

VoIP-szolgáltatások hibamenedzsmentje

Varga Pál^{*}, Moldován István^{*} és Molnár Gergely^{}**

^{*} Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék
1117 Budapest, Magyar Tudósok körútja 2.
Tel: +36 1 4633424, {pvarga,[moldovan](mailto:moldovan@tmit.bme.hu)}@tmit.bme.hu

^{**} Ericsson Magyarország Kft.
1117 Budapest, Irinyi József u. 4-20.
Tel: +36 1 4377763, gergely.molnar@ericsson.com

A VoIP szolgáltatásokat használók számának növekedésével a szolgáltatóknak szüksége van kifejezetten a VoIP-szolgáltatásokra specializálódott hibamenedzsment-alkalmazásokra. Ebben a tanulmányban egy olyan átfogó hibamenedzsment rendszer kerül bemutatásra, amely – a VoIP-szolgáltatók igényeire szabottan – mind a hibadetektálás és hibafeldolgozás, mind pedig a hibaok-meghatározás és a hibajavítási javaslattétel során felmerülő kérdésekre képes hatékony választ adni. A bemutatásra kerülő rendszer ötvözi a szabály alapú hibafeldolgozási algoritmusok, valamint a modell alapú hibaok-meghatározó módszerek sajátosságait, továbbá kihasználja a folyamatok konkurens végrehajtását segítő Petri-hálózatos leírás előnyeit is.

Bevezetés

A Voice over IP (VoIP) hálózatok szolgáltatásainak fennakadásmentes működtetéséhez célszerű átfogó hibamenedzsment rendszert (Fault Management System, FMS) üzemeltetni. Ez a hálózati elemek által jelzett események folyamatos gyűjtése és feldolgozása során képes kiszűrni az egyes hibákat, majd javaslatot tud adni a hibaokok helyére, illetve a hibaelhárítás lépéseire. A hálózat működését felügyelő operátor munkája ezzel jelentősen egyszerűsödik, ám sohasem válik feleslegessé, hiszen az összetettebb hibák felismerése és kiküszöbölése továbbra is az ő feladata marad.

A VoIP-szolgáltatás minőségi mutatóit a hálózati elemek, az IP-hálózat mint üzemeltetendő entitás, valamint a VoIP-hoz kötődő alkalmazások nem kívánt működéséből fakadó hibák is leronthatják. A fenti elemek állapotában bekövetkező „normális” események illetve hibák korrelációjának vizsgálatához egységes kezelési módot kell biztosítanunk a hálózat különféle esemény-bejelentő üzenetei számára (pl.: hálózati elemek hibái, hívásjegyek előállításánál bekövetkező események). Mivel egy adott hibajegy – pl.: alkalmazás-szerver nem elérhető –, több helyen és módon is rögzítésre kerülhet, a hibák feltárása ezen redundáns adatokból igen összetett feladat.

Hibamenedzsment alatt azt a folyamatot értjük, amely során a hibasokaságot az operátor a hibadetektálás, hibajel-feldolgozás, hibaok-meghatározás és hibajavítás lépésein keresztül

megszünteti. A hibadetektálás folyamata során a hálózat különböző eseményüzenet-forrásainak kimenetét gyűjtjük egybe, és alakítjuk azonos formátumúvá. Az így kialakított hibajel adatbázist a hibajel-feldolgozó használja. Ez céljainknak megfelelő szűréseket, eseménykorrelációs- és trendanalízis-műveleteket végez az adatbázis elemein. A hibaok-meghatározás már csak egy csökkentett, letisztázott adatbázisra történik. A folyamat eredményeként a rendszer hibajavítási javaslatokat tesz. Az egyes lépéseknek megfelelő hibamenedzsment-funkciók kialakítása során felmerülő kérdéseket a későbbi fejezetekben ismertetjük [1].

A bemutatásra kerülő VoIP hibamenedzsment-rendszer prototípusát a BME Távközlési és Médiainformatikai Tanszékének próbahálózatán üzemeltük be, az Ericsson Magyarország Kft. és a Kovax'95 Kft. munkatársainak közreműködésével. A hívásadatokból kinyerhető hiba-információ elemzését a NIIFI VoIP-hálózatából származó anonimizált adatsorokon végeztük.

Hibamenedzsment

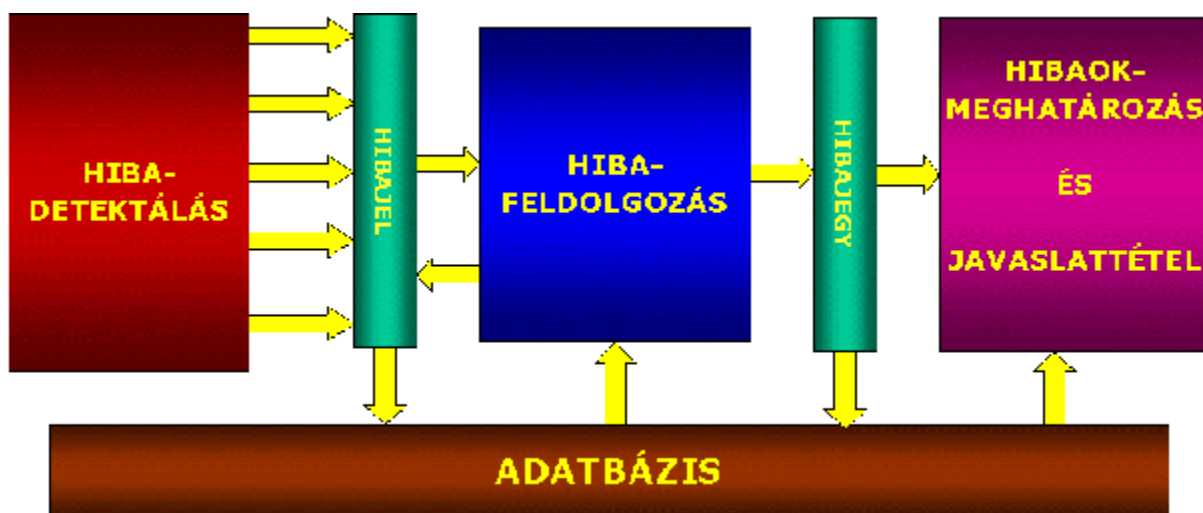
A hibamenedzsmentnek az alábbi funkciókat kell megvalósítania:

- észleli, és összegyűjti a hálózat működése során kialakuló hibüzeneteket,
- segít a hibüzenetek szűrésében,
- diagnosztikai tesztek futtat a hibaforrás felderítésére, javaslatot tesz a hibajavítására.

A hálózatmenedzselő alkalmazás a felügyeleti központban figyeli a hálózatot, hibajelenségekre „vadászva”. A megjelenő hibákat a

- hibadetektálás,
- hibajel-feldolgozás,
- hibaok-meghatározás és hibajavítás

folyamaton keresztül kezeli (lásd 1. ábra). A folyamat eredménye egy hibajavítási javaslat. Az operátor ez alapján elkezdheti a hiba okának megszüntetését. Ha a hibaok-meghatározás során a rendszer nem képes egészen pontosan körvonalazni a hiba okát, a folyamat naplójának vizsgálatával az operátor láthatja az elvégzett lépéseket és azok eredményeit, így ezek megisméltése nélkül további irányokban keresheti a hiba okát.



1. ábra A hibamenedzsment folyamatának elemei és ezek egymásra hatása

A folyamat elemeire való hivatkozás megkönnyítése érdekében érdemes definiálni a hibajel és a hibajegy fogalmát. Hibajel alatt a hibadetektálás során keletkező esemény-jelzéseket értjük.

Ez magában foglalja a hálózati elemek által küldött státusz-változások eseményjelzéseit (nem csak a figyelmeztetéseket vagy konkrét hibajelzéseket), valamint a hibamenedzsment rendszer aktív elemeinek (próbahívó, aktív monitor, stb.) eseményjelzéseit. Hibajegy alatt a hibafeldolgozó alrendszer kimeneti jelzéseit értjük, azaz azokat a hibára utaló figyelmeztetéseket, amelyekkel vagy a hibaok-meghatározó rendszernek, illetve az üzemeltető operátornak feltétlenül foglalkoznia kell [1].

Hibadetektálás

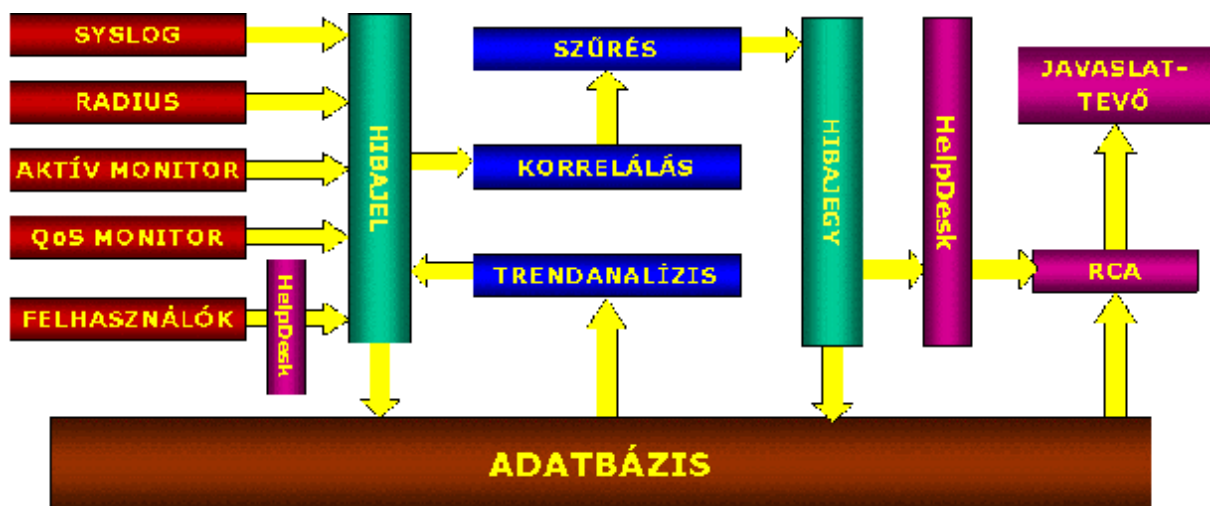
Célunk kifejezetten a VoIP szolgáltatást hátrányosan érintő hibajelenségek minél hamarabbi érzékelése és továbbítása hibafeldolgozásra. A hibajel-adatbázis kialakításához a hálózati elemek által küldött státusz-információt tartalmazó üzenetek mellett az a hibamenedzsment-rendszer aktív ellenőrző elemeinek jelzéseit is felhasználjuk. Ennek az adatbázisnak a kialakításánál már rögtön azzal a nehézséggel találjuk magunkat szemben, hogy a különböző hibajel-források által küldött hibajelek különböző formátumban találhatóak. A későbbi munka megkönnyítése érdekében ezeket a hibajeleket egységes formátumba rendeztük, illetve egységes kezelői felületet biztosítanunk a vizsgálatukra. Az így átalakított hibajeleket már könnyen tudja kezelni a rendszer a hibafeldolgozás során.

A VoIP-szolgáltatások hibamenedzsmentje során felmerülő hibajelek a következők:

- a hálózatban használt hibadetektáló elemek használata során kiszűrt specifikus VoIP hibaüzenetek (Syslog, QoS monitor (próbahívó) [2,3]),
- a VoIP hívások során keletkező hívás rekordok (üzenetek) (Radius rekordok),
- monitorozó elem használata közben keletkezett üzenetek (Aktív monitor),
- felhasználók által jelzett hibák (HelpDesk rendszer).

Hibafeldolgozás

A hibajel-adatbázisba esemény típusú rekordok kerülnek, ezek képezik a hibafeldolgozás bemeneti adathalmazát. Az adatbázis manuális feldolgozása meghaladja az üzemeltető operátor lehetőségeit, hiszen egy absztrakt és valódi értelemben is többszörözött, másodpercenként akár több tucatnyi rekorddal növekvő adathalmazról van szó. A hibafeldolgozás egy automatizált adatfeldolgozó folyamat, melynek eredményeként a keletkezett hibajegyeket (alarmokat) tárjuk csak az operátor elé. Ez már lényegkiemelt, érthető adathalmaz. A hibajelek feldolgozásának menetét a 2. ábra szemlélteti.



2. ábra A hibajelek feldolgozásának folyamata

A hibajelek az adatbázisba és a korrelátor bemeneti FIFO-tárolójába is bekerülnek. Az adatbázisba került hibajeleket a trendanalizátor folyamatosan elemzi – ennek eredményeképpen új, trend-jellegű hibajelet generál. A redundancia-mentesítést, és a „lényeg kiemelését” a szűrő modul végzi. A hibafeldolgozó modulok (szűrő, korrelátor, trendanalizátor) működését a következő fejezetekben vázoljuk.

Szűrő

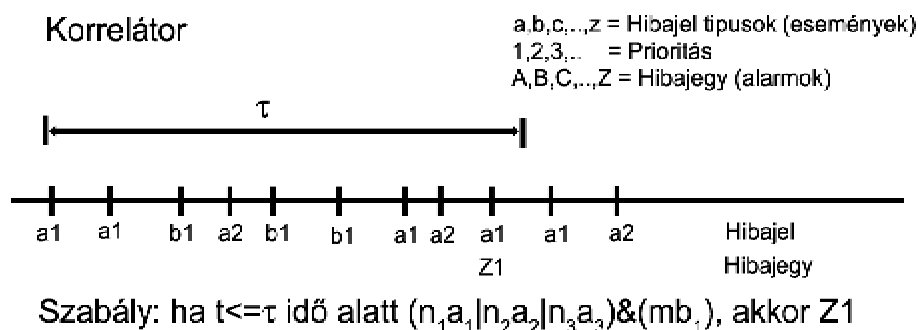
A legegyszerűbb hálózat-felügyeleti algoritmus a szűrés. A jelenlegi hibamenedzsment-rendszerekben használt szűrési algoritmusok közül [4] mi a szabály alapú szűrést [5] választottuk. Ennél a módszernél a szűrőmodul egyenként megvizsgálja a beérkező hibajeleket, hogy talál-e rájuk alkalmazható szabályt. Amennyiben ilyen szabály nincs, a hibajelből alapértelmezés szerint hibajegyét generál. Minden más esetben az alkalmazott szűrőszabálynak megfelelően jut tovább, illetve nyomódik el a hibajel. A szabályok a leírásukban szereplő alap-paraméterek (érvényességi időintervallum, hibajel kód, darabszám) értékének és az adott szabály típusának (elnyomó, számláló, redundancia gátló, dominancia) megfelelően fejtik ki hatásukat [1].

Az operátori munka megkönnyítéséhez a szabály-rendszernek legalább a következő szűrési igényeket kell kielégíteniük:

- a folyamatosan, egymás után beérkező nem súlyos alarmok esetében ne generálódjon mindig egy újabb hibajegy (számláló és elnyomó típusú szűrő),
- ha egy hibajegy keletkezett a rendszerben, akkor addig ne engedjünk több hibajegyét generálódni, amíg tudjuk, hogy a hibát javítják (redundancia gátló),
- a súlyosabb (nagyobb prioritású) alarmok nyomják el a felesleges gyengébb (kisebb prioritású) alarmokat, amelyek ugyanarra a jelenségre, hibára utalnak (dominancia szűrő).

Korrelátor

Az eseménykorreláció a szűréshez képest egy magasabb szintű feldolgozási módszer, amely különböző hibajelek közti összefüggések felderítésére hivatott. Több hibajel rövid időn belüli beérkezése esetén módunk van egy sokkal specifikusabb hibajelet megadni. A korrelátor alkalmazása során nem veszítünk hibajelet: a szabály kielégülése esetén mind a bementi, mind az új, specifikus hibajel továbbküldésre kerül (a szűrő modulba, amely megfelelő módon nyomja majd el ezeket). Az esemény-korrelátor végső célja, hogy lehetőség szerint átfogóbb hibajelet generáljon, számos hibaok-elemet vonjon össze egy eredendő okká. A korrelációs szabályok alkalmazásának hatását a 3. ábra szemlélteti [6].



3. ábra Szabály alapú korrelátor

Trend-analizátor

A trend-analizátor alkalmazásával elérhetjük, hogy adott hibajelek előfordulásaiából előre jelezni tudjuk a rendszer rendellenes viselkedését. A trend-analízis rejt magában némi bizonytalanságot, hiszen a trend meghatározására nincs bizonyítottan jó automatikus módszer.

Az aktuális folyamatra kell találni egy jó trendfüggvényt, amivel lehetőségünk nyílik jósolni. Jelen esetben valamelyest más a helyzet, itt az előre megírt szabályainknak eleget tevő hibamintákat keres az algoritmus. A hibabekövetkezés előre jelzésénél a szabály által meghatározott trendváltozót vizsgáljuk, pl. ennek kell egy bizonyos küszöbértéknél nagyobbak lennie. A hibáknak egy adott időn belül kell bekövetkeznie, sorrendiségük mellékes tényező. Tapasztatok szerint a trend-analízis segítségével alapvetően telítődéses (a hiba folyamatos romlás eredményeképpen, fokozatosan alakul ki) hibákat tudunk jósolni. A hirtelen bekövetkező hibák esetében a módszer használhatatlan, nincs miből jósolni.

Hibaok-meghatározás és javítás

Egy megfelelően összetett szabályrendszer alkalmazásával elérhető, hogy a hibafeldolgozás kimenete ne csak a specifikus hibajegy legyen, hanem paraméterként tartalmazza a hiba helyét, és a valószínűsíthető okot. Ehhez összetett, és egyben paraméterezett korrelációs szabályokra van szükségünk. Az ilyen hibajegyekből származó paramétereket a hibaok-meghatározó rendszer kimeneteinek is tekinthetjük.

Ez a módszer azonban kizárólag passzív hibakorreláció esetében használható hibaok-meghatározásra. A passzív szabályrendszer legnagyobb hátránya, hogy nem ad lehetőséget a hálózati csomópontok aktív lekérdezésére - így (kulcsfontosságú információk hiányában) nem tudjuk az összes hibajegy RCA elemzését automatikusan elvégezni. Az ilyen esetek megoldásához algoritmikus hibaok-meghatározásra van szükség.

Algoritmikus hibaok-meghatározás

A hibaok-analízis (Root Cause Analysis, RCA) során a hálózathoz kinyerhető információk (topológiai, eszköz adatok) felhasználásával határozzuk meg a legvalószínűbb hibát. A folyamat eredménye a hibajavítási javaslat, melynek figyelembe vételével nagyban leegyszerűsíthető a hálózatot üzemeltető operátor munkája.

A VoIP-szolgáltatások felügyeletére kialakított RCA-keretrendszerünkkel szemben az alábbi követelményeket támasztottuk:

- a különböző esemény-adatbázisokban való keresés,
- aktív ellenőrzések beiktatása,
- az ellenőrzések és keresésének párhuzamos futtatásának kezelése,
- a topológiai információ változásának dinamikus követése,
- moduláris fejleszhetőség,
- a fejlesztés megtérülésének gyorsasága.

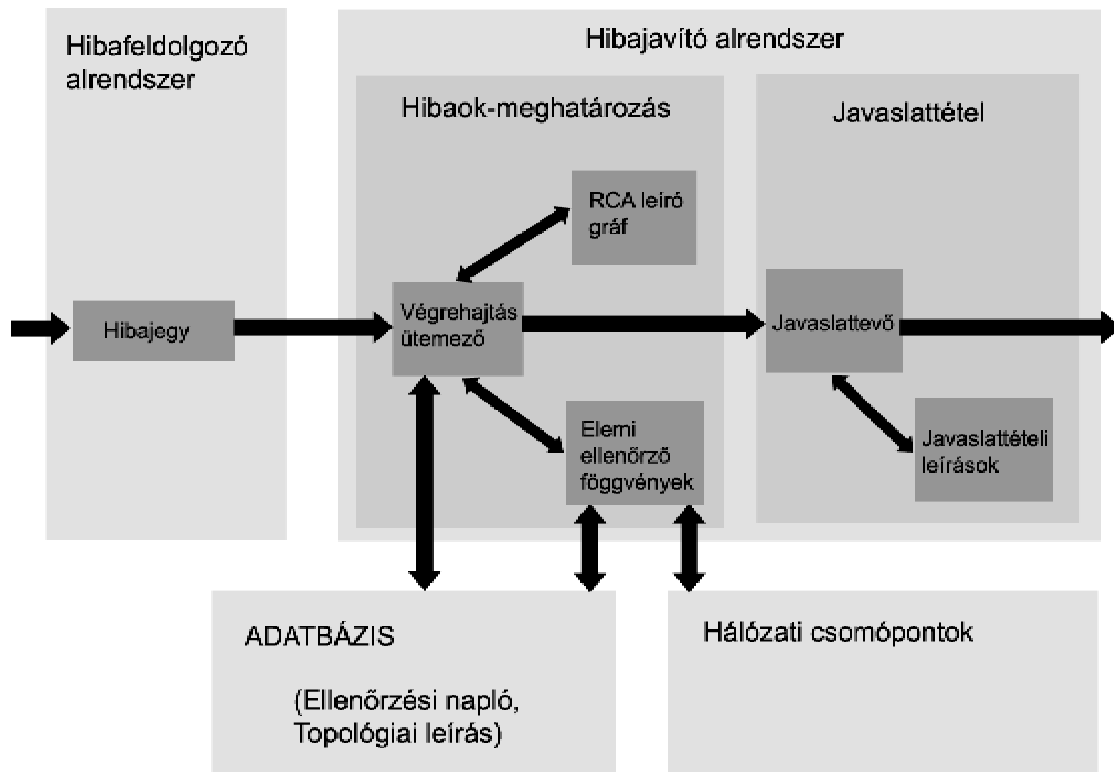
A hibaok-meghatározási módszereket három nagy csoportba rendezhetjük: *hibakorrelációs módszereken alapuló, statisztikai alapon működő* illetve *modell alapú* rendszerek. Az aktív lekérdezések indítását, illetve ezek eredményeinek kiértékelését is támogató módszerként egy modell alapú RCA módszert fejlesztettünk ki.

A modell alapú RCA rendszerek jellemzően egy rugalmas modell segítségével írják le a hálózati topológiát. Új elem bekerülése esetén a korrelációs szabályrendszer generálható, és nem kell manuálisan újraírni. A topológiai leírás (VoIP hálózatot feltételezve) tartalmazza a fizikai csomópontokhoz rendelt saját IP-címet és az adott csomópontokhoz egy hop-on belül csatlakozó csomópontok IP-címeit. A szabályrendszer modelljének kialakításánál a topológiai információt rugalmasan kell kezelni (fel kell készíteni a topológia változására). A modell a hibajelek egymáshoz való viszonyát hierarchikusan írja le.

Petri hálózattal vezérelt RCA folyamat

A hibajavítás folyamata a 4. ábra követhető végig. A hibajegyek a hibafeldolgozó alrendszerből érkeznek a hibajavító alrendszer FIFO-jába. Minden hibajegy-típushoz tartozik

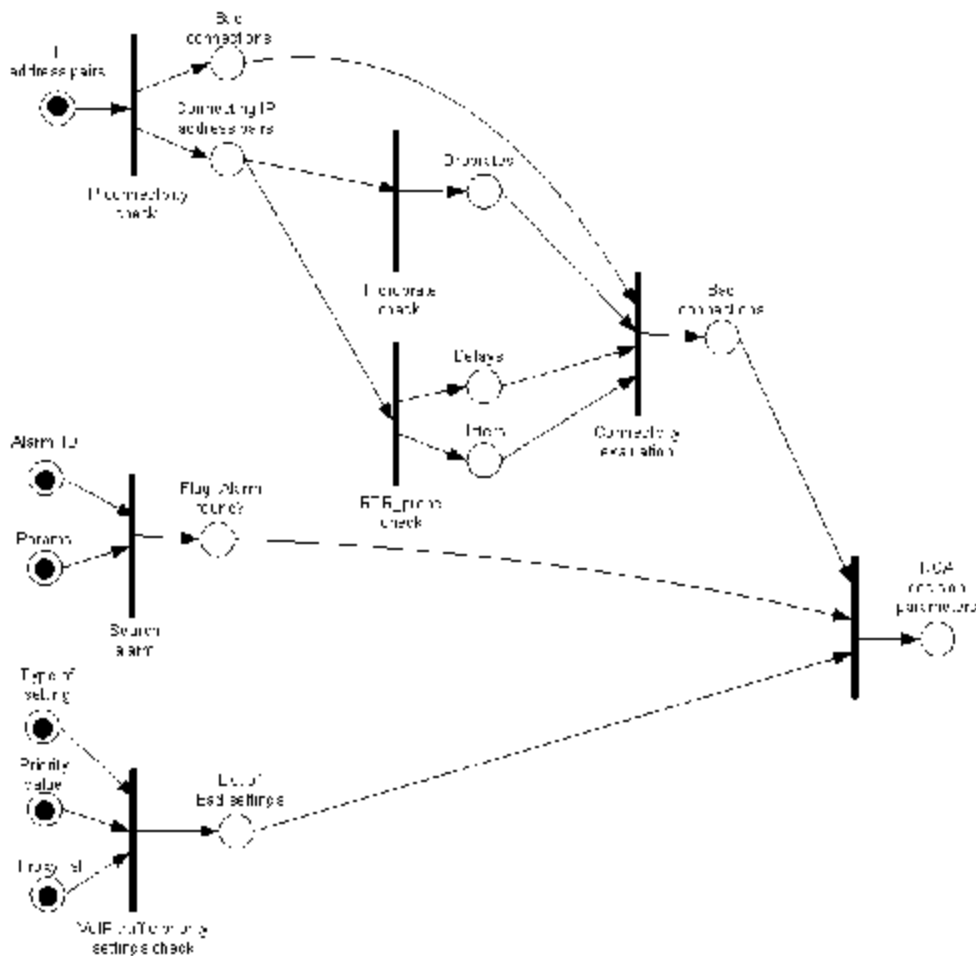
egy Petri-hálózat formájú *RCA-leíró gráf*. Ez grafikusán ábrázolja az hibaok kiértékeléshez szükséges lépések (*elemi függvények*) kapcsolatát: mely adatok rendelkezésre állása esetén mely lépéseket lehet végrehajtani. Az aktuális ütemben végrehajtható lépések (elemi függvények) indítását a *végrehajtás ütemező* végzi. Amint minden eredmény visszaért a végrehajtás ütemezőhöz, és nincs több végrehajtható lépés, a javaslattevő kapja meg a hibaok-meghatározási folyamat eredményét. Ez a végeredményhez rendeli a megfelelő javaslattevői leírást, és az operátor rendelkezésére bocsátja azt.



4. ábra Hibajavító alrendszer architektúrási felépítése

A fenti folyamat központi eleme az *RCA-leíró gráf*. Ezt a hibamenedzsmentben egyedi, újszerű módszerrel, egy Petri-hálózatos leírással valósítottuk meg [7]. A megoldás előnye, hogy az aktív ellenőrzési feladatok konkurens módon, az adatok rendelkezési állásának ütemében hajtódnak végre. A gráfba külön adatlekérdező lépéseket lehet bevezetni annak érdekében, hogy egy adott ellenőrző-függvény összes bemeneti változója rendelkezésre álljon.

Példaképpen tekintsük végig az 5. ábra látható, „magas hálózati csomagvesztés” hibajegyhez tartozó *RCA-leíró Petri-hálózat* alkalmazásának jellegzetességeit. Abban a pillanatban, amikor a hibajavító alrendszerhez megérkezik a hibajegy, a végrehajtás ütemező aktiválja a jegyhez tartozó *RCA-gráfot*. Az első ütemben mindazon „átmenetek” (jelen esetben konkrét elemi ellenőrző függvények) „tüzelnek” (végrehajtódnak), amelyek minden bemenetén van a kiindulási állapotban „token”.



5. ábra “Magas hálózati csomagvesztés” RCA gráfjának Petri hálózatos leírása, kiinduló állapot

Ezek az elemi ellenőrző függvények a VoIP szolgáltatásokhoz kapcsolható hibamenedzsment feladatoknak megfelelően a következő csoportokba sorolhatók:

- interfész-állapotot lekérdező függvények (pl. check_IP_interface, check_analogue_if, check_ISDN_interface, stb.),
- konfigurációval kapcsolatos lekérdezés (check_trunk_setting, check_dialpeer_setting, VoIP_source_priority_settings_check, radius_accounting_check, stb.),
- aktív lekérdezések (IP_connectivity_check, IP_droprate_check, IP_delay_check, RTR-probe_check, stb.),
- egyéb (search_alarm, test_call, stb.).

Amikor egy adott elemi függvény kiértékelésre kerül, a „token” megjelenik a kimenetén (az ehhez tartozó adat más elemi függvények bemenete lehet). A végrehajtás ütemező minden ütemben megvizsgálja, hogy van-e az RCA-gráf átmenetei (elemi függvényei) között olyan, amelynek minden bemenetén van token, s ha talál ilyet, elindítja az adott elemi függvény kiértékelését. Ha az utolsó (tipikusan a végső kiértékelést végző) elemi függvény kimenetén is megjelenik a token, a végrehajtás ütemező átadja az eredményeket a javaslattevőnek, és megszünteti az RCA-entitást.

A Petri-hálós RCA-leírásnak köszönhetően egyes RCA-folyamatok végrehajtási ideje akár felére is csökkenthető [6], ráadásul az ellenőrzések végrehajtásához nem kell előre meghatározni a végrehajtási sorrendet, csak azt, hogy az adott RCA kiértékeléséhez milyen feladatokat kell végrehajtani. A feladatok a Petri háló összeköttetéseit követve az adatok rendelkezésre állásának ütemében hajtódnak majd végre.

Összefoglalás

Megfelelő szűrőszabályokkal megakadályozhatjuk, hogy a VoIP hibamenedzsment-rendszert feleslegesen árasssák el (azonos típusú) hibajegyek. A beérkező hibajegyeket időlegesen teljesen elnyomhatjuk, számukat korlátozhatjuk, prioritásokat határozhatunk meg közöttük. További műveleteket tudunk végezni, ha a hibajegyeket egy adatbázisban tároljuk. Korrelátor-szabályok segítségével összefüggéseket lehet felállítani a hibajelenségek között, több „kisebb” hibát egy összefoglaló hibajegy alá tudunk vonni. Ha csak ezt tárjuk az operátor elé, akkor nagyban megkönnyítjük a munkáját. Megfelelő „tudás” birtokában trendanalízis segítségével képesek vagyunk bizonyos hibákat előre is jelezni. A művelet során mintaillesztéssel kutatunk a hiba-adatbázisban – sikeres találat esetén komolyabb hibák előrejelzésére is lehetőségünk nyílik.

A szűrő-, korrelátor- és trendanalízis-szabályok alkalmazásával az eredendő hibaok nem minden esetben mutatkozik meg. A hibamenedzsment rendszernek ilyenkor el kell indítania egy hibaok-analizáló folyamatot (RCA, Root Cause Analysis). Az RCA során a hibamenedzsment rendszer aktív lekérdezésekkel ellenőrizheti az esetleges hibaforrásokat. Az eredmények kiértékelése után javaslatot tesz a hibaokra és annak helyére a hálózatban, majd a hiba javítására. Az ellenőrzések sorrendezésére és kiértékelésére több módszer létezik. Az IKTA-00092-2002 számú OM-projekt keretében a NIIFI szakembereinek bevonásával kifejlesztettünk egy új, Petri-hálós leíráson alapuló módszert. Ennek alkalmazásával az RCA ellenőrző lépései az adatok rendelkezésre állásának függvényében (konkurens módon) kerülnek végrehajtásra, a folyamat leírása leegyszerűsödik, a végrehajtás felgyorsul.

Irodalomjegyzék

- [1] Felügyeleti rendszer integrált hang-adat hálózatok számára (műszaki melléklet, I. és II. munkaszakaszok) – *BME-TMIT, Ericsson Magyarország Kft., Kovax'95 Kft., IKTA-00092-2002 beszámoló, 2003.*
- [2] ETSI TS 101 329-5 V1.1.2; Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 3; End-to-end Quality of Service in TIPHON systems; Part 5: Quality of Service (QoS) measurement methodologies – *ETSI, 2000.*
- [3] ITU-T Recommendation P.862; Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs – *ITU-T, 2001.*
- [4] A Model For Alarm Correlation in Telecommunications Networks – *Dilmar Malheiros Meira, 1997.*
- [5] Discovering Rules for Fault Management – *Roy Sternitt, Eighth Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS '01), 2001.*
- [6] Hibamenedzsment VoIP hálózatokban – *Szijártó Tamás, BME-VIK TDK konf. 2004.*
- [7] Petri Net Theory and the Modeling of Systems – *James Lyle Peterson, Prentice-Hall, 1981.*