AM
1 blocks
304
sysmon-21w2k3
setup.exe
msnsearchtoolbarsetup_en-us.exe
302
sysmon-21w2k3
winamp52_full_emusic-7plus.exe
Impacted App
Additional Information
pxsetup.exe
emusic-7plus.exe
winamp.exe
winampa.exe
winamp52_full_emusic-7plus.exe
306
sysmon-21w2k3
emusicclient.exe
emusic-7plus.exe

Fig. 3

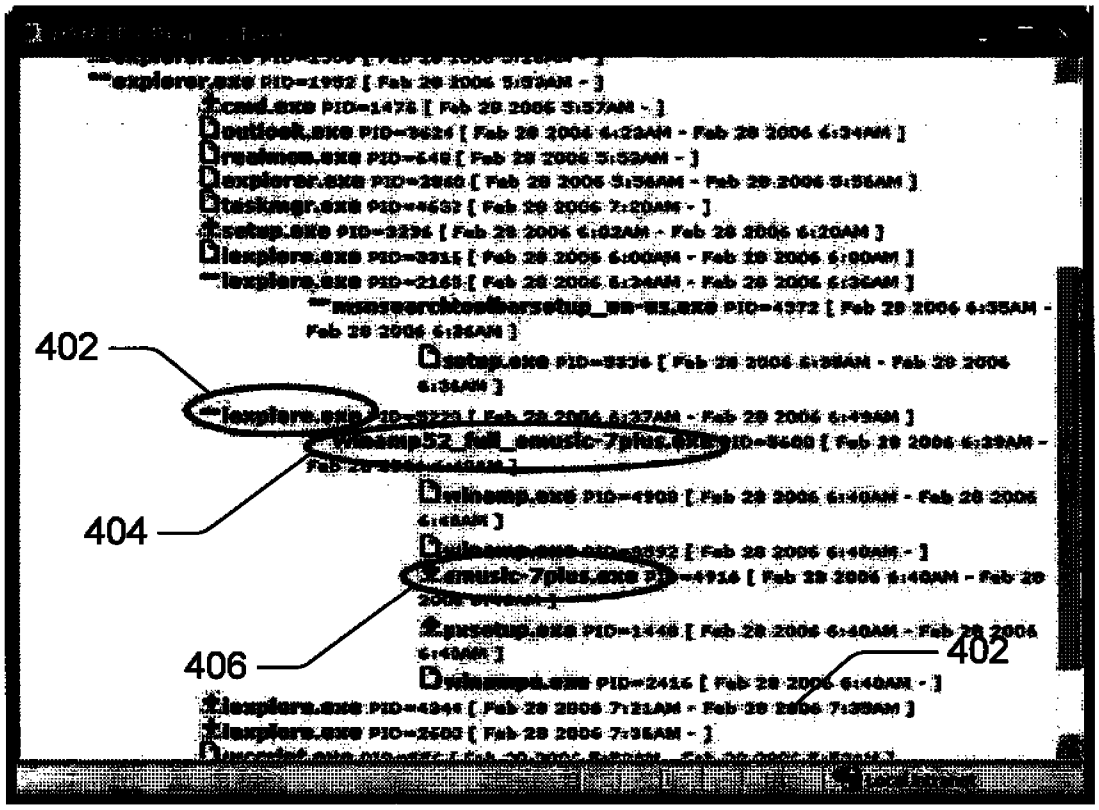


Fig. 4

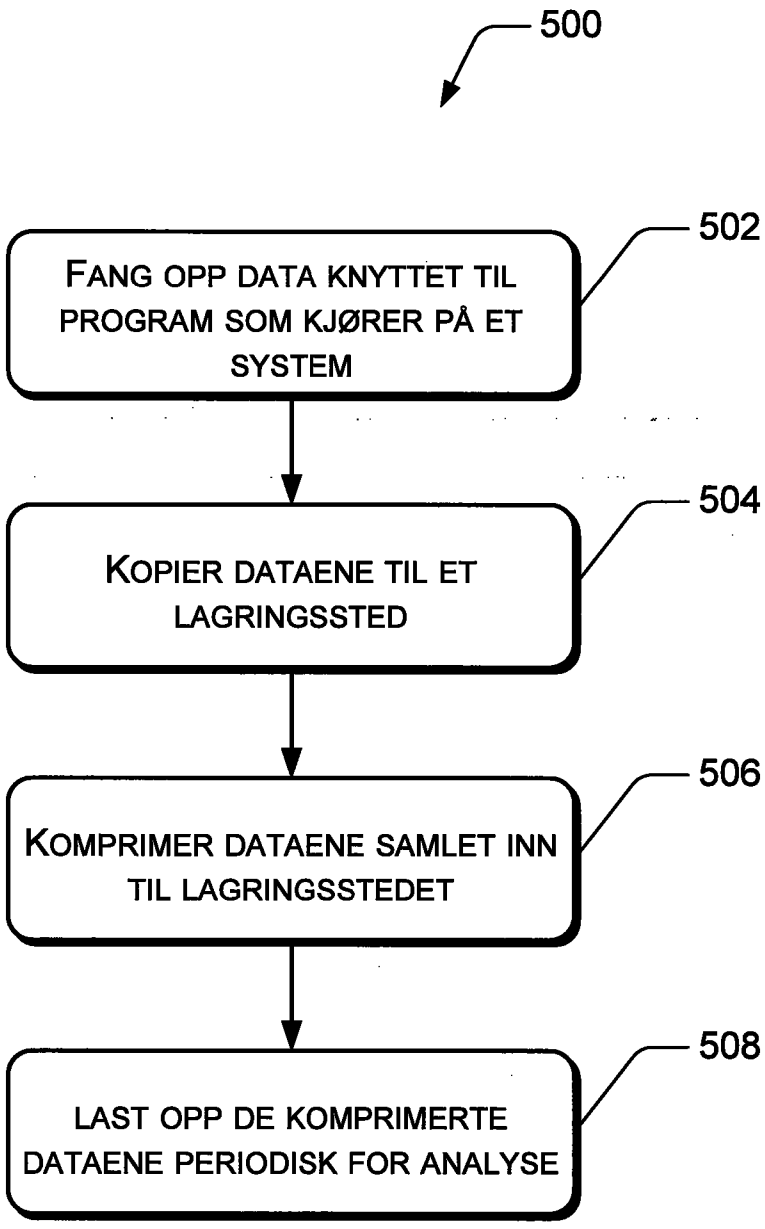

500
FANG OPP DATA KNYTTET TIL PROGRAM SOM KJØRER PÅ ET SYSTEM
502
KOPIER DATAENE TIL ET LAGRINGSSTED
504
KOMPRIMER DATAENE SAMLET INN TIL LAGRINGSSTEDET
506
LAST OPP DE KOMPRIMERTE DATAENE PERIODISK FOR ANALYSE
508

Fig. 5

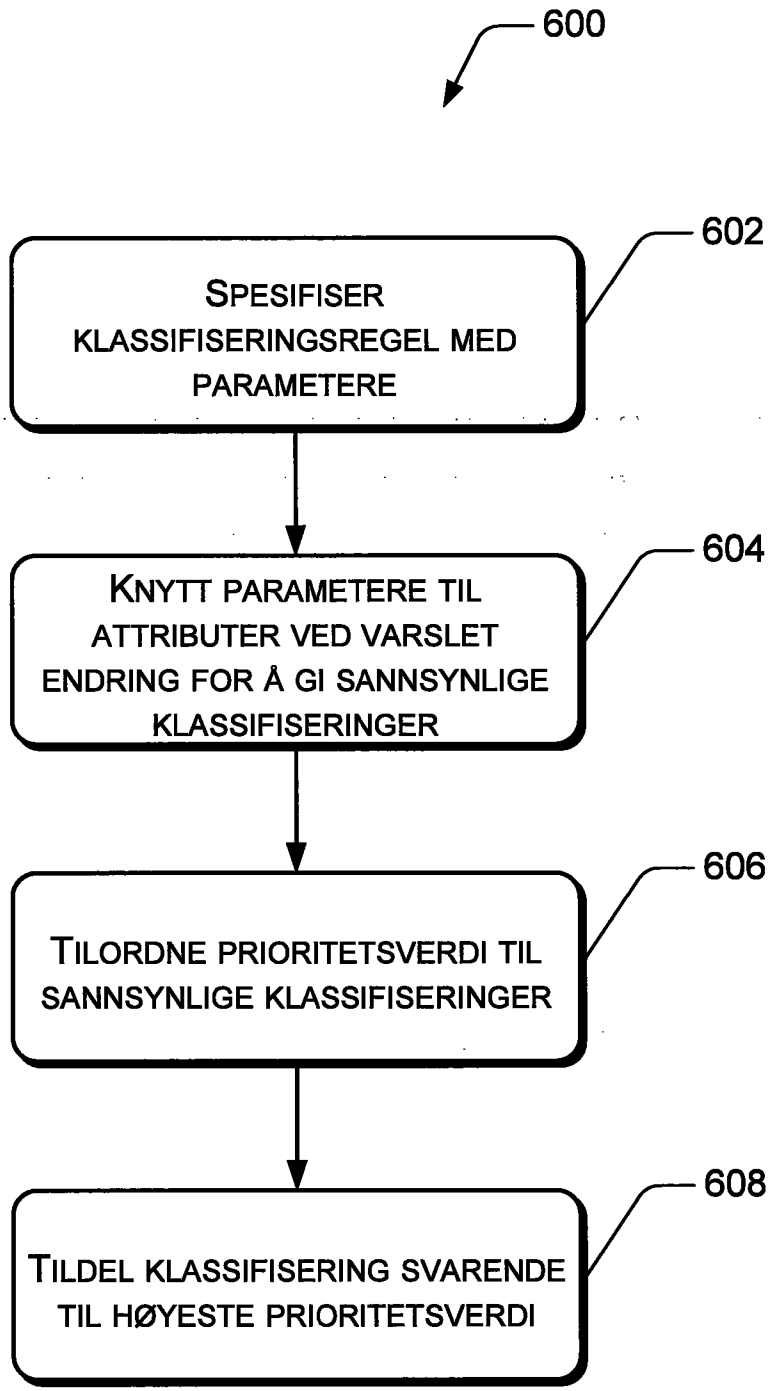

600
602
SPESIFISER KLASSIFISERINGSREGEL MED PARAMETERE
604
KNYTT PARAMETERE TIL ATTRIBUTER VED VARSLET ENDRING FOR Å GI SANNSYNLIGE KLASSIFISERINGER
606
TILORDNE PRIORITETSVERDI TIL SANNSYNLIGE KLASSIFISERINGER
608
TILDEL KLASSIFISERING SVARENDE TIL HØYESTE PRIORITETSVERDI

Fig. 6

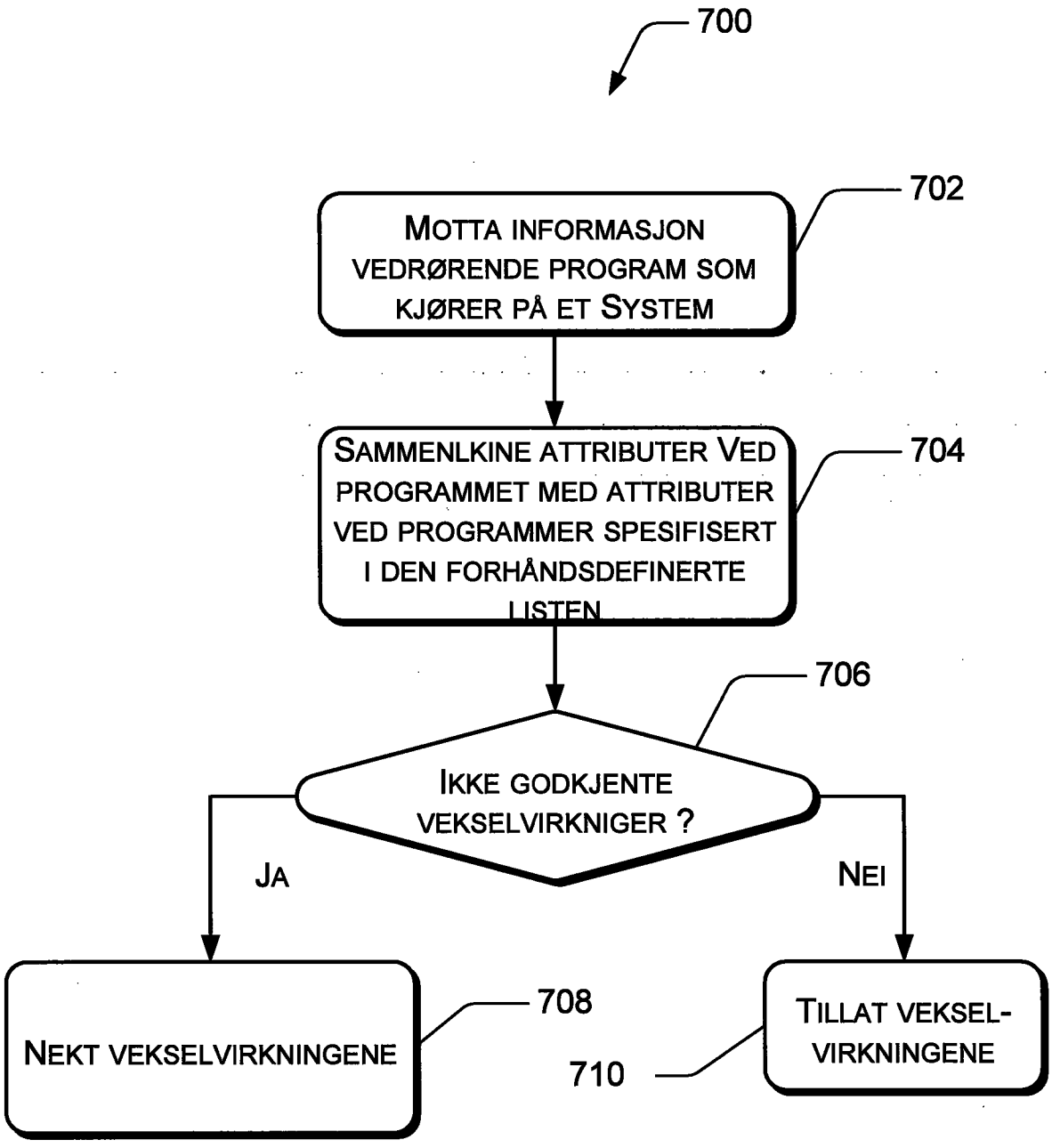

700
MOTTA INFORMASJON VEDRØRENDE PROGRAM SOM KJØRER PÅ ET SYSTEM
702
SAMMENLKINE ATTRIBUTER VED PROGRAMMET MED ATTRIBUTER VED PROGRAMMER SPESIFISERT I DEN FORHÅNDSDEFINERTE LISTEN
704
706
IKKE GODKJENTE VEKSELVIRKNIGER ?
JA
NEI
NEKT VEKSELVIRKNINGENE
708
TILLAT VEKSEL-VIRKNINGENE
710

Fig. 7

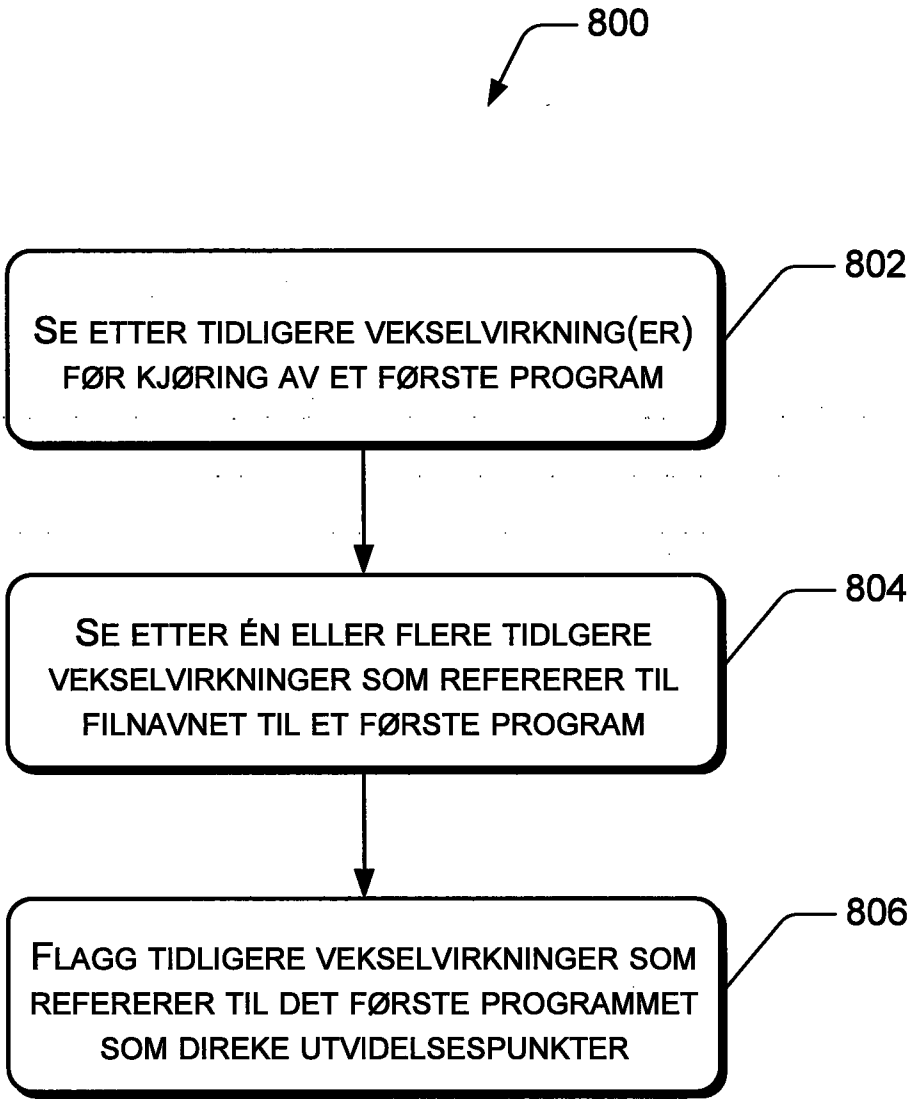

800
SE ETTER TIDLIGERE VEKSELVIRKNING(ER) FØR KJØRING AV ET FØRSTE PROGRAM
802
SE ETTER ÉN ELLER FLERE TIDLGERE VEKSELVIRKNINGER SOM REFERERER TIL FILNAVNET TIL ET FØRSTE PROGRAM
804
FLAGG TIDLIGERE VEKSELVIRKNINGER SOM REFERERER TIL DET FØRSTE PROGRAMMET SOM DIREKE UTVIDELSESPUNKTER
806

Fig. 8

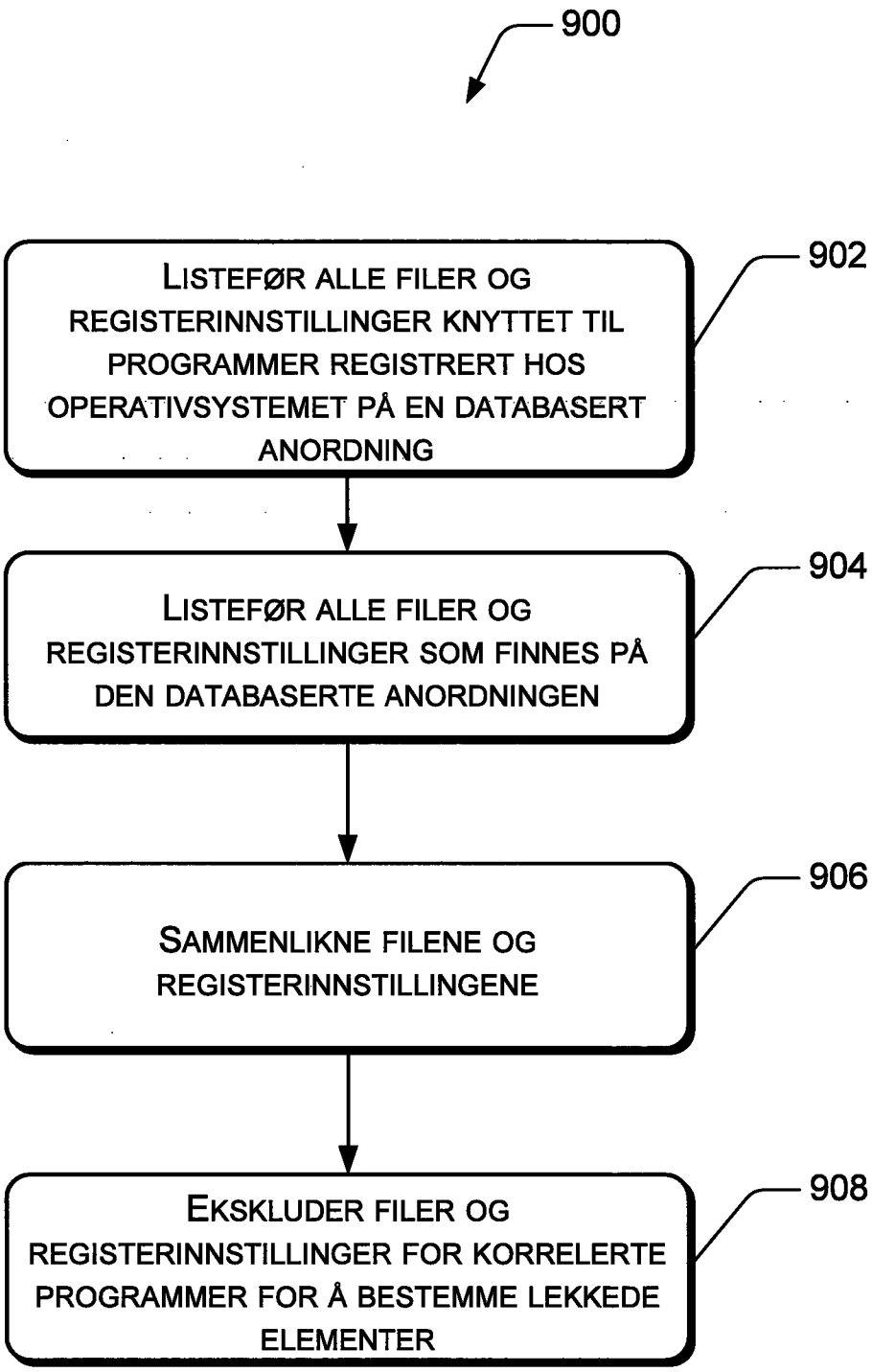

900
LISTEFØR ALLE FILER OG REGISTERINNSTILLINGER KNYTTET TIL PROGRAMMER REGISTRERT HOS OPERATIVSYSTEMET PÅ EN DATABASERT ANORDNING
902
LISTEFØR ALLE FILER OG REGISTERINNSTILLINGER SOM FINNES PÅ DEN DATABASERTE ANORDNINGEN
904
SAMMENLIKNE FILENE OG REGISTERINNSTILLINGENE
906
EKSKLUDER FILER OG REGISTERINNSTILLINGER FOR KORRELERTE PROGRAMMER FOR Å BESTEMME LEKKEDE ELEMENTER
908

Fig. 9

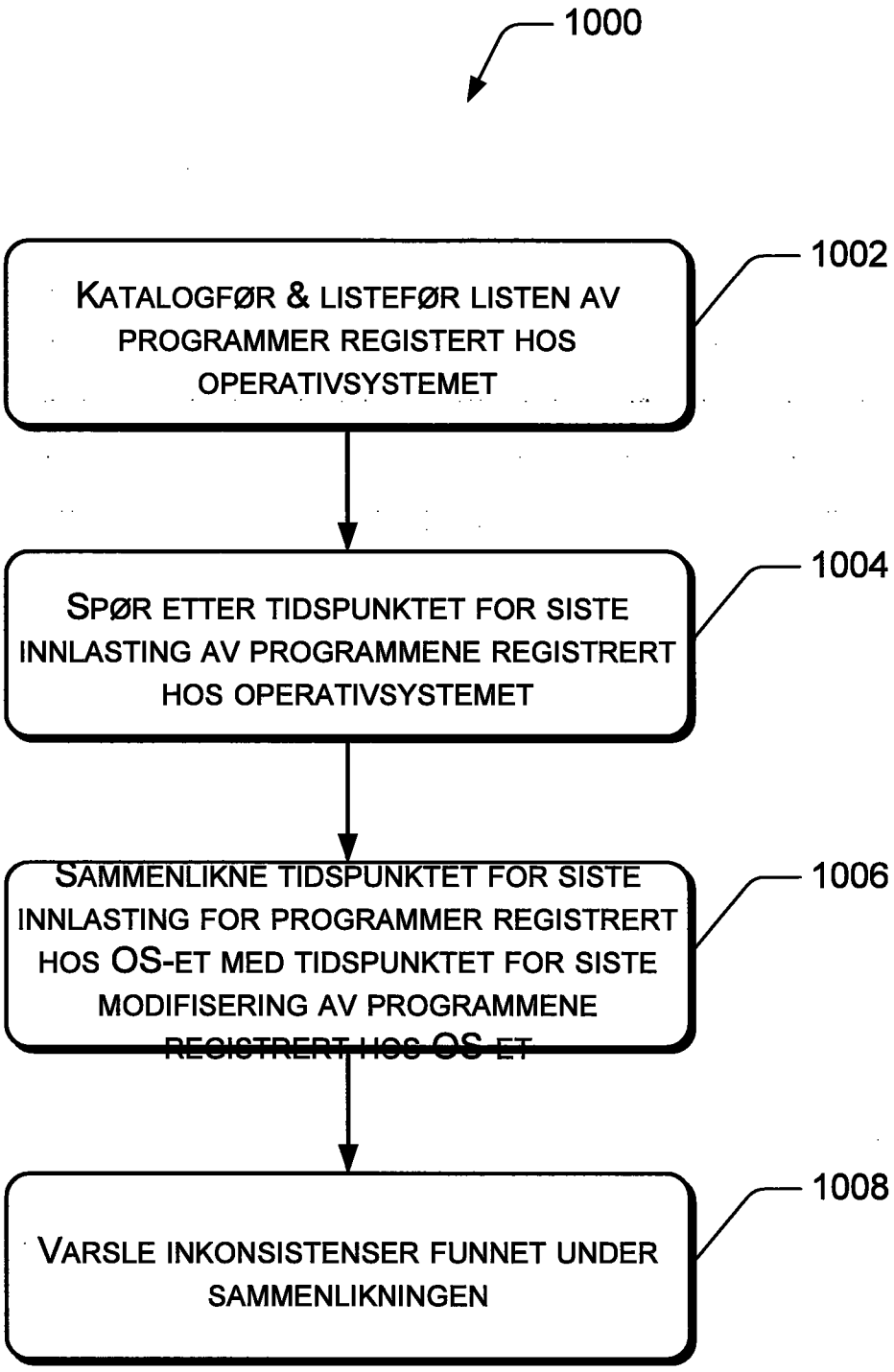

1000
KATALOGFØR & LISTEFØR LISTEN AV PROGRAMMER REGISTERT HOS OPERATIVSYSTEMET
1002
SPØR ETTER TIDSPUNKTET FOR SISTE INNLASTING AV PROGRAMMENE REGISTRERT HOS OPERATIVSYSTEMET
1004
SAMMENLIKNE TIDSPUNKTET FOR SISTE INNLASTING FOR PROGRAMMER REGISTRERT HOS OS-ET MED TIDSPUNKTET FOR SISTE MODIFISERING AV PROGRAMMENE REGISTRERT HOS OS-ET
1006
VARSLE INKONSISTENSER FUNNET UNDER SAMMENLIKNINGEN
1008

Fig. 10

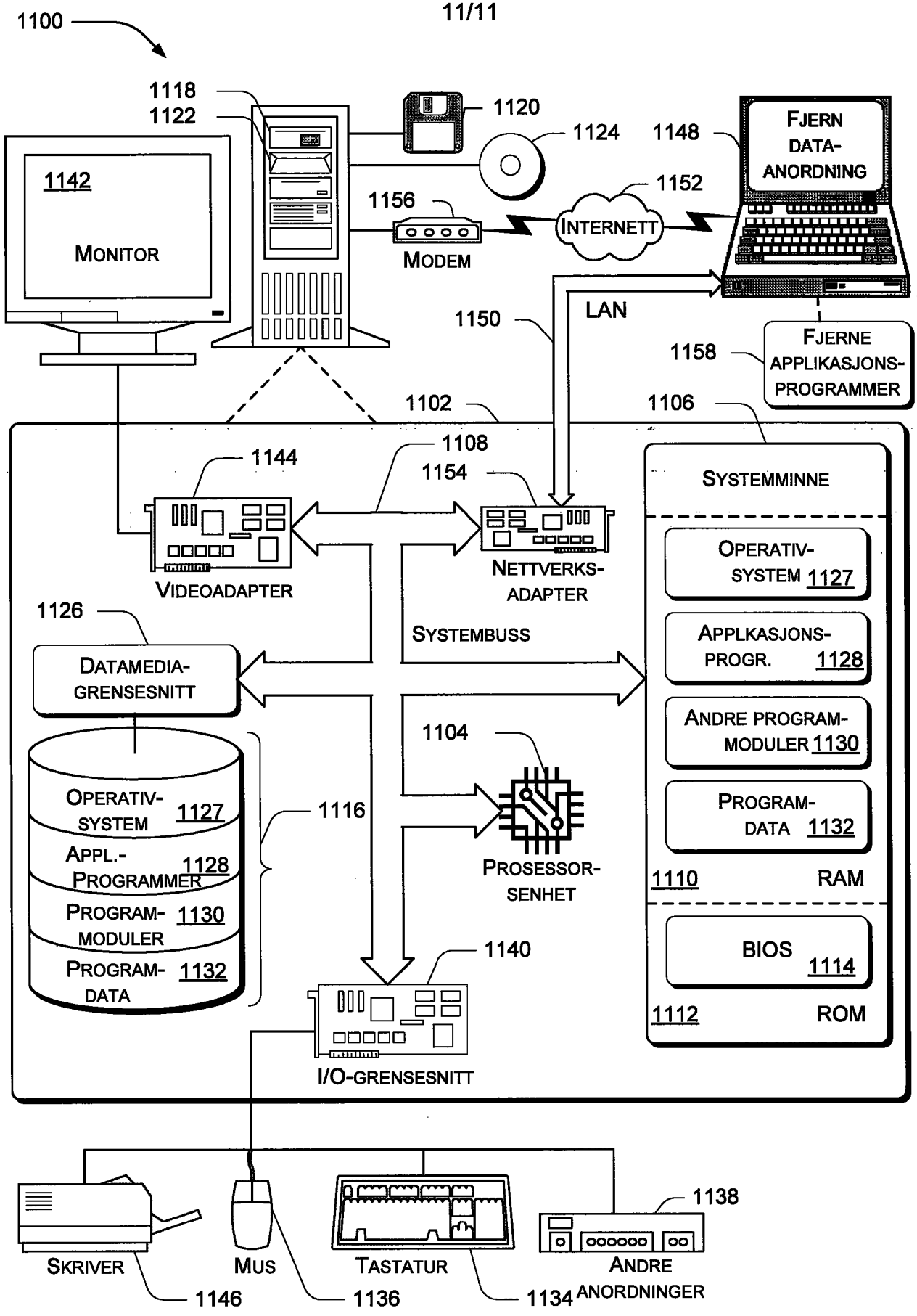

1100
1118
1122
1120
1124
1148
FJERN DATA-ANORDNING
1142
MONITOR
1156
MODEM
1152
INTERNETT
LAN
1150
1158
FJERNE APPLIKASJONS-PROGRAMMER
1102
1106
1108
1144
1154
SYSTEMMINNE
VIDEOADAPTER
NETTVERKS-ADAPTER
OPERATIV-SYSTEM 1127
1126
SYSTEMBUSS
APPLKASJONS-PROGR. 1128
DATAMEDIA-GRENSESNITT
ANDRE PROGRAM-MODULER 1130
1104
OPERATIV-SYSTEM 1127
1116
PROGRAM-DATA 1132
APPL.-PROGRAMMER 1128
PROSESSOR-SENHET
1110
RAM
PROGRAM-MODULER 1130
1140
BIOS 1114
PROGRAM-DATA 1132
1112
ROM
I/O-GRENSESNITT
1138
SKRIVER
MUS
TASTATUR
ANDRE ANORDNINGER
1146
1136
1134

Fig. 11