

KOZZÉTÉTELI
PÉLDÁNY

P0600518

73.775/KOT

KIVONAT

Információ-feldolgozási eljárás, információ-terminál támogató szerver, együttműködési rendszer, és (egy) információ-feldolgozási programot tároló adathordozó

Eljárás, szoftver és rendszer egy együttműködő kliens viselkedésének kontextusnak megfelelő megváltoztatásához egy Interneten való együttműködés során. Egy felhasználói oldal böngészője egy HTTP kérelmet küld a 140 Web szervernek a 110 információ terminál támogató szerveren keresztül. Ezután, a kérésre válaszolva, a 140 Web szerver kontextust küld a 131 felhasználói oldal Web böngészőjének a 110 információ terminál támogató szerveren keresztül. A 110 információ terminál támogató szerver ellenőrzi a kontextus tartalmát, és amikor megállapítja, hogy egy meghatározott feltétellel egyezik, akkor egy parancsot illeszt a kontextusba. A parancs beillesztés nélküli kontextust elküldjük a 131 felhasználói oldal Web böngészőnek, és a paranccsal beillesztett kontextust elküldjük a 131 ügyintéző oldal Web böngészőjének. A 130 ügyintéző oldal információ terminál 135 kliens programja elolvassa a beágyazott parancsot az API segítségével, amellyel a Web böngésző rendelkezik, és elvégzi a feldolgozást, azért, hogy a parancs tartalmának megfelelően megtiltsa az űrlap megtekintését, az űrlap megváltoztatását és specifikus űrlapok vagy mezők elrejtését.

(2. dbr)

73.775/KOT

KÖZZÉTÉTELI PÉLDÁNY

Információ-feldolgozási eljárás, információ-terminált támogató
szerver, együttműködési rendszer, és információ-feldolgozási
programot tároló adathordozó

A találmány tárgya információ feldolgozási eljárás, közelebbről, eljárás egy szervertől küldött tartalom editálásához egy klienstől érkező kérésre, egy adminisztrátor szándékainak megfelelően.

Találmányunk beadásakor ismert az olyan együttműködési technológia, például a Hei. 10-124461 japán szabadalomtól, ahol egy HTML oldalon több felhasználó végezhet egyidejűleg kooperatív munkát (olvasás, mozgatás és változtatás). Az 5. ábrán egy ilyen HTML oldalon történő kooperatív munkára mutatunk példát egy Internetes bankrendszerben való alkalmazás esetében. Egy bankfiókkezelő, egy Internetes vásárló konzultációs központ ügyintézője, stb. felkeresheti ugyanazt az oldalt, mint az ügyfél, és műveleteket hajthat végre ezzel a technológiával.

Speciális HTML dokumentumok, mint például egy termék rendelésére szolgáló képernyő megjelenítés, vagy pénzáttétel esetén a rendeltetési hely megjelölésére szolgáló képernyő megjelenítéskor, vannak olyan esetek, ahol nem kívánatos, hogy egy ügyintéző betekintést nyerjen a HTML űrlapba, vagy a HTML űrlapba bevitt adatokat megváltoztassa. Bár az együttműködési szerve-

reket eredetileg azért fejlesztették ki, hogy el lehessen osztani ugyanazt a HTML tartalmat az együttműködő böngészők között. Ezért nincs mód annak megismerésére, hogy az együttműködő kliensek ilyen viselkedése mikor és hogyan változott meg.

Találmányunk egyrészt együttműködési rendszer, amely alkalmas arra, hogy kontextusnak megfelelően megváltoztassa az együttműködő kliens viselkedését Internet alapú együttműködés során.

Találmányunkkal egy olyan rendszert biztosítunk, ahol egy adminisztrátor szándékosan meg tudja változtatni egy kliens által kért tartalmat.

Találmányunkkal egy kevés költséggel megvalósítható együttműködési rendszert biztosítunk, ahol egy információ terminál támogatásához szükséges erőforrásokat csökkentjük.

A találmányunk szerinti együttműködési rendszerben az információ terminál támogatás közben centralizált vezérlés valósítható meg.

A találmányunk szerinti együttműködési rendszer nem függ a működtetett információ terminál platformjától.

Találmányunkkal egy olyan együttműködés vezérlő rendszert biztosítunk, amely alkalmas arra, hogy szerver változás nélkül vezérelje egy információ terminálnak nyújtott szolgáltatás tartalmát az információ terminál kérésére.

Találmányunk egyik megvalósítása szerint egy felhasználói oldalon levő böngészőtől egy HTTP kérést küldünk egy információ terminál támogató szerveren keresztül egy Web szerverhez. Ezután, a Web szerver a kérésre reagálva elküldi a kontextust a

felhasználói oldal Web böngészőjének az információ terminál támogató szerveren keresztül. Ekkor az információ terminál támogató szerver ellenőrzi a kontextus tartalmát, és amikor megállapítja, hogy az egyezik egy előre meghatározott feltétellel, akkor egy parancsot illeszt a kontextusba. A beillesztett parancs nélküli kontextust elküldjük a felhasználói oldal Web böngészőjéhez, és a parancsot tartalmazó kontextust elküldjük az ügyintéző oldal Web böngészőjéhez. Az ügyintéző oldal információ termináljának a kliens programja ezután elolvassa a beágyazott parancsot egy Web böngésző birtokában levő API segítségével, és elvégzi a feldolgozást, azért, hogy megtiltsa az űrlapba való betekintést, az űrlap megváltoztatását és specifikus űrlapok vagy mezők elrejtését, a parancs tartalmának megfelelően.

Találmányunk egyrészt információ feldolgozási eljárás egy információ feldolgozó rendszerben, amely tartalmaz egy, a felhasználói oldal információ termináljába betöltött böngésző együttműködését támogató információ terminál támogató szerveret, és egy, az ügyintéző oldal információ termináljába betöltött böngészőt. Az eljárás során a következő lépéseket tesszük:

(a) az információ terminál támogató szervernél fogadunk egy, a felhasználói oldal böngésző kérésére válaszolva elküldött HTTP üzenetet;

(b) meghatározzuk, hogy a HTTP üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) ha a HTTP üzenet egyezik az előre meghatározott feltétellel, akkor a HTTP üzenet tartalmát editáljuk;

(d) az editálatlan HTTP üzenetet elküldjük a felhasználói oldal böngészőjének; és

(e) az editált HTTP üzenetet elküldjük az ügyintéző oldal böngészőjének.

Ismertetésünkben az „információ terminál támogató szerver” fogalmába beletartoznak az együttműködő szerverek és a proxy (közvetítő) szerverek.

A találmányunk szerinti információ feldolgozási eljárás során a (c) lépésben az editálást egy, az ügyintéző oldal információ termináljába betöltött kliens program számára szóló parancs beillesztéséhez végezzük. Ismertetésünkben a „parancs” fogalmába beletartoznak azok a feldolgozási utasítás parancsok, amelyeket böngésző specifikus API-kkal lehet elvégezni.

A találmányunk szerinti információ feldolgozási eljárásban a parancs az „ürlap betekintés tiltása”, „ürlap változtatás tiltása”, specifikus ürlap elrejtése” vagy „specifikus mező elrejtése” parancsok valamelyike.

Találmányunk továbbá információ feldolgozási eljárás egy információ feldolgozó rendszerben, amely egy információ terminál támogatásához egy tartalom szerverhez kapcsolódó információ terminál támogató szerverrel rendelkezik, és egy kliens programot tartalmaz. Az eljárás során a következő lépéseket tesszük:

(a) az információ terminál támogató szervernél fogadunk egy üzenetet a tartalom szervertől, válaszolva az információ terminál kérésére;

(b) meghatározzuk, hogy az üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) ha az üzenet egyezik az előre meghatározott feltétellel, akkor az üzenetet editáljuk egy, a kliens programnak szóló parancs beillesztéséhez; és

(d) az editált üzenetet elküldjük az információ terminálnak.

Ismertetésünkben a „tartalom szerver” fogalmába tartoznak az olyan szerverek, mint például a Web szerverek, stb., amelyek a kliens kérésének eleget téve tartalmat küldenek a kliens oldalnak.

Találmányunk továbbá információ terminál támogató szerver, amely egy felhasználói oldal információ termináljába töltött böngésző és egy ügyintéző oldal információ termináljába töltött böngésző együttműködését támogatja. Az információ terminál támogató szerver tartalmaz:

(a) egy szabály meghatározás részt, amely tartalmaz egy feltétel felállítás részt és egy parancs felállítás részt;

(b) egy szabályvezérlés irányítót az információ támogató szervernél a felhasználói oldal kéréseire válaszként küldött HTTP üzenetek figyelésére;

(c) egy HTTP ellenőrt, amely meghatározza, hogy a HTTP üzenet egyezik-e vagy sem a szabály meghatározás rész feltétel felállítás részének feltételével;

(d) egy HTTP szerkesztőt, amely a HTTP üzenet tartalmát a szabály meghatározás rész parancs felállítás részének tartalma szerint editálja, amikor a HTTP üzenet egyezik egy előre meghatározott feltétellel;

(e) felhasználói cache-tárt, amely a felhasználói oldal böngészőjének küldött editálatlan HTTP üzenetet tárolja; és

(f) ügyintéző cache-tárt, amely az ügyintéző oldal böngészőjének küldött editált HTTP üzenetet tárolja.

A találmányunk szerinti információ terminál támogató szerverben a HTTP szerkesztő az editálást egy, az ügyintéző oldal információ termináljába betöltött kliens program számára szóló parancs beillesztéséhez végzi.

A találmányunk szerinti információ terminál támogató szerver az „úrlap betekintés tiltása”, „úrlap változtatás tiltása”, specifikus úrlap elrejtése” vagy „specifikus mező elrejtése” parancsok valamelyikét határozza meg.

Találmányunk továbbá egy információ terminál támogató szerver, amely egy tartalom szerverhez kapcsolódó információ terminált támogat, és egy kliens programmal rendelkezik. Az információ terminál támogató szerver tartalmaz:

(a) egy üzenet ellenőrt, amely meghatározza, hogy az információ terminál kérésének eleget téve a tartalom szervertől küldött üzenet egyezik-e vagy sem egy előre meghatározott feltétellel; és

(b) egy üzenet szerkesztőt, amely a kliens programot utasító parancsot beillesztve editálja az üzenetet, amikor az üzenet egyezik az előre meghatározott feltétellel.

Találmányunk továbbá egy együttműködő rendszer, amely egy felhasználói oldal információ termináljába töltött böngésző és egy ügyintéző oldal információ termináljába töltött böngésző együttműködését támogatja. Az együttműködő rendszer tartalmaz:

(a) egy, a felhasználói oldal információ termináljával együttműködő ügyintéző oldal információ terminált;

(b1) egy szabály meghatározás részt, amely tartalmaz egy feltétel felállítás részt és egy parancs felállítás részt;

(b2) egy szabályvezérlés irányítót egy, a felhasználói oldal kéréseire válaszként küldött HTTP üzenet figyelésére;

(b3) egy HTTP ellenőrt, amely meghatározza, hogy a HTTP üzenet egyezik-e vagy sem a szabály meghatározás rész feltétel felállítás részének feltételével;

(b4) egy HTTP szerkesztőt, amely a HTTP üzenet tartalmát a szabály meghatározás rész parancs felállítás részének tartalma szerint editálja, amikor a HTTP üzenet egyezik egy előre meghatározott feltétellel;

(b5) felhasználói cache-tárt, amely a felhasználói oldal böngészőjének küldött editálatlan HTTP üzenetet tárolja; és

(b6) egy információ terminál támogató szerveret, amely tartalmaz egy ügyintéző cache-tárt, amely az ügyintéző oldal böngészőjének küldött editált HTTP üzenetet tárolja.

Találmányunk továbbá adathordozó, amely egy rendszerben végrehajtott információ feldolgozási programot tárol, a rendszer tartalmaz egy információ terminál támogató szerveret, amely egy felhasználói oldal információ termináljába töltött böngésző és egy ügyintéző oldal információ termináljába töltött böngésző együttműködését támogatja. A program tartalmaz:

(a) program kódot, amely utasítja az információ terminál támogató szerveret egy, a felhasználói oldal böngésző kérésére válaszként elküldött HTTP üzenet fogadására;

(b) program kódot, amely utasítja az információ terminál támogató szerveret, hogy meghatározza, hogy a HTTP üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) program kódot, amely utasítja az információ terminál támogató szerveret, hogy editálja a HTTP üzenet tartalmát, amikor a HTTP üzenet egyezik az előre meghatározott feltétellel;

(d) program kódot, amely utasítja az információ terminál támogató szerveret, hogy az editálatlan HTTP üzenetet elküldje a felhasználói oldal böngészőjének; és

(e) program kódot, amely utasítja az információ terminál támogató szerveret, hogy az editált HTTP üzenetet elküldje az ügyintéző oldal böngészőjének.

A találmányunk szerinti adathordozó program kódja az editálást egy, az ügyintéző oldal információ termináljába betöltött kliens program számára szóló parancs beillesztéséhez végzi.

A beillesztett parancs az „ürlap betekintés tiltása”, „ürlap változtatás tiltása”, specifikus ürlap elrejtése” vagy „specifikus mező elrejtése” parancsok valamelyike.

Találmányunk továbbá adathordozó, amely egy rendszerben végrehajtott információ feldolgozási programot tárol, a rendszer egy információ terminál támogató szerverrel rendelkezik, amely egy tartalom szerverhez csatlakozó információ terminált támogat, és egy kliens programot tartalmaz. A program tartalmaz:

(a) program kódot, amely utasítja az információ terminál támogató szerveret, hogy fogadjon egy üzenetet a tartalom szervertől, válaszolva az információ terminál kérésére;

(b) program kódot, amely utasítja az információ terminál támogató szervert, hogy meghatározza, hogy az üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) program kódot, amely utasítja az információ terminál támogató szervert, hogy az üzenetet editálja egy, a kliens programnak szóló parancs beillesztéséhez, amikor az üzenet egyezik az előre meghatározott feltétellel; és

(d) program kódot, amely utasítja az információ terminál szervert, hogy az editált üzenetet elküldje az információ terminálnak.

Találmányunkat a csatolt ábrák alapján ismertetjük részletesen.

Az 1. ábra egy blokkvázlat, amely a találmány szerinti információ terminál támogató szerver vagy információ terminál hardver konfigurációjának egyik megvalósítását mutatja.

A 2. ábra a találmány előnyös megvalósítása szerinti együttműködő rendszert alkotó elemek blokkvázlata.

A 3. ábra a találmány előnyös megvalósítása szerinti információ terminál támogató rendszer működésének folyamatábrája.

A 4. ábra a találmány előnyös megvalósítása szerinti konfigurációs állományt mutatja.

Az 5. ábra az ismertetett együttműködés technológiát ábrázolja.

Az 1. ábra a találmányunkban alkalmazott 110 információ terminál támogató szerver hardver konfigurációjának egy megvalósítását mutatja. A 110 információ terminál támogató szerver tartalmaz egy 1 központi vezérlőegységet (CPU) és egy 4 memóriát.

Az 1 CPU és a 4 memória a 2 buszon keresztül egy segéd tárhoz csatlakozik, amely egy 13 merevlemez-meghajtó. Egy 20 hajlékonylemez-meghajtó (vagy egy 26, 28, 29 adathordozó-meghajtó, például egy MO vagy egy CD-ROM, stb.) a 19 hajlékonylemez-vezérlőn (vagy a 25 IDE vezérlőn vagy a 27 SCSI vezérlőn, stb.) keresztül a 2 buszhoz van csatlakoztatva.

A 20 hajlékonylemez-meghajtóba (vagy adathordozó-meghajtóba, mint például MO vagy CD-ROM) egy hajlékonylemez (vagy egy adathordozó, mint például MO vagy CD-ROM) van behelyezve. Ez a hajlékonylemez, stb., a 13 merevlemez-meghajtó vagy a 14 ROM, a tálmányunk megvalósításához, egy számítógép program kódját tartalmazza, amely utasításokkal látja el a CPU-t, stb., az operációs rendszerrel összhangban. A programot a végrehajtáshoz betápláljuk a 4 memóriába. Ennek a számítógép programnak a kódját tömöríthetjük, vagy több adathordozón megosztva rögzíthetjük.

A 110 információ terminál támogató szerver fel van szerelve felhasználói interfész hardverrel, amely tartalmazhat egy 7 mutatóeszközt (egér, joystick, stb.) vagy egy 6 billentyűzetet az adatbevitelhez, és egy 12 képernyőt a képi adatok megjelenítéséhez a felhasználó számára. A beviteli eszköz lehet érintéses panel is. Egy 16 párhuzamos porton keresztül csatlakoztathatunk egy nyomtatót, vagy egy 15 soros porton keresztül csatlakoztathatunk egy modemet is. A 110 információ terminál támogató szerver a 15 soros porton keresztül egy hálózathoz van csatlakoztatva, és a modem vagy egy 18 kommunikációs adapter kártya (Ethernet vagy token ring kártya) segítségével alkalmas arra, hogy más számítógépekkel kommunikáljon.

Egy 23 hangbemenet érzékelő egy 22 erősítőn keresztül vesz egy, a 21 hangvezérlővel átalakított D/A (digitális-analóg) átalakítású hangjelet. A 21 hangvezérlő analóg-digitális átalakítást végez a 24 mikrofonnal vett hang információn, hogy a rendszeren kívüli hang információt beviessük a rendszerbe.

Láthatjuk, hogy 110 információ terminál támogató szerver lehet egy olyan információ terminál, amely rendelkezik kommunikációs funkcióval, beleértve egy tipikus személyi számítógépet (PC), nagyteljesítményű számítógépet, hordozható számítógépet, kézi számítógépet vagy hálózati számítógépet, vagy ezek kombinációját. A konfigurációban ezek az alkotóelemek csak példa jellegűek, és nincs szükség az összes felsorolt alkotóelemre a találmány megvalósításához lényeges konfiguráció esetében.

Például, az ismertetett hardver konfigurációban a hangfeldolgozáshoz szükséges 21 hangvezérlő, 22 erősítő, 23 hangbemenet érzékelő és 24 mikrofon, a kezelő számára a bevitelt közvetlenül lehetővé tevő 6 billentyűzet, 7 egér és 5 billentyűzet/egér vezérlő, az adatok képi megjelenítését végző 12 CRT, 11 képernyő, 9 VRAM és 8 VGA, és az egyes 19, 25 és 27 adathordozó processzorok nem szükségesek az információ terminál támogatásához, ezért elhagyhatók.

A 110 információ terminál támogató szerver konfigurációjában számos módosítást végezhetünk, ez egy szakember számára nem jelenthet problémát, például az egyes elemeket több géppel kombinálhatjuk, és a funkciókat megosztott módon végezhetjük el, anélkül, hogy eltérnénk a találmány lényegétől.

A találmányunkban alkalmazott 110 információ terminál támogató szerverhez hasonlóan kialakíthatunk egy 130 információ terminált is, az 1. ábrán mutatott hardver konfigurációval, azaz a 130 információ terminál is alkalmas információ kérés bevitelére és ilyen kérés kibocsátására és fogadására. A 130 információ terminál megvalósítható egy tipikus személyi számítógéppel (PC), hordozható számítógéppel, kézi számítógéppel, különböző háztartási gépekkel, például egy beépített számítógéppel rendelkező televízióval, kommunikációs funkciókkal rendelkező játékgépekkel vagy olyan információ terminálokkal, amelyek kommunikációs funkciókkal rendelkeznek, például telefon, FAX, hordozható telefon, elektronikus notebook, stb. De a konfigurációnak ezek az elemei csak példa értékűek, nem lényegesek a találmány megvalósítása szempontjából.

A 110 információ terminál támogató szerver operációs rendszere nincs korlátozva egy specifikus operációs rendszer környezetre, lehet egy hálózati számítógépen telepített olyan operációs rendszer, amely egy szabványos GUI többablakos környezetet támogat, például Windows NT (Microsoft védjegy), Windows 9x (Microsoft védjegy), Windows 3.x (Microsoft védjegy), OS/2 (IBM védjegy), MacOS (Apple védjegy), Linux (Linus Torvalds védjegy) vagy X-WINDOW rendszer (MIT védjegy) AIX-en (IBM védjegy), karakter alapú környezet, mint például PC-DOS (IBM védjegy) vagy MS-DOS (Microsoft védjegy), valós idejű OS mint például VxWorks (Wind River Systems Inc. védjegy), vagy egy Java OS operációs rendszer, stb.

A 130 információ terminál operációs rendszere sincs korlátozva egy specifikus operációs rendszer környezetre, lehet egy hálózati számítógépen telepített olyan operációs rendszer, amely egy szabványos GUI többablakos környezetet támogat, például Windows NT (Microsoft védjegy), Windows 9x (Microsoft védjegy), Windows 3.x (Microsoft védjegy), OS/2 (IBM védjegy), MacOS (Apple védjegy), Linux (Linus Torvalds védjegy) vagy X-WINDOW rendszer (MIT védjegy) AIX-en (IBM védjegy), karakter alapú környezet, mint például PC-DOS (IBM védjegy) vagy MS-DOS (Microsoft védjegy), valós idejű OS mint például VxWorks (Wind River Systems Inc. védjegy), vagy egy Java OS operációs rendszer, stb.

A 2. ábra egy funkcionális blokkvázlat, amely a találmány előnyös megvalósítása szerinti 110 információ terminál támogató szerveret tartalmazó együttműködési rendszer konfigurációs rendszerét mutatja.

A találmány előnyös megvalósítása szerinti 100 együttműködési rendszer tartalmaz 130 és 132 információ terminálokat, egy 110 információ terminál támogató szerveret, és egy 140 Web szervert.

A találmány előnyös megvalósításában a 130 és 132 információ terminálokon egy 131 Web böngésző van telepítve. A 131 Web böngésző kijelöl egy URL-t, és egy kérelmet küld egy meghatározott Web szervernek. A 131 Web böngésző ezután fogadja a 140 Web szervertől küldött választ a képernyőn való megjelenítéshez. A 130 ügyintéző információ terminálon egy 135 kliens program van telepítve. A 135 kliens program a 133 Web böngésző API segítségével elolvassa a 131 Web böngészőbe táplált HTML tartalmat, és

a 133 Web böngésző API és egy OS API segítségével végrehajtja a parancsokat, amelyeket a proxy szerver cache kezelője ad.

A találmányunk szerinti 110 információ terminál támogató szerver rendelkezik egy 113 Web szerverrel/proxy szerverrel, egy 115 Web szerver/proxy szerver API-val és egy 120 cache kezelővel.

A 120 cache kezelő tárolja a HTML tartalmakat (HTTP válaszokat) a 131 Web böngészők által küldött HTTP kérelmekre. Amikor a két Web böngésző meg van osztva, akkor a 120 cache kezelő az interaktív módon küldött HTTP kérelmeknek megfelelően szabályozza ugyanannak a HTML adatnak a visszatérését. Találmányunk előnyös megvalósításában a 120 cache kezelő a proxy szerver vagy 115 Web szerver/proxy szerver API használatával van megírva, így az összes HTTP üzenetet figyelhetjük a 113 proxy szerverrel vagy Web szerverrel. A HTTP üzenetek kezelése és figyelése a HTTP üzenetek feldolgozását végző 129 szabályvezérlés irányító feladata.

A 129 szabályvezérlés irányító a 113 Web szervertől/proxy szervertől küldött üzeneteket továbbítja a 126 és 127 szabályvezérlőkhöz egy 128 konfigurációs állománnyal kijelölve. A 129 szabályvezérlés irányító több 128 konfigurációs állományt jelölhet ki, és egy kijelölt sorrendben hívhatja a 126 és 127 szabályvezérlőket.

A 126 és 127 szabályvezérlők tartalmazznak egy 125 HTTP üzenet ellenőrzőt és egy 123 HTTP üzenetszerkesztőt. A 125 HTTP üzenet ellenőrző a 121 szabály meghatározás részre hivatkozva dönt egy HTTP üzenet editálásáról. Amikor egy üzenet editálása mellett dönt, akkor ezt a HTTP üzenetet elküldjük a 123 HTTP

üzenetszerkesztőhöz. A 123 HTTP üzenetszerkesztő a 121 szabály meghatározás rész tartalmának megfelelően editálja a HTTP üzenetet, az editált üzenetet eltárolja egy 163 ügyintéző cache-tárban, és az editálás eredményét visszajuttatja a 129 szabályvezérlés irányítóhoz. Amikor úgy dönt, hogy nem editál egy üzenetet, akkor ezt az üzenetet úgy, ahogy van, visszajuttatjuk a 129 szabályvezérlés irányítóhoz. A 121 szabály meghatározás rész tartalmaz egy feltétel felállítás részt és egy parancs felállítást részt, és meghatározza, hogy milyen feltételek mellett és milyen módon kell editálni a HTTP üzeneteket.

A 2. ábrával kapcsolatban ismertetett funkcionális blokkok logikai funkcionális blokkok. Nem szükségszerű, hogy ezeket a funkcionális blokkokat hardverrel és szoftverrel egyenként meg kell valósítanunk, a funkcionális blokkok megvalósíthatók hardver és szoftver kombinációkkal vagy megosztott hardverrel és szoftverrel. Magától értetődik, hogy a 2. ábrán mutatott összes funkcionális blokk nem elengedhetetlenül fontos a találmány megvalósításához szükséges konfigurációban.

A 3. ábra egy folyamatábra, amelyen a találmány előnyös megvalósítása szerinti 100 információ terminál támogató rendszer működésének lépéseit mutatjuk. A folyamat a 400 lépéstől indul. Ekkor, a 130 ügyintéző oldal információ terminál és a 132 felhasználói oldal információ terminál 131 Web böngészője a 110 információ terminál támogató szerverhez van csatlakoztatva.

A találmány szerinti előnyös megvalósításban, amikor a 132 felhasználói oldal információ terminál 131 Web böngészője egy kérelmet küld az együttműködő szerver Web szerverének egy megha-

tárolt URL-ért, akkor egy felhasználó specifikus UAI-t (felhasználó azonosító információ) generálunk a 110 információ terminál támogató szervernél, és ezt a HTTP válasz fejlécénél „Cookie”-ként tüntetjük fel, és elküldjük a felhasználói oldal Web böngészőjének.

Az ügyintéző oldalon egy együttműködés szoftvert indítunk el, és az együttműködési szerver fel tudja ismerni a felhasználó ID és a jelszó regisztrálásából, hogy a 131 Web böngésző, amely egy ügyintéző specifikus ID-vel rendelkezik, csatlakoztatva van. Ezután, a generált összes HTTP kérelem fejléce tartalmaz egy „cookie” parancsot. Ennek eredményeképpen, létrehozhatjuk az ügyintéző jelenlétét és a felhasználói oldal Web böngészőjét egyértelműen azonosító UAI információt. A felhasználót és a megfelelő ügyintézőt egy ügyintéző kezelő táblázat segítségével egyeztetjük egymással. Ez a megoldási mód jól ismert, ezért nem ismertetjük részletesen.

Ebben a kezdeti állapotban, a 401 lépésben, a 132 felhasználói oldal információ terminál 131 Web böngészője a felhasználó által kijelölt URL-t egy HTTP üzenetté alakítja, és ezt a HTTP üzenetet elküldi a 110 információ terminál támogató szervernek. A 403 lépésben a 110 információ terminál támogató szerver fogadja ezt a HTTP üzenetet, és továbbítja azt a HTTP üzenettel kijelölt 140 Web szervernek. A HTTP üzenetet egy konfigurációs állományban kijelölt Web szerverhez továbbítjuk egy együttműködési szerver esetében.

A 140 Web szerver ezután, a 405 lépésben feldolgozza a kapott HTTP üzenetet, és egy új HTTP üzenetet generál, amelyet

visszajuttat a 110 információ terminál támogató szerverhez, a 407 lépésben.

A 110 információ terminál támogató szerver a 140 Web szervertől kapott HTTP üzenetet a 161 felhasználói cache-tárban tárolja, és a 409 lépésben továbbítja a HTTP üzenetet a 129 szabályvezérlés irányítónak, amely a HTTP üzenet feldolgozását végzi. A 129 szabályvezérlés irányító felelős azért, hogy a hívott és elküldött HTTP üzeneteket a 128 konfigurációs állománnyal kijelölve, a 126 és 127 szabályvezérlők által kért sorrendben dolgozza fel (411 lépés). A konfigurációs állományban több, a 4. ábrán mutatott formátumú szabályvezérlőt jelölhetünk ki.

A 126 és 127 szabályvezérlők maguk Osztott Objektum (Shared Object) adatok (Unix-ban) vagy Dinamikus Kapcsolat Könyvtár (Dynamic Link Library) adatok (Windows-ban). A találmány előnyös megvalósításában ezt a programozási interfészt a következő módon határoztuk meg.

1. Táblázat

```
int CscPluginCInit (const char *pszConf)
```

Ez a függvény akkor kerül meghívásra, amikor a Cache kezelőt elindítjuk CscPlhdrInit0-n keresztül.

argumentum

const char : a HTML plugin konfigurációs állomány elérési

útja

visszatérési érték

0 : OK

4900-4999 : hiba

```
int CscPluginCConv ( const CsmConvParam param,
const char      *pIn,
const int       isize,
char            **ppOut,
int             *osize )
```

Ez a függvény kerül meghívásra egy HTML tartalom átalakításához CscPlhdrConv0-n keresztül.

argumentum

```
const CsmConvParam      : információ átalakításra
const char *pIn          : bemeneti tartalom
const int isize          : bemeneti tartalom mérete
char **ppOut             : kimeneti tartalom
int *osize               : kimeneti tartalom mérete
```

viisszatérési érték

0 : OK

4900-4999 : hiba

```
void CsmPluginCFree ( char *p)
```

Ez a függvény kerül meghívásra a CscPluginCConv () ppOut-jának felszabadításához a CscPlhdrFree0-n keresztül.

argumentum

```
char *                : CscPluginCConv () ppOut-ja
```

viisszatérési érték

Nincs

```
int CscPluginCTerminate ()
```

Ez a függvény kerül meghívásra, amikor a Cache Kezelőt leállítjuk CscPlhdrdrTerminate ()-on keresztül.

argumentum

Nincs

viSSzatérési érték

0 : ok

4900-4999 : hiba

A 125 HTTP üzenet ellenőrzőt és a 123 HTTP üzenetszerkesztőt tartalmazó 126 és 127 szabályvezérlők megkezdik a HTTP üzenet ellenőrzését, amikor a 129 szabályvezérlés irányító egy üzenetet küld, és meghatározzák, hogy ez a HTTP üzenet editálásra kerül-e vagy sem a 121 szabály meghatározó résznek megfelelően (413 lépés).

A szabály meghatározás részt a következő formátum használatával jelöljük ki. Példánkban a szabály meghatározás rész tartalmaz egy feltétel felállítás részt és egy parancs felállítás részt. A parancs felállítás részt beállítjuk URL = <url name> - re, és a 125 üzenet ellenőrző megállapítja, hogy a HTTP üzenet megfelel-e vagy sem ennek a feltételnek.

Példánkban egy olyan parancsot jelölünk ki, amely egy meghatározott URL-t illetően megtilt egy meghatározott műveletet, de a HTTP üzenetet analizálhatjuk, és felállíthatunk egy feltételt, hogy a 125 HTTP üzenet ellenőrző meghatározza, van-e egyezés egy meghatározott feltétellel vagy sem (például, egy küldés billentyű, egy bemenet mező vagy meghatározott erőszakos vagy obszcén karakterek vagy képek jelenléte).

URL=<url name>;

AgentActionProhibited=<command>[<command>...] <command>=<submit

```
alterform |  
notshown:[formname=<formname>,  
fieldname=<fieldname>[,<fieldname>...]  
>;
```

Ebben a példában az űrlap megtekintését, az űrlap változtatását és specifikus űrlapok vagy mezők elrejtését jelöljük ki. Amikor a 125 HTTP üzenet ellenőrző azt állapítja meg, hogy egy változtatásra van szükség, akkor a 161 felhasználói cache-tárban levő adatot átmásoljuk a 163 ügyintéző cache-tárba. Ezután meghívjuk a 123 HTTP üzenetszerkesztőt, és elvégezzük az editálást a 121 szabály meghatározás részben kijelölt információnak megfelelően (415 lépés).

A találmány előnyös megvalósításában a 123 HTTP üzenetszerkesztő a következő parancsokat ágyazza be a 163 ügyintéző cache-tárban levő HTTP üzenetbe.

```
<!--#CP_WebC_AgentActionProhibited=<command>[<command>...]--  
><command>=<submit |  
  
alterform |  
  
notshown: [formname=<formname>;]  
  
fieldname=<fieldname>[,<fieldname>...]  
  
>;
```

A 110 információ terminál támogató szerver ezután a HTTP üzenetre egy választ küld a 131 Web böngészőnek a 417 lépésben. Ekkor, a 161 felhasználói cache-tárban levő HTTP üzenetet elküldjük a 132 felhasználói információ terminál 131 Web böngészőjének, és a 163 ügyintéző cache-tárban levő HTTP üzenetet el-

küldjük a 130 ügyintéző információ terminál 131 Web böngészőjének.

A 130 ügyintéző oldali információ terminál 131 Web böngészője megjeleníti a kapott HTTP üzenetet, és értesítést ad a 135 kliens programnak arról, hogy elvégezte ezt a műveletet (419 lépés). Adott esetben, a 135 kliens program jelzi a megjelenítés vége eseményt a 131 Web böngészőnek.

A 135 kliens program elolvassa a beágyazott parancsot a Web böngésző API-jának használatával, és végrehajtja ezt a parancsot a 421 lépésben. Példánkban a 135 kliens program a Web böngésző API-jának vagy az OS használatával utasítást ad az űrlap megtekintés és űrlap változtatás tiltására, és adott űrlapok és mezők elrejtésére. Találmányunk előnyös megvalósításában a következő feldolgozást hajtjuk végre a kliens oldalon, amikor a parancs „NOTSHOWN” (azaz meghatározott információt nem jelenít meg).

Azt a mezőt, amelyet az ügyintéző oldalon nem jelenítünk meg, FieldA-nak vesszük. Amikor egy felhasználó adatokat visz be egy bizonyos mezőbe, akkor a kliens program a bevitt adatokat úgy továbbítja, ahogy azok vannak, de amikor az adatokat a FieldA-ba visszük be, akkor a bevitt adatok helyett ugyanannyi számú „*” szimbólumot továbbítunk az ügyintéző oldal kliensnél való tárolásra, ahol a „*” szimbólumok kerülnek a FieldA-ba. Az ily módon történő adatfeldolgozással különösen nagy biztonságot tudunk nyújtani.

A fentiekben találmányunkat együttműködési technológia vonatkozásában ismertettük egy példán keresztül. Mindamellett, találmányunkban a 140 Web szervertől küldött HTTP üzenetet anali-

záljuk, és amikor egyezést találunk egy meghatározott feltétellel, akkor ezt az üzenetet editáljuk és elküldjük a kliens oldalnak. Ezért ez a technológia nemcsak az együttműködés esetében, de más területeken való használatra is alkalmas, úgymint Internetes oktatás esetében, ahol például a helyes válasz mezőt lefedhetjük, amíg egy meghatározott feltétellel való egyezés létre nem jön. Továbbá, találmányunk előnyös megvalósítását Internet/intranet példáján keresztül ismertettük, de ezt a technológiát alkalmazhatjuk olyan kommunikációs rendszerekben is, ahol nem Internet protokollt, hanem más protokollokat használnak. Ezen felül, találmányunk előnyös megvalósításában egy paranccsal beillesztett kontextust küldünk az ügyintéző oldal Web böngészőjének, és ezt a parancsot a kliens program hajtja végre. De az is lehetséges, hogy a paranccsal beillesztett kontextust a felhasználói oldal Web böngészőjének küldjük el, azért, hogy megakadályozzuk, hogy a felhasználó meghatározott műveleteket hajtson végre, és a kliens programmal hajtjuk végre ezt a parancsot, vagy mind a felhasználóval, mind az ügyintézővel végrehajtjuk a parancsot.

Egy olyan együttműködési rendszert ismertettünk, amely alkalmas arra, hogy kontextusnak megfelelően megváltoztassa az együttműködő kliensek viselkedését Interneten való együttműködés során. A találmányunk szerinti rendszer alkalmas arra, hogy egy klientsztől érkező kérésre reagálva egy adminisztrátor szándékainak megfelelően megváltoztasson egy szervertől küldött tartalmat.

Szabadalmi igénypontok

1. Információ feldolgozási eljárás egy információ feldolgozó rendszerben, amely tartalmaz egy, a felhasználói oldal információ termináljába betöltött böngésző együttműködését támogató információ terminál támogató szervert, és egy, az ügyintéző oldal információ termináljába betöltött böngészőt, azzal jellemezve, hogy az eljárás során a következő lépéseket tesszük:

(a) az információ terminál támogató szervernél fogadunk egy, a felhasználói oldal böngésző kérésére válaszulva elküldött HTTP üzenetet;

(b) meghatározzuk, hogy a HTTP üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) ha a HTTP üzenet egyezik az előre meghatározott feltétellel, akkor a HTTP üzenet tartalmát editáljuk;

(d) az editálatlan HTTP üzenetet elküldjük a felhasználói oldal böngészőjének; és

(e) az editált HTTP üzenetet elküldjük az ügyintéző oldal böngészőjének.

2. Az 1. igénypont szerinti információ feldolgozási eljárás, azzal jellemezve, hogy a (c) lépésben az editálást egy, az ügyintéző oldal információ termináljába betöltött kliens program számára szóló parancs beillesztéséhez végezzük.

3. A 2. igénypont szerinti információ feldolgozási eljárás, azzal jellemezve, hogy a parancs az „űrlap megtekintés tiltása”, „űrlap változtatás tiltása”, specifikus űrlap elrejtése” vagy „specifikus mező elrejtése” parancsok valamelyike.

4. Információ feldolgozási eljárás egy információ feldolgozó rendszerben, amely egy információ terminál támogatásához egy tartalom szerverhez kapcsolódó információ terminál támogató szerverrel rendelkezik, és egy kliens programot tartalmaz, azzal jellemezve, hogy az eljárás során a következő lépéseket tesszük:

(a) az információ terminál támogató szervernél fogadunk egy üzenetet a tartalom szervertől, válaszolva az információ terminál kérésére;

(b) meghatározzuk, hogy az üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) ha az üzenet egyezik az előre meghatározott feltétellel, akkor az üzenetet editáljuk egy, a kliens programnak szóló parancs beillesztéséhez; és

(d) az editált üzenetet elküldjük az információ terminálnak.

5. Információ terminál támogató szerver, amely egy felhasználói oldal információ termináljába töltött böngésző és egy ügyintéző oldal információ termináljába töltött böngésző együttműködését támogatja, azzal jellemezve, hogy az információ támogató szerver tartalmaz:

(a) egy szabály meghatározás részt, amely tartalmaz egy feltétel felállítás részt és egy parancs felállítás részt;

(b) egy szabályvezérlés irányítót az információ támogató szervernél a felhasználói oldal kéréseire válaszként küldött HTTP üzenetek figyelésére;

(c) egy HTTP ellenőrt, amely meghatározza, hogy a HTTP üzenet egyezik-e vagy sem a szabály meghatározás rész feltétel felállítás részének feltételével;

(d) egy HTTP szerkesztőt, amely a HTTP üzenet tartalmát a szabály meghatározás rész parancs felállítás részének tartalma szerint editálja, amikor a HTTP üzenet egyezik egy előre meghatározott feltétellel;

(e) felhasználói cache-tárt, amely a felhasználói oldal böngészőjének küldött editálatlan HTTP üzenetet tárolja; és

(f) ügyintéző cache-tárt, amely az ügyintéző oldal böngészőjének küldött editált HTTP üzenetet tárolja.

6. Az 5. igénypont szerinti információ terminál támogató szerver, azzal jellemezve, hogy a HTTP szerkesztő az editálást egy, az ügyintéző oldal információ termináljába betöltött kliens program számára szóló parancs beillesztéséhez végzi.

7. A 6. igénypont szerinti információ terminál támogató szerver, azzal jellemezve, hogy a parancs az „űrlap megtekintés tiltása”, „űrlap változtatás tiltása”, specifikus űrlap elrejtése” vagy „specifikus mező elrejtése” parancsok valamelyike.

8. Információ terminál támogató szerver, amely egy tartalom szerverhez kapcsolódó információ terminált támogat, és egy kliens programmal rendelkezik, azzal jellemezve, hogy az információ terminál támogató szerver tartalmaz:

(a) egy üzenet ellenőrt, amely meghatározza, hogy az információ terminál kérésének eleget téve a tartalom szervertől küldött üzenet egyezik-e vagy sem egy előre meghatározott feltétellel; és

(b) egy üzenetszerkesztőt, amely a kliens programot utasító parancsot beillesztve editálja az üzenetet, amikor az üzenet egyezik az előre meghatározott feltétellel.

9. Együttműködési rendszer, amely egy felhasználói oldal információ termináljába töltött böngésző és egy ügyintéző oldal információ termináljába töltött böngésző együttműködését támogatja, azzal jellemezve, hogy az együttműködő rendszer tartalmaz:

(a) egy, a felhasználói oldal információ termináljával együttműködő ügyintéző oldal információ terminált;

(b1) egy szabály meghatározás részt, amely tartalmaz egy feltétel felállítás részt és egy parancs felállítás részt;

(b2) egy szabályvezérlés irányítót egy, a felhasználói oldal kéréseire válaszként küldött HTTP üzenet figyelésére;

(b3) egy HTTP ellenőrt, amely meghatározza, hogy a HTTP üzenet egyezik-e vagy sem a szabály meghatározás rész feltétel felállítás részének feltételével;

(b4) egy HTTP szerkesztőt, amely a HTTP üzenet tartalmát a szabály meghatározás rész parancs felállítás részének tartalma szerint editálja, amikor a HTTP üzenet egyezik egy előre meghatározott feltétellel;

(b5) felhasználói cache-tárt, amely a felhasználói oldal böngészőjének küldött editálatlan HTTP üzenetet tárolja; és

(b6) egy információ terminál támogató szerveret, amely tartalmaz egy ügyintéző cache-tárt, amely az ügyintéző oldal böngészőjének küldött editált HTTP üzenetet tárolja.

10. Információ feldolgozó program egy olyan rendszerben való alkalmazáshoz, amely tartalmaz egy információ terminál támogató szerveret, amely egy felhasználói oldal információ termináljába töltött böngésző és egy ügyintéző oldal információ termináljába

töltött böngésző együttműködését támogatja, azzal jellemezve, hogy a program tartalmaz:

(a) program kódot, amely utasítja az információ terminál támogató szervert egy, a felhasználói oldal böngésző kérésére válaszként elküldött HTTP üzenet fogadására;

(b) program kódot, amely utasítja az információ terminál támogató szervert, hogy meghatározza, hogy a HTTP üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) program kódot, amely utasítja az információ terminál támogató szervert, hogy editálja a HTTP üzenet tartalmát, amikor a HTTP üzenet egyezik az előre meghatározott feltétellel;

(d) program kódot, amely utasítja az információ terminál támogató szervert, hogy az editálatlan HTTP üzenetet elküldje a felhasználói oldal böngészőjének; és

(e) program kódot, amely utasítja az információ terminál támogató szervert, hogy az editált HTTP üzenetet elküldje az ügyintéző oldal böngészőjének.

11. Adathordozó, azzal jellemezve, hogy a 10. igénypont szerinti program kódokat tartalmazza.

12. A 11. igénypont szerinti adathordozó, azzal jellemezve, hogy az editálási program kód az editálást egy, az ügyintéző oldal információ termináljába betöltött kliens program számára szóló parancs beillesztéséhez végzi.

13. A 12. igénypont szerinti adathordozó, azzal jellemezve, hogy a beillesztett parancs az „űrlap megtekintés tiltása”, „űrlap változtatás tiltása”, specifikus űrlap elrejtése” vagy „specifikus mező elrejtése” parancsok valamelyike.

14. Információ feldolgozó program egy olyan rendszerben való alkalmazáshoz, amely egy tartalom szerverhez csatlakozó, és egy kliens programot tartalmazó információ terminált támogató információ terminál támogató szerverrel rendelkezik, azzal jellemezve, hogy a program tartalmaz:

(a) program kódot, amely utasítja az információ terminál támogató szerveret, hogy fogadjon egy üzenetet a tartalom szervertől, válaszolva az információ terminál kérésére;

(b) program kódot, amely utasítja az információ terminál támogató szerveret, hogy meghatározza, hogy az üzenet egyezik-e vagy sem egy előre meghatározott feltétellel;

(c) program kódot, amely utasítja az információ terminál támogató szerveret, hogy az üzenetet editálja egy, a kliens programnak szóló parancs beillesztéséhez, amikor az üzenet egyezik az előre meghatározott feltétellel; és

(d) program kódot, amely utasítja az információ terminál szerveret, hogy az editált üzenetet elküldje az információ terminálnak.

15. Adathordozó, azzal jellemezve, hogy a 14. igénypont szerinti program kódokat tartalmazza.

A meghatalmazott

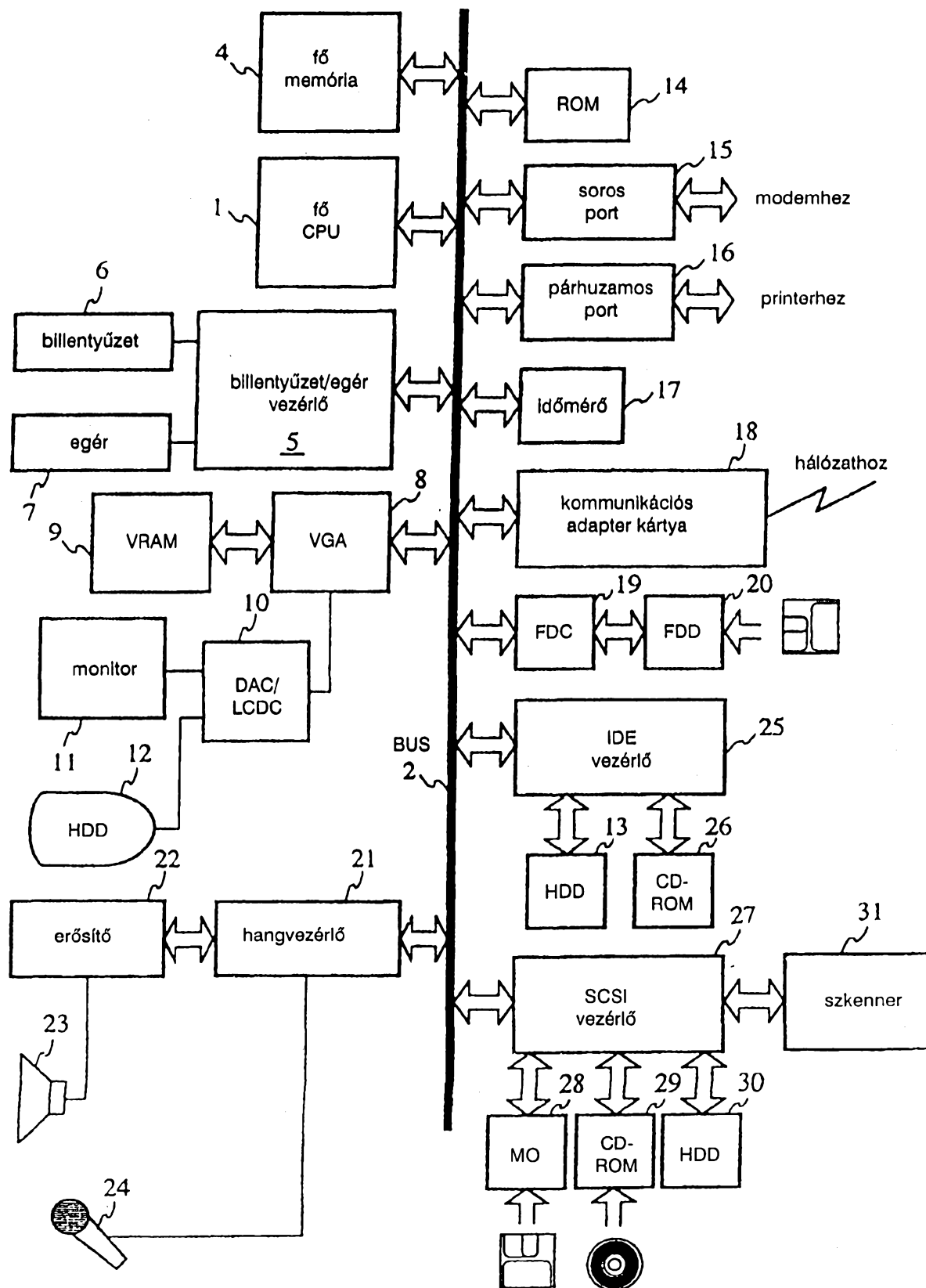
Köteles

Dr. Köteles Zoltán

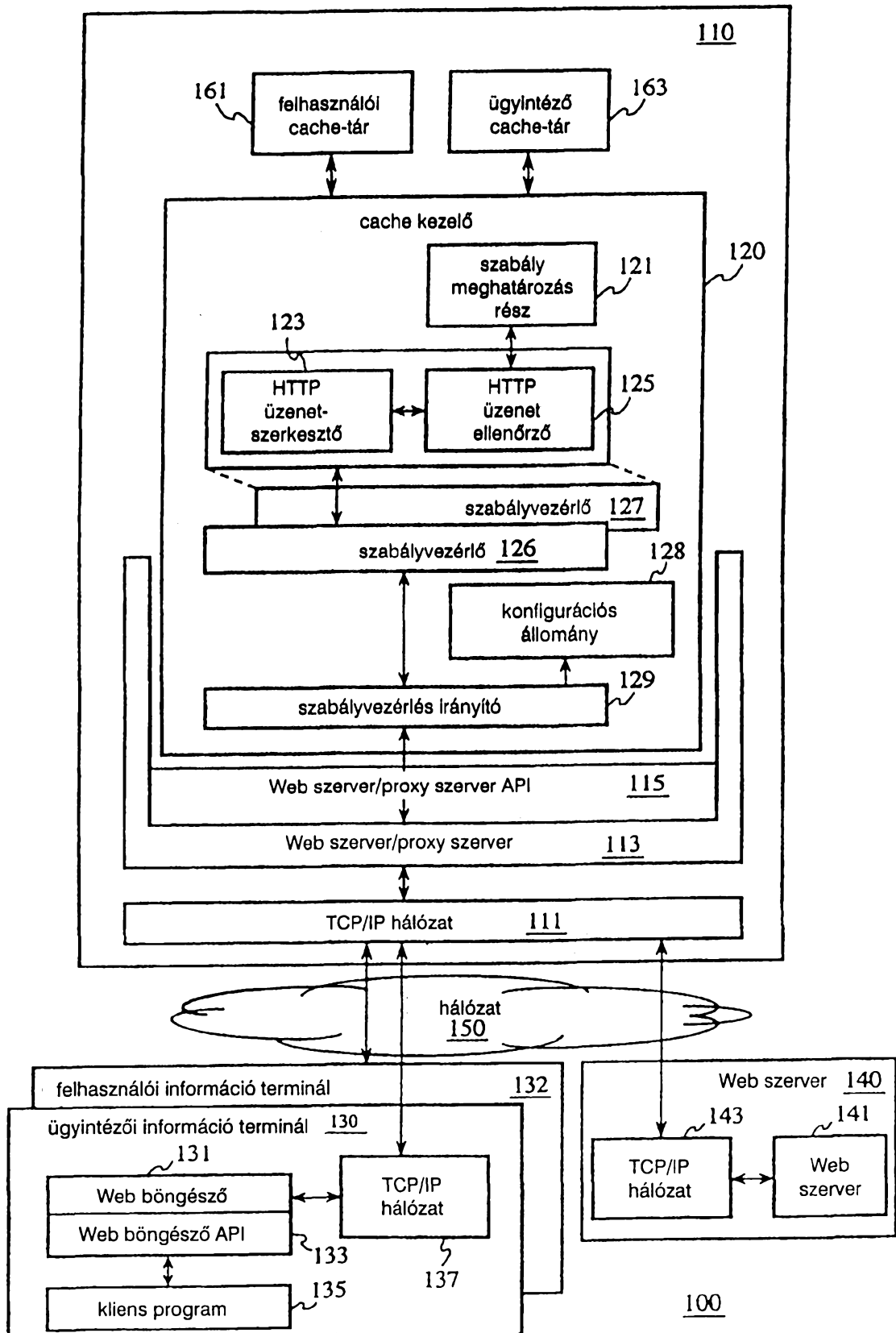
[Signature]

Dr. Köteles Zoltán
szabadalmi ügyvivő
az S.B.G. & K. Szabadalmi Ügyvivői Iroda
tagja
H-1062 Budapest, Andrássy út 113.
Telefon: 461-1000 Fax: 461-1099

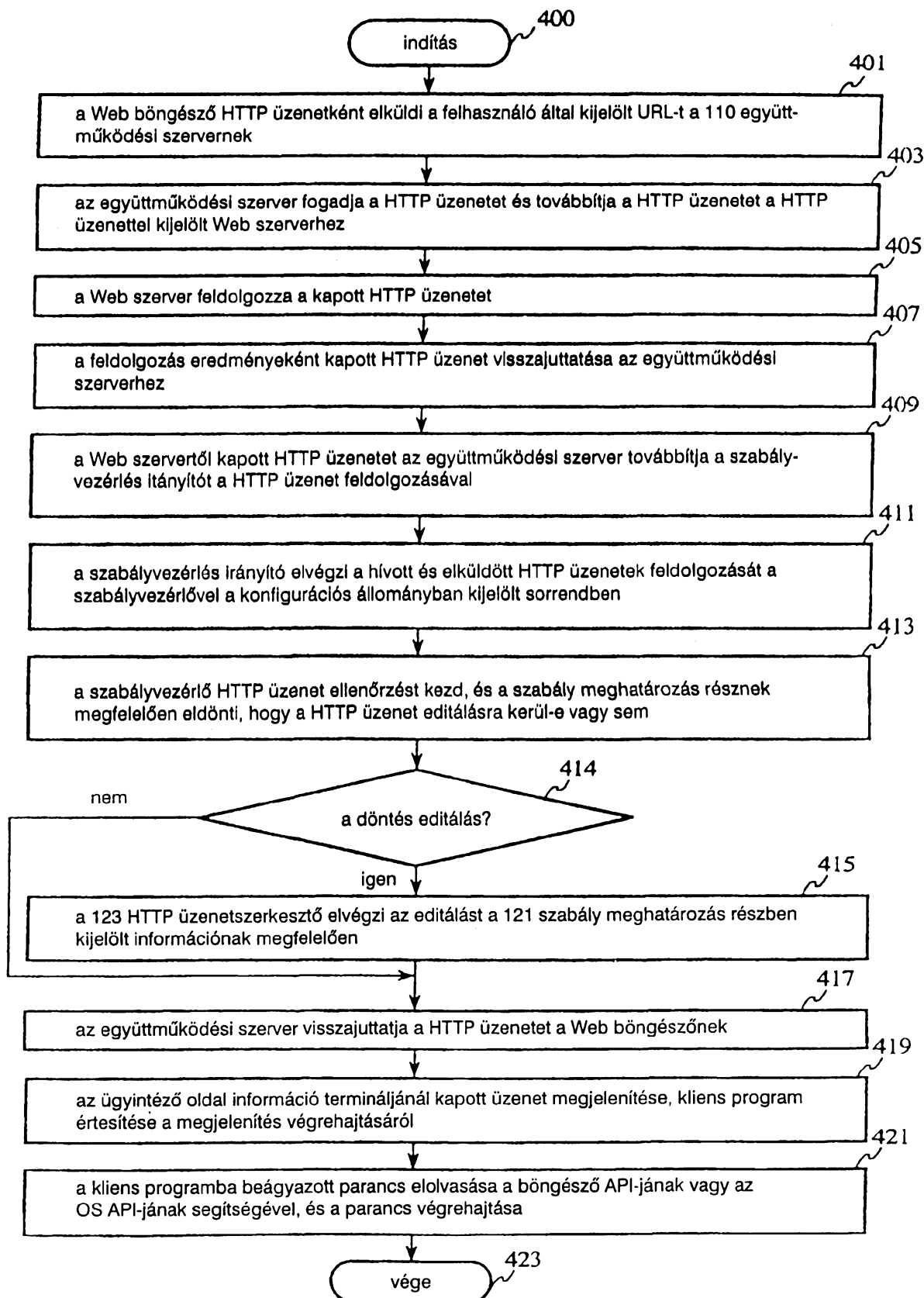
1. ábra



2. ábra



3. ábra



4. ábra

CscHTMLRuleHndlr=/usr/lpp/ibmctf/lib/csmplhdr.so

5. ábra

