



IBM Systems - iSeries
Gestionarea sistemelor
Control funcționare

Versiunea 5 Ediția 4





IBM Systems - iSeries
Gestionarea sistemelor
Control funcționare

Versiunea 5 Ediția 4

Notă

Înainte de a folosi aceste informații și produsul pe care îl suportă, citiți informațiile din “Observații”, la pagina 187.

Ediția a cincea (Februarie 2006)

Această ediție se aplică versiunii 5, ediția 4, modificarea 0 a IBM i5/OS (număr produs 5722-SS1) și tuturor edițiilor și modificărilor ulterioare, până când se indică altceva în edițiile noi. Această versiune nu rulează pe toate modelele RISC și nici pe modelele CISC.

© Copyright International Business Machines Corporation 2004, 2006. Toate drepturile rezervate.

Cuprins

Control funcționare	1	Monitorizare activitate sistem	89
Ce este nou în versiunea V5R4	1	Gestionarea joburilor	96
PDF care poate fi tipărit	2	Gestionarea planificării de job	116
Introducere în Control funcționare	3	Gestionarea subsistemelor	142
Sistemul dumneavoastră ca o afacere	3	Gestionarea pool-urilor de memorie	158
Durata unui job	4	Gestionarea cozilor de joburi	161
Cum se termină lucrul	6	Gestionarea cozilor de ieșire	167
Concepte	8	Gestionarea istoricelor de job	169
Structura sistemului dumneavoastră	8	Gestionarea contabilizării de job	175
Subsistemele	11	Referință	180
Pool-urile de memorie	20	Depanare	181
Joburile	24	Jobul meu este în impas	181
Opțiunile de planificare job	50	Jobul meu are o performanță scăzută	182
Cozile de joburi	54	Investigarea jobului prestart	183
Cozile de ieșire	58	Informații înrudite pentru gestiunea lucrului	184
Istoricul joburilor	66	Anexa. Observații	187
Contabilizarea jobului	73	Mărci comerciale	188
Control funcționare	87	Termenii și condițiile	189
Modificarea programului de pornire IPL	87		
Apelarea unui program de recuperare IPL special	89		

Control funcționare

Control funcționare este o parte importantă în sistemul de operare al serverului iSeries.

Funcțiile sale reprezintă baza prin care toată munca intră în sistem, este procesată, rulată și efectuată pe serverele iSeries. Chiar dacă rulați un simplu job batch o dată pe săptămână sau apelați o aplicație zilnic (precum Lotus Notes), Control funcționare vă ajută să gestionați joburile și obiectele care rulează pe sistemul dumneavoastră. Suportă și comenzile și funcțiile interne necesare pentru a controla operațiile de sistem și pentru a aloca resurse aplicațiilor, când este necesar.

Serverul iSeries este setat și gata de a fi utilizat. Pentru majoritatea utilizatorilor, nu este necesară modificarea setărilor implicite. Totuși, dacă aveți nevoie să ajustați Control funcționare conform nevoilor companiei dumneavoastră, veți avea nevoie să înțelegeți termenii și conceptele asociate și modul cum acestea se integrează una cu alta pentru a vă furniza cea mai bună performanță a serverului dumneavoastră iSeries.

Chiar dacă sunteți un utilizator iSeries experimentat sau abia acum învățați, această colecție de subiecte vă oferă prezentare ușor de înțeles pentru Control funcționare. Acest subiect conține puncte de intrare diferite, astfel încât puteți să alegeți de unde porniți să învățați despre Control funcționare.

Notă: În plus, puteți lucra cu Control funcționare utilizând task-uri Navigator iSeries pe Web. Aceasta vă permite să lucrați cu funcții din Control funcționare folosind un browser de Web. Pentru informații suplimentare, vedeți task-urile Navigator iSeries pe Web.

Ce este nou în versiunea V5R4

Această ediție aduce unele modificări interesante, nu doar funcțiilor de control de funcționare dar și subiectului Control funcționare din Centrul de informare.

Acum aveți mai mult control asupra producerii istoricului de job.

Starea istoric job în așteptare a fost disponibilă de mulți ani încoace. Când un istoric de job este în stare de așteptare, nu este produs nici un istoric de job. O dată cu ediția V5R4, fereastra Istoric job a Proprietăților jobului din Navigator iSeries a fost îmbunătățită astfel încât acum să puteți controla cum și în ce circumstanțe va fi produs istoricul de job pentru un anumit job.

Dacă preferați interfața bazată pe caractere, comanda WRKJOBLOG (Work with Job Logs) este o comandă nouă pe care o puteți utiliza pentru a gestiona istorice de joburi. Când această comandă este rulată, este afișat ecranul Gestionare istorice de joburi. Din acest ecran puteți afișa istorice de joburi, șterge istorice de joburi, puteți lucra cu fișiere puse în spool și cu jobul care este asociat cu istoricul de job.

În descrierea de job, noul atribut **LOGOUTPUT** specifică dacă istoricul de job este produs de către serverul de istorice de job, de către jobul însuși sau dacă nu este produs deloc. Dacă descrierea de job specifică *SYSVAL pentru valoarea **LOGOUTPUT**, atunci este utilizată noua valoare de sistem QLOGOUTPUT (Job log output - ieșire istoric de job) pentru a controla modul cum este produs istoricul de job.

Acum sunt disponibile informații suplimentare pentru stive de apeluri.

Începând cu V5R4, stiva de apeluri poate fi modificată atât prin Navigatorul iSeries cât și prin interfața bazată pe caractere, pentru a adăuga ajutor pentru aplicațiile i5/OS PASE, Java și LIC. Multe câmpuri care existau au fost redenumite, reorganizate sau supraîncărcate, pentru a afișa o varietate mai mare de date.

Comenzile CHGJOB (Change Job - Modificare job) și ENDJOB (End Job - Oprire job) pot opera acum pe orice job curent sau terminat.

În edițiile anterioare, comenzile nu funcționau pentru joburi speciale ca joburile de sistem, joburile monitor subsistem, cititorii de spool și scriitorii de spool. Această modificare unifică modul în care sistemul tratează fișierele de ieșire (puse în spool).

Ecraanul WRKACTJOB (Work with Active Jobs - Gestionare joburi active) a fost îmbunătățit.

Acum puteți vizualiza **Profilul utilizator curent** în ecranul Gestionare joburi active. Acest atribut este deja disponibil în lista de Joburi active a Navigatorului iSeries.



Au fost aduse mai multe îmbunătățiri la Advanced Job Scheduler.

- Work Flow Manager este o unealtă nouă care vă permite să definiți unități de lucru care pot conține o combinație de pași manuali și automați. Unitățile de lucru pot fi atunci planificate sau pornite manual.
- Puteți adăuga mai multe comenzi la un task planificat.
- Aveți mai mult control asupra banner-ului de notificare.
- Puteți utiliza Advanced Job Scheduler pentru a trimite un e-mail.
- Puteți distribui rapoarte prin containerul Operații de bază.

Întregul subiect Control funcționare a fost reorganizat și actualizat pentru Centrul de informare IBM iSeries Versiunea 5 ediția 4 (V5R4).

Noul subiect Control funcționare înlocuiește manualul Control funcționare V4R5 care era disponibil în Centrul de informare IBM iSeries Versiunea 5 ediția 3 (V5R3). Acest manual furniza informații despre cum să gestionați eficient volumul de lucru din sistem modificând obiectele de control funcționare pentru a îndeplini necesitățile dumneavoastră. El mai furniza, de asemenea, indicații pentru reglarea performanței, descrierea valorilor de sistem, informații despre colectarea datelor de performanță, adunarea datelor de utilizare a sistemului, utilizarea intrărilor de lucru și planificarea joburilor batch.

Cum să vedeți ce este nou sau modificat

-  Imagine pentru a marca unde încep informațiile noi sau modificate.
-  Imagine pentru a marca unde se termină informațiile noi sau modificate.

PDF care poate fi tipărit

Utilizați acesta pentru a vizualiza și tipări un PDF care are această informație.

Pentru a vedea sau descărca versiunea PDF a acestui document, selectați Control funcționare (aproximativ 2000 KB).

Puteți vizualiza sau descărca aceste subiecte înrudite:

- Reglarea performanței (1700 KB) conține următoarele subiecte:
 - Planul pentru performanță
 - Gestionarea performanței iSeries
 - Aplicații pentru performanță
- Administrarea centrală (810 KB) conține următoarele subiecte:
 - Gestionarea utilizatorilor și grupurilor
 - Colectarea datelor de inventar și de performanță
 - Gestionarea valorilor de sistem
 - Sincronizarea funcțiilor
 - Împachetarea și trimiterea obiectelor
 - Rulare comenzi
 - Planificarea operațiilor și joburilor dumneavoastră
 - Securitatea și planificare de bază pentru sistem

Salvarea fișierelor PDF

Pentru a salva un PDF pe stația dumneavoastră de lucru pentru vizualizare sau tipărire:

1. Faceți clic dreapta pe PDF din browser-ul dumneavoastră (apăsați clic dreapta pe legătura de mai sus).

2. Faceți clic pe opțiunea care salvează PDF local.
3. Navigați către directorul în care doriți să salvați PDF-ul.
4. Faceți clic pe **Save**.

Descărcarea Adobe Reader

1. Trebuie să aveți instalat Adobe Reader pe sistemul dumneavoastră pentru a vedea sau tipări aceste PDF-uri. Puteți
2. descărca o copie gratuită de pe situl Web Adobe (www.adobe.com/products/acrobat/readstep.html) .

Introducere în Control funcționare

Control funcționare suportă comenzile și funcțiile interne necesare pentru controlul funcționării sistemului și volumul zilnic de lucru din sistem. În plus, controlul de funcționare conține funcțiile de care aveți nevoie pentru a distribui resursele pentru aplicații astfel încât sistemul dumneavoastră să poată trata aplicațiile.

Scopul sistemului dumneavoastră este să realizeze lucrul. Lucrul intră, este procesat și părăsește sistemul. Dacă vă gândiți la controlul de funcționare în acești trei termeni el va fi mai ușor de înțeles. Control funcționare descrie unde lucrul intră în sistem, unde și cu ce resurse lucrul va fi procesat, și unde merge ieșirea lucrului.

Sunteți novice în Control funcționare? Colecția de teme de sub subiectul Introducere în controlul de funcționare este proiectată pentru a vă furniza mai multe perspective generale diferite ale controlului de funcționare. În acest mod, ar trebui să puteți obține fundamente solide în ceea ce privește principiile fundamentale ale controlului de funcționare, indiferent de experiența dumneavoastră anterioară în sisteme.

Sistemul dumneavoastră ca o afacere

Pentru a face lacom conceputul general de Control funcționare mai ușor, încercați să comparați sistemul dumneavoastră cu o afacere.

Un sistem simplu poate fi comparat cu o afacere mică și un sistem complex poate fi comparat cu un mall de cumpărături. Presupuneți că există un mic magazin care se ocupă cu afaceri de construire a mobilierului de lemn lucrat manual. *Intră muncă*, precum și comenzi pentru mese mici, scaune și rafturi pentru cărți. *Munca este procesată*, tâmplarul sună clienții pentru a le confirma comanda și aceștia sunt consultați asupra punctelor din cadrul proiectării inclusiv stil, dimensiune și culoare. Tâmplarul proiectează fiecare piesă de mobilier, adună materialele necesare și apoi construiește mobila. După ce mobilierul este terminat, el este livrat: *munca părăsește magazinul*.

Din moment ce un sistem complex reprezintă o combinație de multe sisteme simple, un exemplu comparabil cu un sistem complex este un mall de cumpărături, multe afaceri mici și mai mari într-o singură zonă. Poate că tâmplarul are o afacere în colțul de nord-vest al mall-ului și un brutar are o afacere în zona de Est. Brutarul și tâmplarul au intrări și ieșiri diferite, aceasta înseamnă că comenzile și produsele lor sunt foarte diferite. În plus, durata de procesare a muncii pentru fiecare afacere este diferită, și utilizatorii lor cunosc și înțeleg acest lucru.

Termeni de Control funcționare

Un sistem complex (mall de cumpărături) este alcătuit din mai multe sisteme simple (magazine). Aceste sisteme simple sunt numite *subsisteme*.

Orice lucru din cadrul afacerii este considerată a fi un *job*. Un exemplu de lucru ar putea fi scrisoarea unui client, un apel telefonic, o comandă sau curățenia de noapte. Același lucru poate fi spus și despre sistemul IBM iSeries. În sistem, fiecare job are un nume unic.

O *descriere de job* descrie cum să fie tratat lucrul care sosește în subsistem. Descrierile de job conțin piese de informație precum ID-uri utilizatori, cozi de joburi și date de rutare. Informațiile din descrierea unui job se poate compara cu descrierile de joburi dintr-o afacere mică.

Cum arată afacerea? Fiecare magazin are schițe sau planuri ale magazinului. Aceste planuri sunt de fapt tocmai descrierile, în detalii variate, ale realizării fizice a afacerii. Poate afacerea are un magazin cu: 2 etaje, 5 uși, 3 cutii poștale și 2 telefoane. În sistemul iSeries, o *descriere de subsistem* conține toate informațiile despre subsistem.

De unde sosește munca? Pentru tâmplar, munca sosește din apelurile telefonice ale clienților, din referințe, și de la persoanele care intră în magazin. În sistemul iSeries, munca poate veni din multe locuri. Exemplele includ cozi de joburi, stații de lucru, comunicații, joburi autostart și joburi prestart.

Unde găsec aceste spații? În mall, fiecare afacere (subsistem) are o anumită suprafață repartizată. În sistemul iSeries, *pool-urile de memorie* vă permit să controlați spațiul de stocare principal (sau suprafața mall-ului) pe care fiecare subsistem (afacere) îl primește pentru a putea să își realizeze munca. Cu cât un magazin (subsistem) are o suprafață mai mare, cu atât mai mulți clienți sau joburi pot încăpea în magazin.

Cum sosește munca? Clienții care nu pot găsi magazinul de care au nevoie, pot găsi un chioșc de informații care să îi trimită în direcția corectă. Același lucru este adevărat și în sistemul iSeries. *Intrările de rutare* sunt similare cu îndrumătoarele de magazin sau cu un chioșc de informații. După ce este găsită intrarea de rutare, ea ghidează jobul către locul lui corect. Intrarea de rutare trebuie, totuși, întâi să fie găsită. Aceasta se realizează prin *datele de rutare*. Datele de rutare sunt ceea ce utilizează jobul pentru a găsi intrarea de rutare corectă.

Cum este tratată munca? Un tâmplar are nevoie să fixeze o prioritate pentru fiecare job. Comanda de scaune pentru sfârșitul săptămânii ar trebui terminată înainte de cea a rafturilor de cărți de la sfârșitul lunii. În sistemul iSeries, *clasele* furnizează informații despre cum este tratat un job cât timp se află în subsistem. Aceste informații includ prioritatea în timpul rulării, spațiul de stocare maxim, timpul maxim CPU și felia de timp. Fiecare din aceste atribute influențează modul și momentul procesării un job.

La fel cum există reguli care afectează toate magazinele dintr-un mall, există reguli care afectează toate subsistemele din sistemul iSeries. Un exemplu de astfel de regulă este o *valoare de sistem*. Valorile de sistem sunt piese de informație care se aplică întregului sistem. Valorile de sistem includ informații precum data și ora, informații de configurare, informații de semnare, securitate a sistemului și manipularea spațiului de stocare.

Clienții dintr-un mall au toți informații specifice lor. În sistemul iSeries, *profilul utilizator* reține informații specifice unui anumit utilizator. Similar cu cartea de credit a unui client, un profil utilizator oferă celui utilizator anumite autorități și alocă atributele de utilizator pentru joburile celui utilizator. Aceste *atribute de job* furnizează informații care includ, dar nu se limitează la, descrierea jobului, coada de ieșire sau dispozitivul de tipărire, coada de mesaje, codul de contabilizare și prioritatea de planificare.

Durata unui job

Pentru a înțelege principiile de bază ale controlului de funcționare a iSeries, urmăriți un job batch simplu pe măsură ce acesta își parcurge calea în interiorul sistemului.

Durata unui job simplu batch începe când îl lansați în sistem. Jobul este atunci trimis la o coadă de joburi unde așteaptă să intre în subsistem unde el poate rula. După ce jobul se mută în subsistem lui îi este alocată memorie în care să ruleze. Fișierul de ieșire imprimantă (numit și fișierul spool) sunt trimise apoi la coada de ieșire pentru a aștepta instrucțiuni ulterioare despre ce să facă (de exemplu, să tipărească). În timp ce nu fiecare job urmează această cale exact, puteți înțelege mai bine cum se efectuează cealaltă muncă din sistem aflând mai multe despre ciclul de viață al acestui job obișnuit.

Lansați jobul → Jobul intră în coada de joburi → Jobul intră în subsistem → Pool-ul de memorie alocă memorie subsistemului → Jobul se termină și este mutat în coada de ieșire

Viața unui job: lansarea unui job

Când un job este lansat, el este creat și intră în sistem. În acest moment, jobului îi sunt date atributele.

Descrierea de job are *atributele* pe care jobul le va utiliza pentru a parcurge ciclul de viață Control funcționare. Aceste atribute includ profilul utilizator sub care jobul va porni să ruleze, datele de cerere (care îi spun jobului ce va face) și porțiunea de utilizator inițială a listei de biblioteci și așa mai departe. Descrierea jobului păstrează de asemenea

informații ce spun jobului în ce coadă de joburi să intre și datele de rutare. Datele de rutare sunt mai târziu folosite de subsistem pentru a găsi intrările de rutare care conțin informațiile necesare pentru ca jocul să pornească rularea. Coada de ieșire este de asemenea definită în descrierea jobului. Ea spune unde va merge ieșirea imprimantă (numită și fișier spool) a jobului.

După ce jobul își primește valorile (inițializare, personalizare) pentru atributele sale de job, el se mută în coada de joburi unde așteaptă să intre în subsistem.

Viața unui job: jobul intră în coada de joburi

Cozile de joburi sunt puncte de intrare în sistem pentru joburile batch. Ele pot fi gândite ca și "camere de așteptare" pentru un subsistem.

Un număr de factori afectează momentul când jobul este scos din coada de joburi în subsistem, cum ar fi prioritatea în coada de joburi, numărul de secvență a cozii de joburi și numărul maxim de joburi active. Când toți acești factori lucrează împreună, jobul va fi scos din coada de joburi pentru a porni rularea în subsistem.

Când un job intră în coada de joburi, el este disponibil unui subsistem care are coada de joburi alocată la el. Deoarece subsistemul poate să aibă mai mult decât o coadă de joburi de alimentare în el (dar, coada de joburi nu poate alimenta mai mult de un subsistem), un număr de secvență în subsistem determina când subsistemul procesează o coadă de joburi. Subsistemul se uită la numărul de ordine al cozii de joburi înaintea priorității de job a joburilor din coada de joburi. Subsistemul utilizează prioritatea din coada de joburi pentru a determina când un job poate intra relativ la celelalte joburi din coada de joburi. Prioritatea jobului și joburile active maxime determină când un job intră în subsistem.

Viața unui job: jobul intră în subsistem

Subsistemele sunt medii de operare unde sistemul gestionează resursele pe care acele joburi le utilizează și controlează joburile care rulează în ele. După ce joburile rulează în subsistem, jobul subsistemului execută cereri de la utilizator precum reținerea, eliberarea și terminarea unui job. Când jobul intră în subsistem el devine activ.

Ca și joburile, subsistemele au descrierea care conține informații importante necesare ca să completeze lucru. În descrierea subsistemului este intrarea de rutare. *Intrarea de rutare* face referire la obiectul de clasă, care conține atributele care controlează mediul de runtime. Totuși, înainte ca jobul să poată obține intrarea de rutare, datele de rutare trebuie să se asocieze o valoare de comparare în intrarea de rutare. Dacă această asociere nu este făcută, jobul nu va rula.

După ce se realizează asocierea între datele de rutare și intrarea de rutare, este determinat obiectul de clasă pe care jobul îl va utiliza. Unele din atributele care controlează mediul de runtime includ prioritatea de rulare, felia de timp, timpul maxim de așteptare, timpul maxim de procesare, spațiul de stocare temporară maxim și numărul maxim de fire de execuție.

Descrierea subsistemului definește pool-urile de memorie care vor fi alocate subsistemului. Descrierea subsistemului de asemenea conține numărul maxim de joburi active, care este numărul maxim de joburi active în același timp în subsistem.

Un job nu poate rula până când nu obține nivelul său de activitate și îi este alocat un pool de memorie. Descrierea subsistemului, la fel ca descrierea jobului, conține informații, cum ar fi folosirea pool-ului de memorie, intrările de rutare, joburile active maxime, și numărul joburilor active curente în subsistem.

Viața unui job: subsistemul utilizează memoria din pool-ul de memorie pentru a rula jobul

Memoria este o resursă din pool-ul de memorie pe care subsistemul o utilizează pentru a rula jobul. Cantitatea de memorie dintr-un pool de memorie, precum și numărul de joburi care mai sunt în competiție pentru obținerea memoriei afectează eficiența rulării unui job.

Pool-urile de memorie furnizează joburilor memorie în care să ruleze. Mulți factori afectează modul în care jobul rulează în pool-ul de memorie, cum ar fi dimensiunea pool-ului de memorie, nivelul de activitate din pool-ul de

memorie și paginarea și lipsa paginilor. Nivelul de activitate din pool-urile de memorie se leagă direct cu numărul de fire de execuție care au permisiunea de a rula în pool-ul de memorie la un moment dat. Rețineți, fiecare job are cel puțin un fir de execuție activ, dar unele pot avea mai multe fire de execuție. Firele de execuție dau unui job abilitatea să facă mai multe lucruri în același timp. De exemplu, un fir de execuție poate să realizeze calculele în timp ce un alt fir de execuție să aștepte mai multe date de procesat.

Paginarea este mutarea datelor în și afară din memorie, ambele sincron și asincron. Paginile pot fi copiate în spațiul de stocare sau mutate din memorie fără să fie scrise dacă ele nu au fost modificate. Paginile lipsă fac să survină paginarea pe server. Paginile lipsă apar când o pagină la care se face referință sau o porțiune de date lipsesc din memorie. Acestea fac ca programele să se oprească deoarece ele trebuie să aștepte ca să fie adusă pagina cu datele necesare.

Subsistemele folosesc pool-uri de memorie diferite pentru a suporta diferite tipuri de joburi care rulează în ele.

Viața unui job: jobul se termină și se mută în coada de ieșire

Ieșirea de imprimantă a jobului (mai este numită și fișier pus în spool) este trimisă într-o coadă de ieșire unde așteaptă să fie trimisă la o imprimantă sau fișier. Coada de ieșire este similară cu coada de joburi prin faptul că ea controlează cum este pusă ieșirea la dispoziția imprimantei. Coada de ieșire permite utilizatorilor să controleze care fișiere sunt tipărite primele.

Cozile de ieșire sunt zone unde fișierele de ieșire imprimantă așteaptă să fie procesate și trimise la imprimantă. Ieșirea imprimantă este creată fie de către sistem fie de către utilizatorul care folosește un fișier de tipărit. Un fișier de imprimantă este similar cu un șablon sau cu o indicație unde sunt setate valorile implicite pentru atributele ieșirii de imprimantă. Este începutul ciclului de viață al ieșirii imprimantei.

Fișierul de imprimantă conține atributele pentru coada de ieșire (OUTQ) și dispozitivul de tipărire (DEV), care dictează modul în care ieșirea imprimantă va fi condusă. Setarea implicită este de obicei *JOB, ceea ce înseamnă că atributele job ale cozii de ieșire și ale dispozitivului de tipărire determină modul cum este direcționată ieșirea imprimantei. Atributele job ale setărilor cozii de ieșire și dispozitivului imprimantă sunt bazate pe informațiile obținute la crearea unui job. Aceasta este bazată pe informațiile din profilul utilizator sub care rulează jobul, descrierea jobului, descrierea dispozitivului stație de lucru și valoarea de sistem QPRTDEV (imprimantă implicită).

Când ieșirea imprimantă este gata să fie creată, sistemul verifică fișierul de imprimantă și atributele de job (în această ordine) pentru a vedea care coadă de ieșire va procesa ieșirea imprimantă și care dispozitiv de imprimantă va fi folosit de către sistem. Dacă o coadă de ieșire specificată nu poate fi găsită, ieșirea imprimantă va fi condusă la QGPL/QPRINT.

După ce fișierul ieșire imprimantă este gata de tipărire, un job scriitor, un job care procesează ieșirea imprimantei din coada de ieșire către dispozitivul de tipărire, ia datele de la fișierele de ieșire imprimantă și le trimite la imprimanta desemnată.

Cum se termină lucrul

Utilizați aceste informații pentru a învăța ce reprezintă lucrul, ce trebuie setat înainte ca lucrul să poată începe, cum călătorește lucrul prin sistem și ce se întâmplă cu lucrul după ce a terminat de rulat.

Ce este lucrul

Pe serverul iSeries, lucrul se realizează întotdeauna, chiar dacă îl inițiați dumneavoastră sau îl inițiază sistemul. Pentru efectuarea oricărei acțiuni pe serverul iSeries se realizează lucru.

Lucru se realizează atunci când porniți sistemul, când deschideți un fișier sau când interogați o bază de date. Fiecare bucată de lucru din sistem este executată de un job. Un job poate fi la fel de simplu ca o aplicație care așteaptă ca utilizatorul să o apeleze sau poate fi la fel de complex ca o interogare de sistem care rulează constant și care monitorizează numărul de utilizatori din sistem în fiecare oră. Unele joburi, în special joburile batch și joburile interactive, au asociate descrieri de job care spun când și unde va rula jobul.

Joburile sunt făcute din programe care execută anumite funcții. Nu există limită pentru cantitatea de funcții executate de un job. Un job conține instrucțiuni pas-cu-pas care trebuie completate pentru ca lucrul să fie făcut. Programele care

formează jobul, rulează într-o anumită ordine. (De exemplu, înainte trebuie să ruleze programul A pentru ca programul B să poată începe.) Firele de execuție ajută un job să își efectueze lucrul. Un job activ conține cel puțin un fir de execuție. Când un job conține mai multe fire de execuție, el are posibilitatea de a realiza mai multe lucruri în același timp. De exemplu, un fir de execuție poate să realizeze calculele în timp ce un alt fir de execuție să aștepte mai multe date de procesat.

Ce se întâmplă înainte ca lucru să intre în sistem

Toate joburile, cu excepția joburilor de sistem, rulează în subsisteme. Pentru ca lucrul să pornească într-un subsistem activ, trebuie să fie stabilite pool-urile de memorie și cel puțin un punct de intrare al sursei de lucru. Cozile jobului sunt un exemplu al surselor de lucru.

Serverul iSeries este livrat cu un set implicit de cozi de joburi, subsisteme și pool-uri de memorie, care pot permite ca lucrul să înceapă imediat ce sistemul este pus sub tensiune.

Puteți ajusta configurațiile subsistemului și ale pool-ului de memorie pentru a vă optimiza capacitățile și performanța serverelor iSeries. De exemplu, dacă un job batch este critic pentru succesul afacerii dumneavoastră, puteți dori să alocați mai multă memorie pentru rularea sa. Sau, puteți determina acel număr de joburi care rulează la un moment dat în subsistemul Qbatch ar fi cel mai mic deci acele joburi pot utiliza maximul de resurse pentru a rula. De asemenea, puteți crea cozi de joburi, subsisteme, și pool-uri de memorie proiectate special pentru a efectua diferite tipuri de lucru. De exemplu, puteți crea o coadă de joburi numită Nightreps, unde rapoartele batch din timpul nopții sunt trimise la un subsistem numit Nightrep care alocă memorie exclusiv pentru rularea acestor joburi batch.

Cum intră lucrul în sistem

Intrările lucrului identifică sursele unde joburile intră într-un subsistem pentru ca să devină disponibil să ruleze. Fiecare tip de job din iSeries are tipuri diferite de intrări de lucru pe care le utilizează.

De exemplu, cele mai multe joburi batch utilizează cozile de joburi pentru a intra în subsistem. Intrările în coadă de joburi sunt mecanismul prin care coada de joburi este definită ca o sursă pentru lucrări la un subsistem.

Intrările de lucru sunt păstrate în descrierea subsistemului. Dacă descrierea unui subsistem nu are o intrare de lucru pentru tipul de lucru făcut, jobul nu poate rula în acest subsistem. Subsistemele IBM au intrări de lucru implicite în descrierea subsistemului. Rețineți, câteva din intrările de lucru care sunt livrate cu subsistemele sunt deja alocate să ruleze anumite joburi.

Cum se procesează lucrul

Când serverul iSeries este pornit, un job de monitor de subsistem începe să ruleze. Jobul de monitor de subsistem controlează joburile în subsisteme. El de asemenea pornește și oprește funcționarea, precum și gestionează resursele pentru lucru în subsistem.

Lucrul (sau joburile) intră în subsistem prin intrări de lucru unde devine activ și eligibil pentru a rula. Lucrul poate fi terminat doar când subsistemul are memorie alocată pentru a rula. Memoria este alocată subsistemului de către pool-ul de memorie.

Cum ajută descrierea subsistemului la procesarea lucrului

Ca și un job, un subsistem are o descriere, numită descriere de subsistem. Descrierea subsistemului conține informații suplimentare care arată cum, unde, cât de multe acțiuni pot fi active într-un subsistem în același timp și care resurse pot fi folosite pentru a realiza activitatea.

Intrare de rutare

În descrierea subsistemului există o intrare de rutare care îi spune subsistemului ce program să ruleze pentru job, în ce pool de memorie să ruleze jobul și ce obiect de clasă să fie utilizat pentru rularea jobului.

Obiect de clasă

Obiectul de clasă definește prioritatea de rulare, timpul de așteptare implicit, felia de timp și alte atribute. Prioritatea de rulare este importantă deoarece ea determină când un job va obține timp de procesor pentru a

rula. Scala priorității de rulare începe de la 0 la 99, cu 0 prioritatea cea mai mare. (Doar joburilor de sistem le este dată prioritatea 0 deoarece acestea sunt joburile care rulează serverul iSeries.)

Când un job intră în subsistem, subsistemul încearcă să potrivească datele de rutare cu valoarea de comparație din intrarea de rutare. Dacă datele de rutare și valoarea de comparație dintr-o intrare de rutare sunt egale, intrarea de rutare este alocată jobului. Dacă nu există o potrivire în nici o intrare de rutare, jobul este oprit.

Alt factor care afectează momentul când un job rulează în subsistem este numărul de joburi care au permisiunea de a fi active în subsistem la un moment dat (cunoscut și ca maxim de joburi active din subsistem). Când este atins numărul maxim de joburi active într-un subsistem, nici mai pot intra joburi în subsistem până când joburile active existente termină rularea. Pentru ca un job să ruleze trebuie să fie alocată memorie subsistemului. Nivelurile de activitate ale pool-urilor de memorie îi spun serverului iSeries câte fire de execuție pot fi active într-un pool de memorie. Nu uitați, un job activ conține cel puțin un fir de execuție. Când este atins nivelul de activitate al pool-ului de memorie, jobul așteaptă ca un alt fir de execuție să îi cedeze folosirea sa a nivelului de activitate. Astfel, un job poate fi activ într-un subsistem și să nu ruleze.

Notă: Nu faceți confuzie între numărul maxim de joburi active în subsistem și nivelul de activitate al pool-ului de memorie.

Cum părăsește lucru sistemul

Coadă de ieșire funcționează similar cu o coadă de joburi care planifică ieșirea să fie tipărită. Ambele, ieșirea la imprimantă și coada de ieșire au atribute care sunt folosite pentru a tipări informații.

Ieșirea imprimantei reține datele de ieșire care așteaptă să fie procesate, cum ar fi informațiile care așteaptă să fie tipărite. Ieșirea imprimantei mai reține, de asemenea, informații importante care sunt utilizate pentru a planifica momentul când vor fi tipărite. Atributele ieșirii imprimantei includ coada de ieșire în care se va afla ieșirea imprimantei, prioritatea, starea și planificarea ieșirii imprimantei.

Coadă de ieșire conține atribute proprii care determină ordinea în care sunt procesate fișierele de ieșire imprimantă. Ea mai conține, de asemenea, autorizarea care este necesară pentru a face modificări asupra ieșirii imprimantei și asupra cozii de ieșire.

Când ieșirea la imprimantă este gata să fie trimisă la imprimantă este recepționată de un job autor. Jobul scriitorului ia datele din ieșirea imprimantei și le pregătește pentru tipărire.

Concepte

Chiar dacă sunteți nou în Control funcționare sau dacă ați utilizat unelte de control de funcționare mulți ani, veți găsi ceva util în secțiunea de Concepte a colecției cu subiectul Control funcționare.

Structura sistemului dumneavoastră

Doar ce ați primit iSeries. Este ne desfăcut și sunteți gata să mergeți - sau sunteți? Ce subsisteme sunt livrate cu sistemul? Există programe pornire pe care trebuie să le modificați? Cu ce fel de interfață de utilizator vă puteți aștepta să lucrați?

Subsistemele livrate cu sistemul

Două configurații de subsisteme complete sunt livrate de către IBM și pot fi folosite fără a fi modificate.

Configurarea pe care sistemul o folosește când este pornit este controlată de către valoarea de sistem QCTLSBSD (subsistem/bibliotecă de control). Configurarea implicită este compusă din următoarele descrieri de subsistem:

Qbase (subsistemul de control)	Qbase suportă joburi de comunicații, batch și interactive. El are un job autostart, care pornește automat subsistemele Qusrwrk, Qserver și Qspl.
Qserver	Acesta este subsistemul server de fișiere.
Qspl	Acesta este subsistemul spool care suportă joburi scriitor și cititor.
Qsyswrk	Acesta este subsistemul de lucru sistem. El conține joburi care suportă funcții sistem care sunt pornite automat la pornirea sistemului și când sistemul iese dintr-o stare restricționată.
Qusrwrk	Acesta este subsistemul de lucru utilizator. El conține joburi care sunt pornite de servere pentru a lucra în numele utilizatorului.

Cealaltă configurare, care este livrată de către IBM, este compusă din următoarele descrieri de subsistem:

Qctl (subsistemul de control)	Qctl are un job autostart, care pornește automat subsistemele Qinter, Qbatch, Qcmn, Qusrwrk, Qserver și Qspl.
Qinter	Acesta este subsistemul care suportă joburi interactive, cu excepția aceloră din consolă.
Qbatch	Acesta este subsistemul care suportă joburi batch.
Qcmn	Acesta este subsistemul care suportă joburi de comunicații, excluzând joburile de comunicații TCP/IP. Aceste joburi de comunicare sunt necesare pentru diferite protocoale de comunicare pe care sistemul i5/OS le suportă.
Qserver	Acesta este subsistemul server de fișiere.
Qspl	Acesta este subsistemul spool care suportă joburi scriitor și cititor.
Qsyswrk	Acesta este subsistemul de lucru sistem. El conține joburi care suportă funcții sistem care sunt pornite automat la pornirea sistemului și când sistemul iese dintr-o stare restricționată.
Qusrwrk	Acesta este subsistemul de lucru utilizator. El conține joburi care sunt pornite de servere pentru a lucra în numele utilizatorului.

Configurația Qbase dă posibilitatea să se ruleze toate funcțiile asemănătoare pe care le puteți rula cu configurația Qctl și este mai ușor de gestionat deoarece este compusă din mai puține subsisteme.

Configurația Qctl implicită vă permite un control individualizat mai puternic asupra operațiilor sistemului dumneavoastră prin divizarea activității sistemului în alte subsisteme pe baza tipului de activitate. De exemplu, dacă doriți să rulați joburi batch la sfârșitul săptămânii, dar nu doriți ca cineva să fie capabil să se semneze (cu excepția consolei), puteți să faceți ușor asta cu configurația Qctl prin oprirea subsistemului Qinter.

Dacă luați în considerare să creați propria dumneavoastră configurație de sistem, ați putea de asemenea găsi că mai ușor să folosiți configurația Qctl ca pe un punct de plecare decât să folosiți configurația Qbase.

Programe de pornire

QSTRUPPGM este un program de pornire. Acesta este o valoare de sistem care specifică numele programului apelat dintr-un job autostart când este pornit un subsistem de control. Acest program realizează funcții de setare, precum pornirea subsistemelor și a imprimantelor. Această valoare de sistem poate fi modificată doar de responsabilul cu securitatea sau de cineva cu autorizarea responsabilului cu securitatea. O modificare asupra acestei valori de sistem va intra în vigoare data următoare când este realizat un IPL.

QSTRUPPGM poate avea aceste valori:

- QSTRUP QSYS: Programul specificat este rulat ca rezultat al unui transfer de control către el de la jobul autostart din subsistemul de control.
- *NONE: Jobul autostart se termină normal fără a apela un program.

Ce se întâmplă în timpul unui IPL: Programul implicit de pornire QSYS/QSTRUP face următoarele:

- Pornește subsistemul QSPL pentru lucrul pus în spool.
- Eliberează cozile de joburi QS36MRT și QS36EVOKE dacă au fost reținute (acestea sunt utilizate de către mediul System/36).
- Pornește curățarea de Asistent operațional, dacă îi este permis.
- Pornește toți scriitorii de imprimantă doar dacă utilizatorul nu a specificat Nu în ecranul Opțiuni IPL.
- Pornește subsistemele QSERVER și QUSRWRK.
- Dacă subsistemul de control este QCTL, el pornește subsistemele QINTER, QBATCH și QCMN.

Tipuri de porniri

În timpul unui IPL (Initial program load-Încărcare inițială program), programele sistemului se încarcă din dispozitivul sursă de încărcare desemnat în spațiul de stocare auxiliar al sistemului. Este verificat și hardware-ul sistemului. Panoul de control al serverului iSeries afișează o serie de coduri de referință a sistemului care indică starea sa curentă și vă avertizează dacă există vreă problemă. Când se termină IPL-ul, interfața bazată pe caractere prezintă ecranul de semnare și utilizatorii pot să se semneze cu Navigatorul iSeries.

Există mai multe opțiuni pentru pornirea unui server iSeries. Puteți:

- Porni sistemul fără a face modificări de configurare. Acesta se numește *IPL nesupravegheat*.
- Modificați configurația sistemului dumneavoastră în timpul unui IPL. Acesta este numit *IPL supravegheat*.
IPL-urile supravegheate afișează ecrane suplimentare în funcție de opțiunile pe care le selectați în ecranul de opțiuni IPL. Acestea pot include ecrane care vă permit să modificați valorile de sistem și alte atribute de sistem în timpul IPL-ului, să reconstruiți căi de acces, să verificați starea restricțiilor de fișier fizic, să configurați și să denumiți noi dispozitive și să specificați opțiuni pentru mediul de operare.
- Modificați tipul de IPL din panoul de control al sistemului dumneavoastră.
- Planificarea opririi activității unui sistem și a repornirii.

Problemele generale din timpul unui IPL sunt denumite *IPL anormal*.

Pentru informații suplimentare despre IPL și oprirea sistemului, vedeți subiectul din Centrul de informare, Pornirea și oprirea iSeries.

Oprirea alimentării sistemului dumneavoastră

Oprirea alimentării sistemului dumneavoastră necesită deosebită atenție. Dacă opriți sistemul fără a termina anumite operații puteți cauza deteriorarea datelor sau comportarea imprevizibilă a sistemului.

Următoarele subiecte ale Centrului de informare conțin mai multe informații despre oprirea alimentării sistemului dumneavoastră în siguranță.

- Cum să opriți în siguranță iSeries-ul dumneavoastră când sunt prezente servere Windows integrate
- Oprirea alimentării unui sistem cu partiții logice
- Program de ieșire pentru oprirea alimentării sistemului
- Program de ieșire pentru ajustarea opririi alimentării

Navigators iSeries

Navigators iSeries este o interfață grafică puternică pentru clienții Windows. Cu Navigators iSeries, puteți gestiona și administra sistemele iSeries de la desktop-ul dumneavoastră Windows. Puteți folosi Navigatorul pentru realiza majoritatea operațiilor asociate cu gestionarea lucrului.

Această interfață a fost proiectată pentru a vă face mai productivi. De aceea, este recomandat să folosiți Navigatorul, care are ajutor online să vă ghideze. În timp ce această interfață este dezvoltată, s-ar putea să aveți nevoie să folosiți un emulator tradițional cum ar fi PC5250 pentru a face unele din operațiile dumneavoastră. Dacă un subiect discută o asemenea operație, veți fi sfătuit să folosiți interfața bazată pe caractere în pașii de instrucțiuni ai subiectului.

Subsistemele

Subsistemul este locul unde este procesat lucrul pe serverul iSeries. Un subsistem este un mediu de operare singular, predefinit prin care sistemul coordonează fluxul de muncă și utilizarea resurselor. Sistemul poate conține câteva subsisteme, toate operând independent față de celelalte. Subsistemele gestionează resursele.

Toate joburile, cu excepția joburilor de sistem, rulează în subsisteme. Fiecare subsistem poate rula operații unice. De exemplu, un subsistem poate fi setat să manipuleze doar joburi interactive, în timp ce alt subsistem manipulează doar joburi batch. Subsistemele pot fi proiectate și pentru a trata mai multe tipuri de lucrări. Sistemul vă permite să decideți numărul de subsisteme și ce tipuri de lucru va trata fiecare subsistem.

Caracteristicile de runtime ale unui subsistem sunt definite într-un obiect numit descrierea subsistemului. De exemplu, dacă doriți să modificați permanent cantitatea de muncă (numărul de joburi) care provine dintr-o coadă de joburi într-un subsistem, trebuie doar să modificați intrarea în coada de joburi din descrierea subsistemului.

Subsistemul de control

Subsistemul de control este subsistemul interactiv care pornește automat când pornește sistemul și este subsistemul prin care operatorul de sistem controlează sistemul prin consola sistemului. El este identificat în valoarea de sistem pentru subsistemul/biblioteca de control (QCTLSBSD).

IBM livrează două descrieri complete de subsistem de control: QBASE (subsistemul de control implicit) și QCTL. Doar un subsistem de control poate fi activ pe sistem la un moment dat.

Când sistemul este într-o condiție restricționată majoritatea activității din sistem este oprită și doar o stație de lucru este activă. Sistemul trebuie să fie în această condiție pentru ca anumite comenzi cum ar fi SAVSYS (Save System - Salvare sistem) sau RCLSTG (Reclaim Storage - Pretindere spațiu de stocare) să poată rula. Unele programe pentru probleme de echipament de diagnoză necesită, de asemenea, ca sistemul să fie într-o condiție restricționată. Pentru a opri această condiție, trebuie să porniți din nou subsistemul de control.

Notă: Mai există și o stare batch restricționată în care un job batch poate să fie activ.

Când toate subsistemele, inclusiv subsistemul de control sunt terminate, este creată o condiție restricționată. Puteți opri fiecare subsistem individual sau puteți utiliza ENDSBS SBS(*ALL) OPTION(*IMMED).

Important: Sistemul nu poate atinge starea restricționată până când nu există decât un singur job rămas în subsistemul de control. Câteodată este posibil să pară că a rămas un singur job, dar sistemul nu intră în starea restricționată. În acest caz trebuie să verificați că nu există joburi cerere sistem (Sysreq) suspendate, joburi de grup suspendate sau joburi deconectate în ecranul rămas activ. Utilizați comanda WRKACTJOB (Work with Active Jobs - Gestionare joburi active) și apăsați F14=Include pentru a afișa orice job suspendat sau deconectat. Dacă astfel de joburi există, trebuie să le opriți pentru ca sistemul să intre în starea restricționată. Funcțiile ENDSYS și ENDSBS vor trimite un mesaj de informare CPI091C către emitentul comenzii atunci când este detectată această condiție.

De ce să fie luate în considerare subsistemele multiple

Pe măsură ce numărul de utilizatori din sistem crește, un singur subsistem este de cele mai multe ori insuficient pentru un set de lucru. Împărțind utilizatorii dumneavoastră în subsisteme multiple câștigați mai multe avantaje.

O administrare îmbunătățită a lucrului

Obțineți un control mai bun asupra lucrului care rulează în fiecare subsistem. De exemplu, pentru joburi de server, este posibil să doriți să izolați toate joburile serverului de bază de date pe un singur subsistem, joburile serverului de comandă de la distanță pe un subsistem diferit, joburile serverului DDM pe încă un subsistem diferit și așa mai departe. În plus, prin utilizarea mai multor subsisteme puteți izola grupuri de joburi cu propriile lor pool-uri de memorie. În acest mod, un grup nu are un impact advers asupra celorlalte joburi.

Impact de timp de nefuncționare redus pentru utilizatori

De exemplu, dacă în fiecare Vineri după-amiază trebuie să aduceți sistemul în starea restricționată pentru scopuri de copiere de rezervă, puteți deconecta utilizatorii în mod gradual, oprind câte un subsistem pe rând.

Accesibilitate și disponibilitate crescută

Având un singur subsistem care realizează lucrul pentru mai puțini utilizatori, subsistemul este mai puțin ocupat și poate fi mai sensibil la cererile de lucru pe care le tratează.

Toleranță crescută la erori în subsistemele interactive

Împrăștiind lucrul pe mai multe subsisteme, dacă ar surveni o eroare de rețea, mai multe subsisteme ar putea gestiona procesul de recuperare a dispozitivului.

Timp îmbunătățit de pornire pentru un subsistem interactiv

Puteți păstra timpii de pornire ai subsistemului mai scurți subdivizând munca pe mai multe subsisteme.

Opțiuni suplimentare pentru reglarea de performanță

Utilizând mai multe subsisteme puteți seta subsistemele cu un număr mai mic de intrări de rutare.

Descrierea subsistem

O descriere de subsistem este un obiect de sistem care conține informații ce definesc caracteristicile unui mediu de operare controlat de către sistem. Identificatorul recunoscut de sistem pentru tipul obiectului este *SBSD. O descriere de subsistem definește cum, unde și cât de mult lucru intră într-un subsistem și ce resurse utilizează subsistemul pentru a realiza lucrul. Un subsistem activ ia numele simplu al descrierii subsistem.

Asemeni unui set de schițe detaliate, fiecare descriere de subsistem este unică, conținând caracteristicile specifice care descriu subsistemul. O descriere include locul unde lucrul poate intra în sistem, cât lucru poate trata un subsistem, cât spațiu de stocare principal (memorie) va fi utilizat și cât de repede pot rula joburile din subsistem.

Puteți utiliza o descriere de subsistem livrată cu sistemul dumneavoastră (făcând sau nu modificări asupra sa), sau puteți să vă creați propria descriere de subsistem.

Atribute descriere subsistem:

Atributele de descriere subsistem sunt atribute de sistem generale, obișnuite. Când creați un subsistem, primul pas este să definiți atributele subsistemului.

Atributele subsistemului includ:

- Numele descrierii subsistemului și biblioteca unde aceasta este memorată
- Toate definițiile de pool de memorie pe care acest subsistem le utilizează

o definiție de subsistem poate avea specificate maxim 10 definiții de pool de memorie. În această definiție de subsistem sunt incluse:

- Identificatorul de definiție pool: Acesta este identificatorul aflat în descrierea subsistemului, al definiției de pool de spațiu de stocare.
- Dimensiune: Aceasta este dimensiunea pool-ului de spațiu de stocare exprimat în multipli de KB (1KB=1024 octeți) și este cantitatea din memoria principală pe care pool-ul o poate utiliza.
- Nivel de activitate: Acesta este numărul maxim de fire de execuție care pot rula în același timp în pool.
- Numărul maxim de joburi care pot fi active în subsistem în același timp
- O descriere text a descrierii subsistemului
- Numele și biblioteca fișierului de afișare pentru semnare care este utilizat pentru a arăta ecrane de semnare la stațiile de lucru care sunt alocate subsistemului.
- Un nume de bibliotecă a subsistemului pe care îl puteți utiliza dacă doriți să specificați o bibliotecă ce ar trebui introdusă înaintea altor biblioteci în porțiunea de sistem a listei de biblioteci (Acest parametru vă permite să utilizați o bibliotecă de limbaj secundară.)

În descrierea subsistemului mai sunt incluse informații despre nivelurile de autorizare la subsistem. Aceste informații sunt păstrate de Securitate și nu sunt memorate cu celelalte atribute ale descrierii subsistemului. Puteți vizualiza autorizarea descrierii subsistemului utilizând comanda (DSPOBJAUT)(Display Object Authority - Afișare autorizare obiect).

Intrări de lucru:

Intrările de lucru identifică sursele unde joburile pot intra într-un subsistem. Sunt utilizate tipuri anumite de intrări de lucru, pentru tipuri diferite de joburi. Intrările de lucru sunt părți ale descrierii subsistemului.

Următoarele informații descriu tipurile diferite de intrări de lucru și modul în care să le gestionați. Există cinci tipuri de intrări de lucru: intrări de job autostart, intrări de comunicații, intrări în coadă de joburi, intrări de job prestart și intrări de stație de lucru.

Intrări de joburi autostart:

Intrările joburilor autostart identifică joburile autostart care vor porni imediat ce subsistemul pornește. Când un subsistem pornește, sistemul alocă mai multe elemente și pornește joburile autostart și joburile prestart înainte de a fi gata să lucreze.

Joburile autostart asociate cu un subsistem sunt pornite automat de fiecare dată când este pornit subsistemul. Un job autostart din subsistemul de control poate fi utilizat pentru a porni alte subsisteme (precum face subsistemul de control livrat de IBM). Un job autostart este un job batch care face muncă repetitivă.

De exemplu: Pentru a apela un program de recuperare special, dacă IPL simte că oprirea anterioară a sistemului a fost anormală, puteți adăuga o intrare de job autostart la descrierea de subsistem pentru subsistemul de control. Acest program verifică valoarea de sistem QABNORMSW, a stării opririi anterioare a sistemului. Pentru o oprire normală a sistemului, valoarea lui QABNORMSW este '0' și pentru o oprire anormală a sistemului, valoarea lui QABNORMSW este '1'.

Intrări de comunicații:

Intrarea de lucru pentru comunicații îi identifică subsistemului sursele pentru jobul de comunicații pe care acesta îl va procesa. Procesarea jobului începe când subsistemul recepționează o cerere de pornire a programului de comunicații de la un sistem aflat la distanță și se găsește o intrare de rutare corespunzătoare pentru cerere.

Din motive de performanță, în loc să fie pornit un job de comunicații de fiecare dată când este recepționată o cerere de pornire program, puteți configura un job prestart să trateze o cerere de pornire program de la un sistem aflat la distanță.

Pentru ca un job de comunicații batch să ruleze pe un sistem iSeries, trebuie să existe pe sistem o descriere de subsistem care să conțină o intrare de lucru pentru comunicații.

Intrări în coada de joburi:

Intrările în coada de joburi dintr-o descriere de subsistem specifică din ce cozi de joburi urmează să se recepționeze joburi. Când este pornit subsistemul, subsistemul încearcă să aloce fiecare coadă de joburi definită în intrările în coada de joburi a subsistemului.

De exemplu, o intrare în coada de joburi din descrierea subsistemului QSYS/QBASE specifică faptul că joburile pot fi pornite utilizând coada de joburi QGPL/QBATCH. Joburile pot fi puse într-o coadă de joburi chiar dacă subsistemul nu a fost pornit. Când subsistemul QBASE este pornit, el procesează joburile din coadă. O descriere de subsistem poate specifica numărul maxim de joburi (batch sau interactive) care pot fi procesate în același timp. Numărul de joburi care pot fi active din orice coadă de joburi este specificat în intrarea în coada de joburi.

Intrări de job prestart:

Definiți jobul prestart utilizând o intrare de job prestart. O intrare de job prestart nu afectează alocarea dispozitivelor sau alocarea cererii de pornire program.

Atributele de job ale unui job prestart nu sunt modificate de către subsistem când o cerere de pornire program se atașează la un job prestart. Totuși, joburile de server, în general, modifică atributele de job cu cele ale profilului utilizator cu care s-a făcut schimb (swap).

Comanda CHGPJ (Change Prestart Job - Modificare job prestart) permite jobului prestart să modifice unele din atributele de job cu cele ale descrierii de job (specificate în descrierea de job asociată cu profilul utilizator al cererii de pornire program sau în descrierea de job specificată în intrarea jobului prestart).

Joburi prestart pentru servere:

În modelul de job prestart există un job de ascultare primară, numit, în general, jobul demon sau jobul ascultător, și mai multe joburi de server care procesează cererile clienților. Jobul demon ascultă pe port pentru cereri de conexiune. Când este recepționată o nouă conexiune, demonul face o muncă generală, apoi dă descriptorul de socket unui job de server prestart aflat în așteptare.

Joburile prestart pot fi reutilizate. Când jobul a terminat lucrul pentru un client, mediul este resetat și jobul este pus la dispoziție pentru a trata o cerere de la alt client.

Pentru joburile de server care rulează codul de utilizator (de exemplu, serverul de comandă de la distanță), jobul nu este, de obicei, reutilizat. Aceasta deoarece codul de utilizator este posibil să fi modificat ceva în job și nu există o cale sigură de a reseta mediul pentru un client nou. Dacă serverul nu reutilizează jobul, API-ul Change Job (QWTCHGJB) poate fi utilizat pentru a modifica atributul jobului înapoi la o stare cunoscută după ce s-a efectuat cererea clientului.

Serverele care utilizează modelul de job prestart includ serverele gazdă, serverul SMTP, servere PPP, serverul DDM/DRDA, serverul SQL și altele.

Intrări în stația de lucru:

Un job interactiv este un job care pornește atunci când utilizatorul se semnează pe o stație de afișare și se termină utilizatorul iese. Pentru ca jobul să ruleze, subsistemul caută descrierea jobului, care poate fi specificată în intrarea în stația de lucru sau în profilul utilizator.

Intrarea în stația de lucru ghidează subsistemul către stații de lucru viitoare. Dacă o stație de lucru este disponibilă, subsistemul trimite un ecran de semnare către stația de afișare.

Notă: Descrierea de subsistem pentru subsistemul de control trebuie să conțină o intrare de stație de lucru pentru consolă și acea intrare trebuie să fie de tipul *SIGNON. (*SIGNON este o valoare pentru parametrul AT, specificat în comanda (ADDWSE) (Add Work Station Entry - Adăugare intrare stație de lucru.) Valoarea *SIGNON indică faptul că ecranul de semnare este afișat la stația de lucru când este pornit subsistemul. Această cerință asigură că subsistemul are un dispozitiv interactiv pentru comenzi de intrare sistem și nivel subsistem. Comanda (ENDSYS)(End System - Oprire sistem) oprește programul cu licență iSeries la o singură sesiune (sau ecran de semnare) la consola din subsistemul de control. O descriere de subsistem care nu conține o intrare de stație de lucru pentru consolă nu poate fi pornită ca subsistem de control.

Intrări rutate:

Intrarea de rutare identifică pool-ul de spațiu de stocare principal al subsistemului care trebuie utilizat, programul de control care trebuie rulat (de obicei programul QCMD livrat cu sistemul) și informații de run-time suplimentare (memorate în obiectul clasă). Intrările de rutare sunt memorate în descrierea subsistemului.

O intrare de rutare poate fi asemănată cu o singură intrare dintr-un îndrumar de mall de cumpărături. Clienții care nu pot găsi magazinul de care au nevoie pot utiliza un îndrumar care să îi ajute arătându-le direcția corectă. Același lucru este valabil și pe sistemul iSeries. Intrările de rutare ghidează jobul către locul corect. Intrările de rutare dintr-o descriere de subsistem specifică programul care să fie apelat pentru a controla un pas de rutare pentru un job care rulează în subsistem, ce pool de memorie va utiliza jobul și din ce clasă să obțină atributele de runtime. Datele de rutare identifică o intrare de rutare pe care jobul să o utilizeze. Împreună, intrările de rutare și datele de rutare furnizează informații despre pornirea unui job într-un subsistem.

Intrările de rutare conțin următoarele părți: descrierea subsistemului, clasa, date de comparație, maximul de pași de rutare activi, ID-ul pool de memorie, programul de apelat, afinitatea resurselor firelor de execuție, grupul de afinitate a resurselor și numărul de ordine.

Clasă:

Atributele de run sunt conținute în obiectul de clasă care este specificat în parametrul (CLS) din intrarea de rutare. Dacă un job conține mai mulți pași de rutare, clasa utilizată de fiecare pas de rutare următor este specificată în intrarea de rutare utilizată pentru a porni pasul de rutare. Dacă clasa nu există când este adăugată intrarea de rutare, trebuie specificat un calificativ de bibliotecă deoarece numele de clasă calificat este păstrat în descrierea subsistemului.

Atributele de runtime care sunt incluse într-o clasă de intrare de rutare sunt:

Prioritatea de rulare (RUNPTY)

Prioritatea de rulare este o valoare din intervalul de la 1 (cea mai mare prioritate) la 99 (prioritatea cea mai mică) care reprezintă prioritatea la care jobul se află în competiție pentru unitatea de procesare relativ la celelalte joburi care sunt active în același timp. Pentru joburi cu mai multe fire de execuție, prioritatea de rulare este, de asemenea, cea mai mare prioritate de rulare permisă pentru orice fir de execuție din job. Firele de execuție individuale din job este posibil să aibă o prioritate mai mică.

Felie de timp (TIMESLICE)

Aceasta este felia de timp care stabilește durata necesită de firul de execuție pentru a realiza o cantitate semnificativă de procesare. La sfârșitul feliei de timp, firul de execuție poate fi pus într-o stare inactivă astfel încât celelalte fire de execuție să poată deveni active în pool-ul de spațiu de stocare.

Timp de așteptare implicit (DFTWAIT)

Acesta specifică timpul maxim implicit (în secunde) cât așteaptă un fir de execuție dintr-un job ca o instrucțiune de sistem, cum ar fi instrucțiunea interfeței mașină LOCK (MI), să obțină o resursă. Acest timp de așteptare implicit este utilizat atunci când nu este altfel specificat un timp de așteptare pentru o situație dată. În mod normal, aceasta este durata cât utilizatorul sistemului ar fi dispus să aștepte sistemul înainte ca cererea să fie oprită. Dacă timpul de așteptare pentru orice instrucțiune este depășit, poate fi afișat un mesaj de eroare sau aceasta poate fi tratată automat de către o comandă MONMSG (Monitor Message - Mesaj monitor).

Timp CPU maxim (CPUTIME)

Acesta specifică timpul maxim al unității de procesare (în milisecunde) pe care un job îl poate utiliza. Dacă

jobul conține mai mulți pași de rutare, fiecare pas de rutare are permisiunea de a utiliza această cantitate din timpul unității de procesare. Dacă timpul maxim este depășit, jobul este oprit.

Spațiu de stocare temporar maxim (MAXTMPSTG)

Acesta specifică cantitatea maximă de spațiu de stocare temporar (auxiliar) pe care un job îl poate utiliza. Dacă jobul conține mai mulți pași de rutare, acesta este maximul de spațiu de stocare temporar pe care pasul de rutare îl poate utiliza. Acest spațiu de stocare temporar este utilizat pentru spațiul de stocare necesitat de programul însuși și de obiectele de sistem interne create implicit utilizate pentru a ajuta jobul. Nu include spațiul de stocare din biblioteca QTEMP. Dacă este depășit spațiul de stocare temporar maxim, jobul este oprit. Acest parametru nu se aplică utilizării spațiului de stocare permanent care este controlat prin profilul utilizator.

Număr maxim de fire de execuție (MAXTHD)

Acesta specifică numărul maxim de fire de execuție pe care un job care utilizează această clasă le poate rula în orice moment. Dacă mai multe fire de execuție sunt inițiate simultan, această valoare poate fi depășită. Dacă valoarea maximă este depășită, firele de execuție în exces vor avea permisiunea de a rula până la terminarea lor normală. Inițierea de fire de execuție suplimentare va fi interzisă până când numărul maxim de fire de execuție din job scade sub această valoare maximă.

Descriere text (TEXT)

Aceasta specifică textul care descrie pe scurt obiectul. Acesta este un atribut al obiectului de clasă când este creat, dar nu este un atribut de runtime pentru un job.

Autorizare (AUT)

Aceasta specifică autorizarea pe care o dați utilizatorilor care nu au autorizare specifică pentru obiect, care nu sunt pe o listă de autorizație, și a căror profil de grup sau profiluri de grup suplimentare nu au autorizare specifică pentru obiect. Acesta este un atribut al obiectului de clasă atunci când este creat dar nu este un atribut de runtime pentru un job.

Date de comparație:

Parametrul Valoare de comparație (CMPVAL) al intrării de rutare specifică datele care sunt comparate cu datele de rutare pentru a determina ce intrare de rutare să se utilizeze. (Intrarea de rutare mai specifică, de asemenea, poziția de pornire pentru comparație.) Datele de rutare sunt comparate cu valoarea de comparație a fiecărei intrări, în ordinea numărului de ordine, până se găsește o potrivire. Numărul de ordine conținut într-o intrare de rutare definește ordinea în care sunt scanate intrările de rutare și poate fi utilizat ca identificator al intrării de rutare.

Când o intrare de rutare este găsită cu o valoare de comparație care se potrivește cu datele de rutare, este pornit un pas de rutare și este apelat programul specificat în intrarea de rutare. Atributele de runtime din clasa asociată cu intrarea de rutare sunt utilizate pentru pasul de rutare și pasul de rutare rulează în pool-ul de spațiu de stocare specificat în intrarea de rutare.

Puteți specifica o valoare de comparație *ANY în intrarea de rutare cu numărul cel mai mare. *ANY înseamnă că este forțată o potrivire indiferent de datele de rutare. Doar o intrare de rutare poate conține valoarea de comparație *ANY, și trebuie să fie ultima intrare (cu numărul de ordine cel mai mare) din descrierea subsistemului.

Număr maxim de pași de rutare activi:

Parametrul Număr maxim de pași de rutare activi (MAXACT) al intrării de rutare specifică numărul maxim de pași de rutare (joburi) care pot fi activi în același timp prin această intrare de rutare.

Într-un job, doar un pas de rutare este activ la un moment dat. Când un subsistem este activ și s-a atins numărul maxim de pași de rutare, orice încercare ulterioară de pornire a unui pas de rutare prin intrarea de rutare eșuează. Jobul care a încercat să pornească pasul de rutare este oprit și este trimis un mesaj de către subsistem către istoricul jobului.

În mod obișnuit nu este nici un motiv pentru a controla numărul de pași de rutare, astfel valoarea recomandată este *NOMAX.

ID pool de memorie:

Parametrul ID pool de memorie (**POOLID**) al intrării de rutare specifică identificatorul de pool al pool-ului spațiului de stocare în care rulează programul. Identificatorul de pool specificat aici are legătură cu pool-urile de spațiu de stocare din descrierea subsistemului.

Programul de apelat

Parametrul Program de apelat (**PGM**) al intrării de rutare specifică numele și biblioteca programului apelat ca primul program rulat în pasul de rutare. Nu poate fi transmis nici un parametru programului specificat. Numele programului poate fi fie specificat explicit în intrarea de rutare sau extras din datele de rutare.

Dacă numele programului este specificat într-o intrare de rutare, selecția acelei intrări de rutare are ca rezultat programul intrării de rutare care este apelat (indiferent de numele programului transmis într-o funcție EVOKE). Dacă programul specificat în funcția EVOKE trebuie să fie apelat, trebuie specificat *RTGDTA în acest parametru. Dacă programul nu există când este adăugată sau modificată intrarea de rutare, trebuie specificat un calificativ de bibliotecă deoarece numele de program calificat este păstrat în descrierea subsistemului.

Număr de ordine

Parametrul Număr de ordine (**SEQNBR**) al intrării de rutare îi spune subsistemului ordinea în care se va căuta în intrările de rutare o potrivire a datelor de rutare. În intrările de rutare se face căutarea conform numărului de ordine. Când adăugați intrări de rutare la o descriere de subsistem, ar trebui să le ordonați astfel încât intrările care este probabil să fie comparate mai des să fie primele. Aceasta reduce timpul de căutare.

Număr de ordine	Valoare de comparație
10	'ABC'
20	'AB'
30	'A'
40	'E'
50	'D'

În exemplul de mai sus, căutarea în intrările de rutare se face în ordinea numărului de ordine. Dacă datele de rutare sunt 'A', căutarea se termină cu intrarea de rutare 30. Dacă datele de rutare sunt 'AB', căutarea se termină cu intrarea de rutare 20. Dacă datele de rutare sunt 'ABC', căutarea se termină cu intrarea de rutare 10. Deoarece datele de rutare pot fi mai lungi decât valoarea de comparație a intrării de rutare, comparația (care se realizează în ordinea de la stânga la dreapta) se oprește când ajunge la sfârșitul valorii de comparație. De aceea, dacă datele de rutare sunt 'ABCD', căutarea se termină cu intrarea de rutare 10.

Când definiți intrări de rutare, ele trebuie să fie ordonate de la cea mai specifică la cea mai generală. Următorul exemplu arată o metodă corectă și una incorectă de a defini intrările de rutare:

Corect		Incorect	
Număr de ordine	Valoare de comparație	Număr de ordine	Valoare de comparație
10	'ABC'	10	'ABC'
20	'AB'	20	'ABCD'
30	'A'		
40	'E'		
9999	*ANY		

În exemplul incorect, nu mai este posibil să se potrivească intrarea de rutare 20 deoarece toate datele de rutare care se potrivesc cu valoarea de comparație pentru intrarea de rutare 20 se potrivesc mai întâi cu intrarea de rutare 10. Când o intrare de rutare este modificată sau adăugată la o descriere de subsistem cu o valoare de comparație care cauzează această situație, sistemul trimite un mesaj de diagnostic care identifică situația.

Programul denumit în intrarea de rutare primește controlul când este pornit pasul de rutare pentru job. Parametrii pentru controlul mediului de runtime (prioritate, felie de timp și așa mai departe) al pasului de rutare pentru job sunt luați din clasa specificată în intrarea de rutare.

Cum pornește un subsistem

Când pornește un subsistem, sistemul alocă mai multe elemente și pornește joburi autostart și prestart înainte ca subsistemul să fie gata de lucru.

Descrierea de subsistem este utilizată pentru a determina cum sunt alocate elementele. Următoarea listă reprezintă secvența de evenimente care apar când pornește subsistemul:

1. **Este lansată cererea de pornire subsistem.** Este lansată comanda Pornire subsistem (STRSBS). Informația de pornire importantă se află în descrierea subsistemului.
2. **Pool-urile de memorie sunt alocate.** Memoria este alocată pool-urilor definite în descrierea subsistem. Memoria care este alocată fiecărui pool definit este luată din pool-ul de memorie de bază. Sistemul nu alocă memorie unui pool dacă cantitatea de memorie disponibilă în pool-ul de memorie Bază este mai puțină decât dimensiunea minimă specificată în valoarea de sistem Dimensiune minimă pool de memorie de bază QBASPOOL. Dacă sistemul nu poate alocă toată memoria cerută, el alocă atâta memorie cât este disponibilă și alocă restul după ce apare memorie disponibilă.
3. **Joburile prestart sunt pornite.** Această informație vine de la intrările de job prestart.
4. **Joburile autostart sunt pornite.** Această informație vine de la intrările joburilor autostart.
5. **Stațiile de afișare sunt alocate (ecranele de semnare sunt sus).** Dacă sunt intrări de stație de lucru și dispozitivul este variat pe activat și nu a fost alocat de nici un alt subsistem, subsistemul poate îl alocă și afișa ecranul de semnare. Dacă dispozitivul este variat pe activat și a fost alocat de către alt subsistem și este în ecranul de semnare (ecranul de semnare a fost afișat înainte ca al doilea subsistem să fie pornit), un al doilea subsistem poate alocă dispozitivul din primul subsistem și poate afișa ecranul de semnare. Dacă dispozitivul nu este variat pe activat, subsistemul nu îl poate alocă. Arbitrul sistemului (QSYSARB) și joburile QCMNARB rețin blocări pe toate dispozitivele variate pe dezactivat. Intrările de stație de lucru furnizează informații despre ce dispozitive să verifice pentru alocare.

Notă: Pentru dispozitive de afișare virtuale, promptul de semnare este afișat când dispozitivul devine complet variat pe activat. Aceasta se întâmplă când un utilizator se conectează la iSeries utilizând descrierea de dispozitiv (presupunând că cererea de conexiune nu conține date care sunt utilizate pentru a ocoli procesarea promptului de semnare). Poate fi luat un dispozitiv dintr-un pool de descrieri de dispozitive create anterior și variat pe activat ca parte a procesării acelei conexiuni, sau poate fi creat un dispozitiv și variat pe activat. Ca o pornire de subsistem, subsistemul va aștepta o blocare pentru oricare din descrierile de dispozitiv create anterior pe care le dorește.

6. **Cozile de joburi sunt alocate.** Subsistemul nu va fi capabil să aloce o coadă de joburi dacă este deja alocată pentru un alt subsistem. Această informație provine de la intrările de coadă de joburi.
7. **Dispozitivele de comunicații sunt alocate.** Cererile sunt trimise către jobul de sistem QLUS (servicii LU), care tratează alocarea dispozitivelor pentru toate dispozitivele de comunicații. Această informație provine de la intrările de comunicații.
8. **Mediul este pregătit de lucru.**

Cum sunt alocate dispozitivele stație de lucru:

Subsistemele încearcă să aloce toate dispozitivele de stație de lucru din descrierea lor de subsistem pentru intrări de stație de lucru AT(*SIGNON).

Pot apărea următoarele situații la pornirea unui subsistem:

- Dacă dispozitivul nu este variat pe activat, subsistemul nu îl poate aloca. Arbitrul de sistem (QSYSARB) și joburile QCMNARBxx rețin blocări pe toate dispozitivele variate pe dezactivat.
- Dacă dispozitivul este variat pe activat și nu a fost locat de nici un alt subsistem, subsistemul îl poate aloca și afișa ecranul de semnare.
- Dacă dispozitivul este variat pe activat și a fost alocat de către alt subsistem și este în ecranul de semnare (ecranul de semnare a fost afișat înainte ca al doilea subsistem să fie pornit), un al doilea subsistem poate aloca dispozitivul din primul subsistem și poate afișa ecranul de semnare.

Dacă mai mult de un subsistem încearcă să aloce aceeași stație de lucru (după cum este specificat în intrările stație de lucru) și stația de lucru este variată pe dezactivat, nu se poate prezice subsistemul care obține stația de lucru când aceasta este variată pe activat. În mod asemănător, dacă o intrare de stație de lucru specifică un tip de stație de lucru în locul unui nume de stație de lucru, un subsistem poate primi toate, unele, sau nici unele din stațiile de lucru de acel tip. (Aceasta se aplică, de asemenea, intrărilor de stație de lucru cu nume generale.) Pentru a evita o astfel de situație, puteți seta intrările de stație de lucru pentru subsisteme astfel încât mai multe subsisteme să nu utilizeze aceleași stații de lucru.

După ce un utilizator s-a semnat

Când un utilizator se semnează pe o stație de lucru, jobul rulează în subsistemul care a fost arătat în ecranul de semnare pe stația de lucru (subsistemul este identificat în ecranul de semnare livrat de IBM). Următoarele situații este posibil să apară după ce utilizatorul s-a semnat:

- Dacă un al doilea subsistem este pornit și încearcă să aloce stația de lucru pe care utilizatorul s-a semnat, al doilea subsistem nu o poate aloca. Jobul utilizatorului continuă să ruleze în primul subsistem.
- Dacă utilizatorul selectează opțiunea 1 (Afișare semnare pentru joburi alternative) din meniul Cerere sistem sau lansează comanda TFRSECJOB (Transfer to Secondary Job - Transfer la job secundar), noul job rulează în același subsistem ca și jobul original.
- Când utilizatorul renunță la semnare, stația de lucru rămâne alocată subsistemului folosit când utilizatorul s-a semnat, doar dacă utilizatorul nu s-a transferat în subsistem utilizând comanda TFRJOB (Transfer Job - Transfer job) și a specificat AT(*ENTER) pentru intrarea de stație de lucru pentru această stație de lucru. Este arătat un ecran de semnare și toate joburile următoare din stația de lucru continuă să ruleze în acel subsistem, (doar dacă nu este pornit alt subsistem care alocă stația de lucru în timp ce el este la ecranul de semnare).
- Dacă utilizatorul renunță la semnare și subsistemul în care rula jobul este oprit, dispozitivul este dezalocat. Un al doilea subsistem poate atunci să aloce dispozitivul și să afișeze ecranul de semnare.

Scenariu: Alocare de stație de lucru:

Următorul este un exemplu de cum sunt alocate două stații de lucru la două subsisteme diferite.

În acest scenariu, subsistemul A și subsistemul B au stațiile de lucru DSP01 și DSP02 în descrierile lor de subsistem (intrările de stație de lucru specifică AT(*SIGNON)).

Nume dispozitiv	Alocat la
DSP01	Subsistem A
DSP02	Subsistem A

Presupunem că ambele stații de lucru sunt variate pe activat când este pornit subsistemul A.

Subsistemul A alocă ambele stații de lucru și arată ecranul de semnare pe ambele. Deși subsistemul A are ecranul de semnare afișat pe stațiile de lucru, ele pot fi alocate de către alt subsistem sau job; stația de lucru nu va mai fi atunci disponibilă subsistemului A.

Nume dispozitiv	Alocat la
DSP01	USER1

Nume dispozitiv	Alocat la
DSP02	Subsistem A

Când un utilizator (USER1) se semnează pe stația de lucru DSP01, dispozitivul este alocat jobului utilizatorului USER1, care rulează în subsistemul A. Stația de lucru DSP02 este încă la ecranul de semnare. Așa încât poate fi alocată de un alt subsistem sau job. Atunci nu mai este disponibilă subsistemului A.

Nume dispozitiv	Alocat la
DSP01	USER1
DSP02	Subsistem B

Subsistemul B este pornit. Deoarece USER1 s-a semnat pe stația de lucru DSP01, subsistemul B nu poate alocă dispozitivul. Subsistemul B cere alocarea dispozitivului când el devine disponibil. DSP02 este alocată subsistemului B deoarece nimeni nu s-a semnat pe ea în subsistemul A. Orice job pornit pe DSP02 va rula în subsistemul B.

Nume dispozitiv	Alocat la
DSP01	Subsistem A
DSP02	Subsistem B

USER1 renunță la semnare. Deoarece jobul utilizatorului rula în subsistemul A, acel subsistem afișează ecranul de semnare astfel încât alt utilizator să se poată semna pe stația de lucru și să ruleze în subsistemul A. Dacă subsistemul A este oprit, stația de lucru DSP01 este alocată de către subsistemul B (deoarece acesta are o cerere de alocare a dispozitivului nerezolvată.)

Numele subsistemului care are alocată curent o stație de lucru apare în colțul din dreapta sus al ecranului de semnare livrat de IBM.

Pool-urile de memorie

Un pool de memorie este o diviziune logică a memoriei sau a spațiului de stocare principale, care este rezervat pentru procesarea unui job sau a unui grup de joburi. Pe serverul iSeries, tot spațiul de stocare principal poate fi împărțit în alocări logice numite pool-uri de memorie. În mod implicit, sistemul gestionează transferul de date și programe în pool-urile de memorie.

Pool-ul de memorie din care utilizatorul își obține memorie este întotdeauna același pool care limitează nivelul său de activitate. (Nivelul de activitate al unui pool de memorie este numărul de fire de execuție care pot fi active în același timp într-un pool de memorie.) Excepții de la aceasta fac joburile de sistem (precum Scpf, Qsysarb și Qlus) care își obțin memoria din pool-ul Bază dar care utilizează nivelul de activitate al pool-ului Mașină. Suplimentar, monitoarele de subsistem își obțin memoria din primul pool de descriere subsistem, dar utilizează nivelul de activitate al pool-ului Mașină. Aceasta permite monitorului de subsistem să fie mereu capabil să ruleze indiferent de setarea nivelului de activitate.

De ce să se utilizeze pool-uri de memorie

Puteți controla cât de multă muncă poate să fie realizată într-un subsistem controlând numărul și dimensiunea pool-urilor. Cu cât dimensiunea pool-urilor dintr-un subsistem este mai mare, cu atât mai multă muncă poate fi realizată în acel subsistem.

Utilizarea pool-urilor de memorie partajate permite sistemului să distribuie joburi pentru utilizatori interactivi prin mai multe subsisteme în timp ce permite încă joburilor lor să ruleze în același pool de memorie.

Pool-urile multiple dintr-un subsistem vă ajută să controlați competiția joburilor pentru resursele sistemului. Avantajele faptului că există pool-uri multiple într-un subsistem sunt acelea că puteți separa cantitatea de muncă făcută și timpul

de răspuns pentru aceste joburi. De exemplu, în timpul zilei este posibil să doriți ca joburile interactive să ruleze cu un timp de răspuns bun. Pentru o eficiență mai bună puteți mări pool-ul interactiv. Noaptea, este posibil să rulați mai multe joburi batch, deci veți mări pool-ul batch.

Notă: Deși ajustarea și gestionarea sistemului dumneavoastră poate crește eficiența fluxului de muncă din serverul dumneavoastră iSeries, nu poate conta pentru resurse hardware neadecvate. Luați în considerare posibilitatea unei modernizări hardware dacă cererile pentru volumul dumneavoastră de lucru sunt semnificative.

Modul în care sunt manipulate datele în pool-urile de memorie

Dacă datele sunt deja în spațiul de stocare principal, se poate face referire la ele independent de pool-ul de memorie în care acestea se află. Totuși, dacă datele necesare nu există în nici un pool de memorie, ele sunt aduse în același pool de memorie pentru jobul care a făcut referire la ele (aceasta este cunoscută sub numele de o pagină lipsă). Pe măsură ce datele sunt transferate într-un pool de memorie, alte date sunt înlocuite și, dacă sunt modificate, sunt automat înregistrate în spațiul de stocare auxiliar (aceasta se numește paginare). Dimensiunea pool-ului de memorie ar trebui să fie destul de mare pentru a păstra transferul de date (paginarea) la un nivel rezonabil deoarece rata afectează performanța.

Tipuri de pool-uri de memorie

Pe serverul iSeries, toată memoria principală poate fi împărțită în alocări logice numite *pool-uri de memorie*. Toate pool-urile de memorie dintr-un sistem sunt fie private sau partajate. Există pool-uri de memorie private, pool-uri de memorie partajate și pool-uri de memorie special partajate. Până la 64 de pool-uri de memorie, în orice combinație de pool-uri de memorie private sau partajate, pot fi active în același timp.

Pool-uri de memorie private

Pool-urile de memorie private (cunoscute, de asemenea, ca pool-uri de memorie definite de utilizator) conțin o anumită cantitate de memorie principală care poate fi utilizată de un singur subsistem pentru a rula joburi. Aceste pool-uri nu pot fi partajate de mai multe subsisteme. Ele sunt identificate în Navigatorul iSeries după numele subsistemului. Puteți avea până la 62 pool-uri de memorie private alocate pentru utilizare în subsistemele active.

Pool-uri de memorie partajate

Pool-urile partajate sunt fie speciale sau generale; pool-ul Mașină sau pool-ul Bază sunt considerate pool-uri partajate speciale și toate celelalte pool-uri partajate sunt considerate pool-uri partajate generale. Puteți specifica 63 din cele 64 de pool-uri de memorie partajate care sunt definite în sistem pentru utilizare atunci când se creează descrieri de subsistem (pool-ul Mașină este rezervat pentru uz de către sistem).

Pool-urile partajate speciale(*MACHINE și *BASE)

*MACHINE

Pool-ul de memorie Mașină este utilizat pentru programele de Mașină și de sistem de operare supra-partajate. El este identificat ca Mașină în Navigatorul iSeries. Pool-ul de memorie Mașină furnizează spațiu de stocare pentru operațiile pe care sistemul trebuie să le ruleze și care nu necesită atenția dumneavoastră. Dimensiunea pentru acest pool de memorie este specificată în valoarea de sistem Dimensiune pool de memorie Mașină (QMCHPOOL). Nici un utilizator de joburi nu rulează în acest pool de memorie. (În ecranul Gestionare stare sistem (WRKSYSSTS), pool-ul de memorie Mașină apare ca identificatorul de pool de sistem 1.)

*BASE

Pool-ul de memorie Bază, identificat ca Bază în Navigatorul iSeries, conține toată memoria principală nealocată din sistem, (toată memoria principală care nu este necesită de alt pool de memorie). Pool-ul de bază conține spațiul de stocare care poate fi partajat de către multe subsisteme. Pool-ul de memorie Bază este utilizat pentru lucrul batch și pentru diversele funcții de sistem. Valoarea de sistem QBASPOOL, dimensiunea minimă a pool-ului de memorie Bază, specifică dimensiunea minimă a pool-ului de memorie Bază. Nivelul de activitate pentru acest pool de memorie este specificat în valoarea de sistem QBASACTLVL, numărul maxim de fire de execuție eligibile pentru pool-ul de memorie Bază. (În ecranul Gestionare stare sistem (WRKSYSSTS), pool-ul de memorie Bază apare ca identificatorul de pool de sistem 2.)

Pool-urile partajate generale

Pool-urile partajate generale sunt pool-uri de memorie principală pe care mai multe subsisteme le pot utiliza simultan. În interfața bazată pe caractere, ele sunt identificate după cum urmează:

- *INTERACT este pool-ul de spațiu de stocare interactiv utilizat pentru joburile interactive.
- *SPOOL este pool-ul de spațiu de stocare utilizat pentru scriitorii de spool.
- De la *SHRPOOL1 la *SHRPOOL60 sunt pool-uri de spațiu de stocare pe care le puteți utiliza pentru uzul dumneavoastră personal.

În Navigator iSeries, pool-urile partajate generale sunt identificate ca Interactiv, Spool și Partajat 1 - Partajat 60.

Scheme de numerotare pool

Pool-urile au două seturi de scheme de numerotare: una este utilizată într-un subsistem și una este pentru tot sistemul. Subsistemul utilizează un set de numere care se referă la pool-urile pe care le utilizează. Astfel, când creați sau modificați o descriere de subsistem puteți defini unul sau mai multe pool-uri și le puteți eticheta 1, 2, 3 și așa mai departe. Acestea sunt etichetări ale pool-urilor de subsistem și nu corespund cu numerele de pool afișate pe ecranul WRKSYSSTS (Work with System Status - Gestionare stare sistem).

Un set diferit de numere este utilizat pentru a ține socoteala tuturor pool-urilor din sistem. Ecranul WRKSBS (Work with Subsystems - Gestionare subsisteme) leagă identificatorii pool-urilor subsistemului și anteturile coloanelor la identificatorii pool-urilor sistemului.

```

Work with Subsystems
System: XXXXXXXX
Type options, press Enter.
 4=End subsystem 5=Display subsystem description
 8=Work with subsystem jobs

-----Subsystem Pools-----
Opt Subsystem Storage (M) 1 2 3 4 5 6 7 8 9 10
- NYSBS .48 2 4 5
- PASBS .97 2 6 5
- QINTER 11.71 2 3

Bottom

Parameters or command
====>
F3=Exit F5=Refresh F11=Display system data F12=Cancel
F14=Work with system status

```

Cum sunt numerotate pool-urile - exemplu

Următorul exemplu ilustrează cum sunt numerotate pool-urile.

Subsistemele		
CRTSBSD QINTER	CRTSBSD NYSBS	CRTSBSD PASBS
Pool-uri (1 *BASE)	Pool-uri (1 *BASE)	Pool-uri (1 *BASE)
(2 1200 25)	(2 500 3)	(2 1000 3)
	(3 *SHRPOOL2)	(3 *SHRPOOL2)
(Pool-uri sistem 2, 3)	(Pool-uri sistem 2, 4, 5)	(Pool-uri sistem 2, 5, 6)

După ce pornește QINTER, sunt alocate următoarele pool-uri:

Număr pool sistem	Descriere	QINTER
1	Pool *MACHINE	
2	Pool *BASE	1
3	Pool privat QINTER	2

După ce pornește NYSBS sunt alocate următoarele pool-uri:

Număr pool sistem	Descriere	QINTER	NYSBS
1	Pool *MACHINE (mașină)		
2	Pool *BASE	1	1
3	Pool privat QINTER	2	
4	Pool privat NYSBS		2
5	Pool partajat *SHRPOOL2		3

După ce PASBS pornește sunt alocate următoarele pool-uri:

Număr pool sistem	Descriere	QINTER	NYSBS	PASBS
1	Pool *MACHINE			
2	Pool *BASE	1	1	1
3	Pool privat QINTER	2		
4	Pool privat NYSBS		2	
5	Pool privat SHRPOOL2		3	3
6	Pool privat PASBS			2

Alocarea pool-ului de memorie

Când porniți un subsistem, sistemul încearcă să aloce pool-urile de spațiu de stocare definite utilizator care sunt definite în descrierea subsistemului pornit.

Dacă sistemul nu poate aloca tot spațiul de stocare cerut, el va aloca atât spațiu de stocare cât este disponibil și apoi va aloca restul de spațiu de stocare pe măsură ce devine disponibil. De exemplu, luați în considerare următoarea tabelă. Dacă sunt disponibili 700 KB și dacă *SHRPOOL2 este definit la 500 KB, atunci sunt alocați 300 KB primului pool de spațiu de stocare și 400 KB sunt alocați celui de al doilea pool de spațiu de stocare.

ID pool specificat în SBS	1	2
Spațiu de stocare cerut	300K	*SHRPOOL2
ID pool sistem	3	4
Spațiu de stocare alocat	300K	400K
Nivel de activitate	1	
Tip pool	Privat	Partajat

Pool-urile de stocare pe care le definiți scad dimensiunea pool-ului de memorie Bază atunci când sunt alocate. Sistemul îi aloca pool-ului privat doar atât spațiu de stocare cât are disponibil în pool-ul de memorie Bază. Valoarea de sistem QBASPOOL, dimensiune minimă pool de memorie Bază, determină dimensiunea minimă a pool-ului de bază.

Nivelul de activitate al pool-ului de memorie

Nivelul de activitate al unui pool de memorie este numărul de fire de execuție care pot folosi activ CPU în același timp într-un pool de memorie. Aceasta permite utilizarea eficientă a resurselor sistemului. Sistemul gestionează controlul nivelului de activitate.

Adesea în timpul procesului într-un fir de execuție, un program așteaptă o resursă a subsistemului sau un răspuns de la un utilizator al unei stații de lucru. În timp ce așteaptă, un fir de execuție încetează folosirea nivelului de activitate al pool-ului de memorie astfel încât alt fir de execuție care este gata să fie procesat să îi ia locul.

Când mai multe fire de execuție sunt pornite care pot rula în același timp firele de execuție în exces trebuie să aștepte să folosească unitatea de procesare (în mod normal această așteptare este scurtă). Nivelul de activitate al pool-ului de memorie vă permite să limitați cantitatea de memorie principală în diverse pool-uri de memorie în subsistemele dumneavoastră.

Numărul de fire de execuție care rulează (sau fire de execuție active) se referă la numărul de fire de execuție care sunt eligibile să concureze pentru procesor și acest număr se compară cu nivelul de activitate pentru un pool de memorie. În acest sens, firele de execuție active nu includ fire de execuție care așteaptă după intrare, un mesaj, un dispozitiv ca să fie alocat sau un fișier să fie deschis. Firele de execuție active nu includ fire de execuție care sunt ineligibile (fire de execuție care sunt gata dar nivelul de activitate al pool-ului de memorie este la maxim).

Cum funcționează nivelurile de activitate

Mai multe fire de execuție poate fi active în același timp într-un pool de memorie deoarece procesarea pentru un fir de execuție poate fi întrerupt până când datele necesare sunt extrase din spațiul de stocare auxiliar. În timpul acestei întârzieri, care este în mod normal scurtă, un alt fir de execuție poate rula. Folosind nivelul de activitate, mașina poate procesa un număr mare de fire dintr-un pool de memorie și în același timp să țină nivelul de conflict la limita la care o specificați.

Nivel maxim de activitate

După ce nivelul de activitate maxim pentru un pool de memorie a fost atins, firele de execuție suplimentare care au nevoie de pool-ul de memorie sunt plasate în stare ineligibilă pentru a aștepta ca numărul de fire de execuție active din pool-ul de memorie să cadă sub nivelul de activitate maxim sau pentru un fir de execuție să ajungă la finalul feliei sale de timp. Imediat ce un fir de execuție își termină utilizarea pool-ului de memorie, alte fire de execuție care nu sunt active devin eligibile să ruleze după prioritatea lor. De exemplu, dacă un fir de execuție care rulează așteaptă un răspuns de la o stație, el cedează nivelul de activitate și nivelul de activitate nu este mai lung decât maximumul său.

Definirea nivelurilor de activitate ale pool-urilor de memorie

Definirea pool-urilor de memorie și nivelurile de activitate sunt general dependente de dimensiunea pool-ului de memorie, numărul de CPU-uri, numărul de domenii elemente de disc, și caracteristicile aplicațiilor.

Joburile

Toată munca efectuată pe un sistem este realizată prin joburi. Fiecare job are un nume unic în sistem. Toate joburile, cu excepția joburilor de sistem, rulează în subsisteme. Un job poate intra în subsistem din orice intrare de lucru, precum o intrare într-o coadă de joburi, o intrare de comunicații, o intrare de stație de lucru, o intrare de job autostart sau o intrare de job prestart.

Fiecare job activ conține ce puțin un fir de execuție (firul de execuție inițial) și poate conține fire de execuție adiționale secundare. Firele de execuție sunt unități independente ale lucrului. Atributele de job sunt partajate de-a lungul firelor de execuție totuși firele de execuție au unele atribute proprii, cum ar fi o stivă de apeluri. Atributele joburilor conțin informații despre modul cum este procesat lucrul. Jobul servește ca proprietar pentru atributele care sunt partajate de-a lungul firelor de execuție, în același job. Control funcționare vă furnizează o cale de a controla, prin atributele unui job, lucrul care se face pe sistemul dumneavoastră.

Autorizare corespunzătoare

Pentru a face majoritatea modificărilor asupra atributelor unui job trebuie să aveți fie autorizare specială de control job (*JOBCTL) sau profilul dumneavoastră de utilizator trebuie să se potrivească cu identitatea de utilizator job a jobului care este modificat.

Există puține atribute pentru care este necesară autorizarea specială *JOBCTL pentru a face orice modificare. Aceste atribute sunt:

- Timp de așteptare implicit
- Prioritate la rulare
- Felie de timp

Notă: Dacă plănuieți să faceți modificări asupra codului de contabilizare a jobului, vă trebuie autorizare *USE pentru comanda CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare) în plus față de autorizarea specială *JOBCTL sau un profil utilizator care să se potrivească cu identitatea de utilizator job a jobului.

Pentru orice atribut de job care face referire la un obiect iSeries, precum cozile de joburi, cozile de ieșire și tabelele cu ordinea de sortare, trebuie să aveți autorizarea corectă la obiect. Pentru detalii suplimentare asupra autorizărilor iSeries,

vedeți Anexa D Autorizări necesare pentru obiectele utilizate de comenzi din Securitatea iSeries Referință .

Caracteristici job

Control funcționare vă furnizează o cale pentru a controla lucrul realizat în sistemul dumneavoastră prin atributele unui job. Totuși, înainte să puteți controla diversele aspecte ale unui job, trebuie să înțelegeți diferitele caracteristici ale unui job.

Următoarele informații descriu caracteristicile joburilor:

Sintaxă nume job:

Pentru a ușura controlul și identificarea joburilor în sistem, fiecare job are un nume de job calificat unic. Numele de job calificat conține trei părți: numele jobului (sau numele de job simplu), numele utilizator și numărul jobului.

- Pentru joburile interactive, numele jobului este același cu numele stației de lucru sau a sesiunii emulatorului pe care v-ați semnat. Pentru joburile batch puteți specifica propriul dumneavoastră nume de job. Numele jobului poate avea până la 10 caractere lungime.
- Numele de utilizator este numele profilului utilizator sub care este pornit jobul. Pentru joburile interactive, numele de utilizator este profilul utilizator folosit pentru semnarea pe sistem. Acesta este numele de utilizator pe care l-ați introdus în câmpul utilizator în ecranul de semnare. Dacă utilizați Telnet și ocolirea semnării, acesta este numele de utilizator pe care îl utilizați pentru a vă semna automat pe sistem. Pentru joburile batch puteți specifica profilul utilizator sub care este rulat jobul batch. Numele de utilizator poate avea până la 10 caractere lungime.
- Numărul de job este un număr unic alocat de către sistem astfel încât să puteți identifica joburile, chiar dacă mai multe au același nume de job și același nume de utilizator. Numărul jobului este întotdeauna format din 6 cifre numerice.

Sintaxă

Sintaxa pentru numele de joburi calificate este similară cu numele calificate pentru obiecte. De exemplu, dacă numele jobului este DSP01, utilizatorul este QPGMR și numărul de job este 000578, numele calificat al jobului este introdus în comanda WRKJOB (Work with Job - Gestionare job) după cum urmează:

```
WRKJOB JOB(000578/QPGMR/DSP01)
```

O altă similitudine cu numele de obiecte este că nu trebuie să specificați toate calificativele. De exemplu luați în considerare următoarele:

```
WRKJOB JOB(QPGMR/DSP01)
```

sau

```
WRKJOB JOB(DSP01)
```

Aceasta funcționează ca și introducerea întregului nume calificat al jobului. Dacă mai multe joburi din sistem se potrivesc cu porțiunea din numele jobului pe care ați introdus-o, apare ecranul Selectare job. Acest ecran vă permite să selectați ce job doriți dintr-o listă de nume de job duplicat.

Atribute job:

Atributele joburilor determină modul în care sistemul rulează fiecare job. Unele atribute de job sunt setate din profilul utilizator. Alte atribute de job provin din valori de sistem, din Locale-uri, dintr-o comandă SBMJOB (Submit Job - Lansare job), dintr-o descriere de job și din comanda CHGJOB (Change Job - Modificare job) (din care puteți modifica valori pentru atribute în timp ce jobul rulează).

Controlul atributelor de job vă oferă flexibilitatea de a controla joburile la nivel de job, de utilizator sau de sistem. De exemplu, puteți seta sistemul dumneavoastră să meargă până la valoarea de sistem pentru a obține atribute de job (ceea ce reprezintă setarea implicită a sistemului). Apoi dacă doriți să modificați o valoare pentru toate joburile noi din sistem, puteți modifica valoarea de sistem.

Specificând o valoare dintr-o descriere de job, puteți afecta toate tipurile de joburi care utilizează acea descriere. De exemplu, dacă toate joburile dumneavoastră batch utilizează aceeași descriere de job, atunci modificarea descrierii de job pentru joburile batch poate afecta toate joburile dumneavoastră batch și poate lăsa celelalte joburi nemodificate.

Descriere de job:

Descrierea jobului vă permite să creați un set de atribute de job care sunt memorate și disponibile pentru utilizări multiple. Descrierea de job poate fi utilizată ca o sursă pentru unele dintre atributele de job care spun sistemului cum să ruleze un job. Atributele spun sistemului când să pornească un job, de unde să ia jobul și cum va rula jobul. Puteți să vă gândiți la o descriere de job ca la un șablon pe care îl utilizează multe joburi, în felul acesta reducând numărul de parametrii specifici pe care trebuie să îi setați pentru fiecare job individual.

Descrierile de job sunt utilizate de tipurile de job autostart, batch, interactive și prestart. Puteți folosi aceeași descriere de job pentru diverse joburi. Când definiți un job, puteți utiliza descrierea de job în unul din două moduri:

- Puteți utiliza o descriere de job fără să suprascrieți nici unul din atributele sale. De exemplu:

```
SBMJOB JOB(OEDAILY) JOB(QBATCH)
```

- Utilizați o descriere de job specificată dar suprascrieți unele din atribute (utilizând comanda BCHJOB sau SBMJOB). De exemplu, pentru a suprascrie înregistrarea în istoric a mesajului din descrierea de job QBATCH, specificați:

```
SBMJOB JOB(OEDAILY) JOB(QBATCH)  
LOG(2 20 *SECLVL)
```

Notă: Nu puteți suprascrie nici un atribut al unei descrieri de job autostart, job de stație de lucru sau de comunicații.

Descrieri de job și securitate:

Fiecare job din sistem utilizează o descriere de job în timpul inițializării jobului. Aceasta controlează diversele atribute ale unui job. Parametrul USER controlează numele unui profil utilizator alocat jobului. O descriere de job care are un nume de profil utilizator (USER) specificat, ar trebui să fie autorizată doar anumitor indivizi. Dacă nu, la nivelul de securitate 30 și mai jos, alți utilizatori vor fi capabili să lanseze joburi care să ruleze sub acel profil utilizator.

De exemplu, luați în considerare

```
CRTJOB JOB(X) USER(JONES) . . . AUT(*USE)
```

Acest exemplu are riscuri de securitate deoarece orice utilizator poate lansa un job utilizând descrierea de job X și poate fi autorizat la orice este autorizat și JONES. Dacă acest tip de descriere de job este utilizată într-o intrare de stație

de lucru, ea permite oricui să se semneze ca acel utilizator, doar apăsând tasta Enter. Pentru a evita orice expunere de securitate, nu autorizați acest tip de descriere pentru *PUBLIC.

Notă: La nivelurile de securitate 40 și 50, comanda SBMJOB (Submit Job - Lansare job) necesită ca cel care lansează să fie autorizat (*USE) la profilul utilizator numit în descrierea de job. Aceasta presupune că SBMJOB specifică utilizatorul (*JOBID). Cu toate acestea, evitați specificarea unui utilizator într-o descriere de job decât dacă este necesar pentru un anumit motiv (precum un job autostart) și controlați îndeaproape accesul la el.

Parametrul USER și joburile interactive

Descrierea de job care va fi utilizată este definită în comanda ADDWSE (Add Work Station Entry - Adăugare intrare stație de lucru). Setarea implicită este să se utilizeze descrierea de job din profilul utilizator. Dacă este specificat USER(*RQD) în descrierea de job, utilizatorul trebuie să introducă un nume de utilizator. Dacă este specificat USER(XXXX) (unde XXXX este un anumit nume de profil utilizator), utilizatorul are voie să apese tasta Enter în ecranul de semnare și să opereze sub numele de profil utilizator XXXX, doar dacă nivelul de securitate nu este 40 sau mai mare.

Parametrul USER și joburile batch

Descrierea de job utilizată pentru joburi batch este specificată în comanda SBMJOB (Submit Job - Lansare job) sau comanda BCHJOB (Batch Job - Job batch).

Dacă este introdus un flux de intrare care conține comanda BCHJOB, utilizatorul care introduce una din comenzile de Pornire cititor (STRDBRDR, STRDKTRDR) sau una din comenzile Lansare job (SBMDBJOB, SBMDKTJOB) și așa mai departe trebuie să aibă autorizare operațională de obiect (*OBJOPR) asupra descrierii de job care este specificată. Când este utilizat un flux de intrare, joburile operează întotdeauna sub profilul utilizator al descrierii de job și nu a utilizatorului care pune joburile în coada de joburi. Dacă este specificat USER(*RQD) în descrierea de job, nu este valid să se utilizeze descrierea de job pe o comandă BCHJOB.

Dacă este utilizată o comandă SBMJOB, comanda acționează implicit astfel încât jobul batch să opereze sub numele de profil utilizator al celui care a lansat comanda. Totuși, dacă este specificat USER(*JOBID) în comanda SBMJOB, jobul operează sub același nume specificat în parametrul USER al descrierii de job.

În mod frecvent este necesar un nume specific din descrierea de job pentru a permite utilizatorilor să lanseze lucrul pentru un anumit profil utilizator. De exemplu, descrierea jobului QBATCH este livrată cu USER(QPGMR) pentru a permite aceasta. Pentru a evita orice expunere de securitate, nu autorizați acest tip de descriere de job pentru *PUBLIC.

Stive de apeluri:

Stiva de apeluri este lista ordonată a tuturor programelor sau procedurilor care rulează momentan pentru un job. Programele și procedurile pot fi pornite explicit cu instrucțiunea CALL sau implicit dintr-un alt eveniment.

Stiva de apeluri este disponibilă atât la nivelul jobului cât și la nivelul firului de execuție. În interfața bazată pe caractere, stiva de apeluri este o listă ultimul-sosit-primul-plecat (LIFO) de intrări ale stivei de apeluri, o intrare pentru fiecare procedură sau program apelate. În Navigatorul iSeries, în mod implicit, ultima intrare din stivă apare în vârful listei. Totuși, ordonarea poate fi modificată utilizând butoanele **Sortare crescătoare** sau **Sortare descrescătoare**.

Informațiile care sunt incluse în ecranul Stivă de apeluri includ informațiile de invocare pentru modelul programului original (OPM), mediul de limbă integrat (ILE), Mediul de soluții pentru aplicații portabile i5/OS (PASE) și aplicații Java. De asemenea, dacă rulați sub un profil utilizator cu autoritate specială *SERVICE, veți putea să vizualizați intrări suplimentare pentru codul intern autorizat (LIC) și PASE i5/OS Kernel.

Obiect de clasă:

Un obiect de clasă conține atributele de rulare care controlează mediul runtime al unui job. Obiectele de clasă sau clasele livrate de IBM îndeplinesc atât necesitățile aplicațiilor interactive tipice cât și pe cele ale aplicațiilor batch. Următoarele clase (după nume) sunt livrate cu sistemul:

- QGPL/QBATCH: Pentru a fi utilizată de joburile batch
- QSYS/QCTL: Pentru a fi utilizată de subsistemul de control
- QGPL/QINTER: Pentru a fi utilizată de joburile interactive
- QGPL/QPGMR: Pentru a fi utilizată de subsistemul de programare
- QGPL/QSPL: Pentru a fi utilizată de scriitorul imprimantei subsistemului de depanare
- QGPL/QSPL2: Pentru a fi utilizată pentru depanare generală în pool-ul sistem Bază

Atribute de runtime

Următoarea este o listă a câtorva atribute de runtime sau parametri care se găsesc într-un obiect de clasă și care sunt importante pentru Control funcționare.

Prioritate de rulare (RUNPTY)

Un număr care specifică nivelul de prioritate alocat tuturor joburilor în curs de rulare care utilizează clasa. Nivelul de prioritate este utilizat pentru a determina care job, din toate joburile care concurează pentru resursele sistemului, va fi rulat următorul. Valoarea poate fi de la 1 la 99, unde 1 este cea mai mare prioritate (toate joburile care au prioritate 1 sunt rulate primele). Această valoare este cea mai mare prioritate de rulare permisă pentru orice fir de execuție din job. Firele de execuție individuale din job pot avea prioritate mai scăzută. Modificarea priorității de rulare a jobului va afecta prioritățile de rulare ale tuturor firelor de execuție din job. De exemplu, dacă jobul rulează la prioritate 10, firul de execuție A din job rulează la prioritatea 10 și firul de execuție B din job rulează la prioritatea 15. Dacă prioritatea jobului este modificată la 20, atunci prioritatea firului de execuție A este ajustată la 20 și prioritatea firului de execuție B este ajustată la 25.

Felie de timp (TIMESLICE)

Cantitatea maximă din timpul procesorului (în milisecunde) dată fiecărui fir de execuție din job utilizând această clasă, înainte ca altor fire de execuție din job sau altor joburi să li se dea oportunitatea de a rula. Felia de timp stabilește cantitatea de timp necesară unui fir de execuție dintr-un job pentru a realiza o cantitate semnificativă din procesare. La sfârșitul feliei de timp, firul de execuție este posibil să fie pus într-o stare inactivă astfel încât alte fire de execuție să poată deveni active în pool-ul de stocare.

Timp de așteptare implicit (DFTWAIT)

Durata implicită cât sistemul așteaptă terminarea unei instrucțiuni care realizează o așteptare. Acest timp de așteptare se aplică atunci când o instrucțiune așteaptă o acțiune a sistemului, nu atunci când o instrucțiune așteaptă un răspuns de la un utilizator. În mod normal, aceasta este cantitatea de timp cât dumneavoastră doriți să așteptați sistemul înainte de a opri cererea. Dacă timpul de așteptare este depășit, jobului îi este transmis un mesaj de eroare. Acest timp de așteptare implicit este utilizat atunci când nu este altfel specificat un timp de așteptare pentru o situație dată.

Timpul de așteptare utilizat pentru alocarea resurselor unui fișier este specificat în descrierea fișierului și poate fi înlocuit printr-o comandă de înlocuire. El specifică faptul că se utilizează timpul de așteptare specificat în obiectul de clasă. Dacă resursele fișierului nu sunt disponibile atunci când fișierul este deschis, sistemul așteaptă ca resursele să devină disponibile până când se termină timpul de așteptare.

Notă: Atributele de clasă se aplică fiecărui pas de rutare al unui job. Cele mai multe joburi au doar un pas de rutare, dar dacă jobul este rerutat (din cauza unor comenzi precum Rerutare job (RRTJOB) sau Transfer job (TFRJOB)) atributele clasei vor fi resetate.

Timp CPU maxim (CPUTIME)

Durata maximă din timpul procesorului permisă pentru terminarea procesării unui pas de rutare al unui job. Dacă pasul de rutare a jobului nu s-a terminat în această durată el este terminat și se scrie un mesaj în istoricul jobului.

Spațiu de stocare temporar maxim (MAXTMPSTG)

Cantitatea maximă de spațiu de stocare temporar care poate fi utilizată de către pasul de rutare al unui job. Acest spațiu de stocare temporar este utilizat pentru programele care rulează în job, pentru obiectele de sistem utilizate pentru a ajuta jobul și pentru obiecte temporare create de către job.

Număr maxim de fire de execuție (MAXTHD)

Numărul maxim de fire de execuție în care poate rula în orice moment un job din această clasă. Dacă sunt inițiate simultan mai multe fire de execuție, această valoare este posibil să fie depășită. Firele de execuție în exces vor avea permisiunea să ruleze și să se încheie normal. Inițierea unor fire de execuție suplimentare va fi împiedicată până când numărul maxim de fire de execuție din job va scădea sub valoarea maximă.

Notă: Resursele utilizate de către firele de execuție și resursele disponibile în sistem pot varia. De aceea, inițierea unor fire de execuție suplimentare este posibil să fie împiedicată înainte ca această valoare să fie atinsă.

Identitate utilizator job:

Identitatea utilizator job (JUID) este numele profilului utilizator după care acest job este cunoscut față de celelalte joburi. Acest nume este utilizat pentru verificări de autorizare atunci când alte joburi încearcă să opereze împotriva acestui job.

Unele exemple de funcții care operează împotriva altui job includ comanda STRSRVJOB (Start Service Job - Pornire service job), API-ul QUSRJOBI (Retrieve Job Information - Extragere informații job), API-ul QWTCHGJB (Change Job - Modificare job), toate comenzile de control job și funcțiile care trimit semnale de la un job la altul.

În situații în care joburile fac schimb de profil utilizator, profilul utilizatorului curent identifică profilul sub care rulează firul de execuție inițial în loc de JUID.

JUID nu este utilizat pentru a face verificări de autorizare din interiorul unui job. Autorizarea pentru realizarea unei funcții este întotdeauna bazată pe profilul utilizator curent al firului de execuție în care este apelată funcția.

Când un job este într-o coadă de joburi sau într-o coadă de ieșire, JUID-ul este întotdeauna același cu numele utilizatorului jobului și nu poate fi modificat.

Când un job pornește și la începutul oricărui pas de rutare următor JUID-ul este același cu numele profilului utilizator curent al jobului. În timp ce un job este activ, JUID-ul poate fi modificat în următoarele moduri.

- JUID poate fi setat explicit de către orice aplicație care utilizează API-ul QWTSJUID (Set Job User Identify - Setare identitate utilizator job) sau funcția QwtSetJuid(). JUID-ul este stat cu numele profilului utilizator sub care rulează firul de execuție care a apelat API-ul sau funcția.
- JUID poate fi curățat explicit de către o aplicație care utilizează API-ul QWTSJUID sau funcția QwtClearJuid(). Jobul trebuie să ruleze ca un job cu un singur fir de execuție la un moment dat. Când este curățat, JUID-ul este setat implicit de către sistem cu numele profilului utilizator sub care rulează singurul fir de execuție al jobului la acel moment.
- Dacă jobul rulează ca un job cu un singur fir de execuție și JUID-ul nu a fost setat explicit de către o aplicație, atunci de fiecare dată când jobul utilizează API-ul QWTSETP (Set Profile - Setare profil) pentru a rula sub un profil utilizator diferit JUID-ul este setat implicit de către sistem cu numele profilului utilizator care a fost setat de către QWTSETP.
- Când un job cu un singur fir de execuție inițiază un fir de execuție secundar și JUID-ul nu a fost setat explicit de către o aplicație, atunci sistemul va seta implicit JUID-ul cu numele profilului utilizator sub care rula singurul fir de execuție al jobului în momentul în care jobul a inițiat al doilea fir de execuție.

Când jobul se întoarce la un singur fir de execuție, sistemul setează implicit JUID-ul cu numele profilului utilizator sub care rulează singurul fir de execuție la acel moment.

Exemple de identitate a utilizatorului unui job:

Acest subiect conține exemple care vă vor ajuta să înțelegeți mai bine cum este alocată identitatea utilizatorului de job (JUID), în diferite situații.

- Un job rulează sub un profil utilizator numit USERA. JUID-ul este USERA. Dacă jobul utilizează API-ul QWTSETP pentru a comuta pe USERB, JUID-ul se modifică cu USERB.

În această situație, valoarea Setare după pentru JUID este *DEFAULT. Deoarece jobul care rulează are un singur fir de execuție, identitatea utilizatorului jobului este profilul utilizator curent sub care rulează firul de execuție inițial al jobului (doar dacă identitatea utilizatorului nu este setată explicit de către o aplicație). Pentru joburile din coada de joburi și pentru joburile terminate, identitatea utilizatorului jobului este numele utilizatorului din numele de job calificat.

- Un job cu un singur fir de execuție rulează sub profilul utilizator USERX. JUID-ul este USERX. Dacă jobul inițiază fire de execuție secundare, JUID-ul rămâne USERX. Dacă toate firele de execuție fac schimb la USERY, JUID-ul este încă USERX.

În această situație, valoarea Setare după pentru JUID este *SYSTEM. Deoarece acesta este un job activ care rulează momentan ca job cu mai multe fire de execuție, identitatea utilizatorului jobului este setată implicit de către sistem. Identitatea utilizatorului jobului este setată pe numele profilului utilizator sub care rula jobul când a devenit job cu mai multe fire de execuție. Când jobul rulează din nou ca job cu un singur fir de execuție, identitatea de utilizator al jobului va fi resetată pe valoarea *DEFAULT.

- Dacă un server care rulează sub un profil utilizator numit SERVER apelează API-ul QWTSJUID, JUID-ul va fi setat pe SERVER. Dacă serverul apelează atunci API-ul QWTSETP (Set Profile - Setare profil) pentru a seta profilul său curent de utilizator pe CLIENT în timp ce procesează lucrul în numele celui client, JUID-ul rămâne pe SERVER. De asemenea, dacă serverul inițiază fire de execuție secundare care fiecare apelează QWTSETP pentru a rula sub diverse profiluri utilizator, JUID-ul rămâne SERVER.

În această situație, valoarea Setare după pentru JUID este *APPLICATION. Identitatea utilizatorului jobului este setată explicit de către aplicație, utilizând un API. Această valoare se aplică atât joburilor cu un singur fir de execuție cât și celor cu mai multe fire de execuție.

Fire de execuție:

Termenul fir de execuție este stenografia pentru "fir de execuție de control". Un fir de execuție este calea luată de program în timp ce rulează, pașii realizați și ordinea în care sunt realizați pașii. Un fir de execuție rulează codul de la locația de pornire într-o ordine predefinită pentru un set dat de intrări.

Folosirea firelor de execuție dintr-un job permite executarea în același timp a mai multor lucruri. De exemplu, în timp ce un job procesează, un fir de execuție poate extrage și calcula date necesitate de către job pentru a termina de procesat

Fiecare job activ are cel puțin un fir de execuție, care este numit firul de execuție inițial. Firul de execuție inițial este creat ca parte a pornirii jobului. În firele de execuție din Navigatorul iSeries, în mod implicit, veți vedea **Inițial** ca fiind tipul primului fir de execuție din listă. Firul de execuție inițial este primul fir de execuție creat în job când el pornește.

Tipuri de fire de execuție

Tipul firului de execuție determină modul în care firul de execuție a fost creat sistemul.

Utilizator

Firul de execuție poate fi creat de către aplicația clientului. Firul de execuție inițial dintr-un job este mereu un fir de execuție utilizator. Câmpul Permite fire de execuție multiple trebuie să fie setat pe da pentru ca mai multe fire de execuție utilizator să fie folosite.

Sistemul

Firul de execuție este creat de către sistem în numele utilizatorului. Unele funcții sistem folosesc fire de execuție sistem pentru a completa procesarea. Dacă o aplicație client folosește o funcție sistem care folosește fire de execuție, sunt folosite fire de execuție sistem.

Autorizarea corespunzătoare a firului de execuție:


Înainte de a putea lucra cu firele de execuție sunt necesare anumite niveluri de autorizare.

Pentru a vizualiza și modifica cele mai multe atribute ale unui fir de execuție trebuie să aveți autorizare specială *JOBCTL sau profilul dumneavoastră de utilizator trebuie să se potrivească cu identitatea de utilizator a jobului care

conține firul de execuție. Pentru a modifica autorizarea de rulare a unui fir de execuție, trebuie să aveți autorizare specială *JOBCTL. Autorizarea de Control al firului de execuție vă va permite să vizualizați unele din atributele unui fir de execuție.

Pentru a reține sau elibera un fir de execuție, trebuie să aveți autorizare specială *JOBCTL sau autorizare Control fir de execuție sau profilul dumneavoastră utilizator trebuie să se potrivească cu identitatea utilizator job a jobului care conține firul de execuție. Pentru a termina un fir de execuție, trebuie să aveți autorizare specială *SERVICE sau autorizare Control fir de execuție.

Pentru orice atribut de fir de execuție care face referire la un obiect iSeries, cum ar fi o bibliotecă din lista de biblioteci, utilizatorul trebuie să aibă autorizarea corespunzătoare pentru obiect.

Pentru mai multe detalii asupra autorizărilor iSeries, vedeți Anexa D Autorizări necesare pentru obiectele utilizate de către comenzi din Referință de securitate iSeries .

Notă: Autorizarea de control fire de execuție vă permite să extrageți informații despre firele de execuție ale altui job. Controlul firelor de execuție poate fi acordat și revocat pentru utilizatori individuali utilizând suportul Administrare aplicații din Navigator iSeries sau utilizând API-ul Change Function Usage (QSYCHFUI), cu un ID de funcție QIBM_SERVICE_THREAD. Pentru mai multe informații detaliate despre administrarea aplicațiilor, vedeți subiectul Administrare aplicații din Centrul de informare.

Stare fir de execuție:

Starea curentă a unui fir de execuție este vizualizată din pagina General din fereastra Proprietăți fir de execuție sub starea Detaliat.

Un exemplu de o stare detaliată este:

Așteptarea unei scoateri din coadă

Firul de execuție al jobului așteaptă pentru terminarea operației de scoatere din coadă. O scoatere din coadă este o operație pentru îndepărtarea mesajelor din cozi. Mesajele sunt comunicații trimise de la o persoană sau program la altă locație. În particular, un mesaj este introdus (plasat) într-un obiect sistem coadă de către un fir de execuție și scos din coadă de către un alt fir de execuție.

Notă: Când se arată Așteptare pentru scoatere din coadă într-o pagină de proprietăți, sunt afișate informații care identifică acea coadă care este așteptată. Când un job sau un fir de execuție așteaptă ca operația de scoatere din coadă să se termine pentru un obiect i5/OS, veți vedea un nume de obiect de 10 caractere, biblioteca sa și tipul obiectului. Dacă jobul sau firul de execuție așteaptă ca operația de scoatere din coadă să se termine pentru un obiect intern, veți vedea un nume de obiect de 30 de caractere. Pentru obiectele interne aveți nevoie de autorizare specială de control job (*JOBCTL) pentru a vedea un nume de 30 de caractere.

Starea detaliată poate afișa o valoare de stare asociată, care furnizează detalii suplimentare despre starea curentă a firului de execuție. Un exemplu de stare detaliată la care se adaugă valoarea stării asociate este:

Reținut (n)

Un fir de execuție individual este reținut. Spre deosebire de un job, un fir de execuție poate avea mai multe rețineri pe el în același timp. Un număr (de exemplu, Reținut (3)) care urmează starea firului de execuție spune utilizatorului de câte ori firul de execuție a fost reținut fără să fi fost eliberat. De exemplu, dacă un fir de execuție a fost reținut de 3 ori și apoi a fost eliberat o dată, el are încă 2 rețineri asupra lui. Este arătat un număr numai când starea apare în pagina de proprietăți și nu va fi arătat când este afișată într-o listă. Pentru a relua procesarea firului de execuție, selectați acțiunea Eliberare pentru firul de execuție.

Pentru mai multe informații despre diferitele stări ale firelor de execuție, vedeți ajutorul online Navigator iSeries.

Obiecte blocate:

Joburile și firele de execuție folosesc obiecte pentru procesarea lucrului.

Deoarece mai multe părți din lucru sunt procesate în același timp, un blocaj este pus pe un obiect și astfel integritatea datelor este reținută. *Obiecte blocate* sunt obiecte de sistem folosite de joburi și fire de execuție pentru a procesa lucru. După ce jobul sau firul de execuție au terminat de rulat, obiectul este deblocat și gata de a fi utilizat pentru a procesa mai mult lucru. Dependența de tipul de cerere blocată folosită, căutați un obiect care permite doar unui utilizator să folosească un obiect în același timp. De exemplu, dacă doi sau mai mulți utilizatori încearcă să modifice un obiect în același timp, modificările făcute de al doilea utilizator asupra obiectului sunt blocate până când primul utilizator a terminat de actualizat obiectul. Cu utilizarea posesorilor blocați, un utilizator poate vedea ce a fost blocat curent sau este curent în așteptarea unui blocaj pentru un obiect.

Domeniu specifică dacă blocajul este asociat cu un job, un fir de execuție, sau un spațiu de blocare. Domeniu de asemenea definește cât timp este blocajul disponibil și ce tip de cerere blocată și reguli de conflict are obiectul.

Tipuri de cereri blocate sunt niveluri diferite ale accesării unui job, fir de execuție sau spațiu blocat pe care le poate avea un obiect care este blocat. De exemplu, o blocare exclusivă, blocare de tipul fără citire este utilizată dacă un obiect se modifică sau se șterge din sistem. Acest tip de cerere de blocare nu permite nimănui să utilizeze obiectul, nici măcar să îl citească.

Diferite tipuri de cereri de blocare:

Exclusiv-Fără citire

Obiectul este rezervat pentru utilizare exclusivă. Dar, dacă un obiect este blocat de altă cerere de blocare, nu puteți obține folosirea exclusivă a obiectului. Această stare de blocare este corespunzătoare când un utilizator nu dorește ca nici un alt utilizator să aibă acces la obiect până când funcția realizată este completă.

Exclusiv-Citire

Obiectul poate să fie partajat doar cu tipul de cerere blocare partajată - citire. Această blocare este corespunzătoare când un utilizator dorește să împiedice alți utilizatori să realizeze alte operații în afară de citire.

Partajat-Actualizare

Obiectul poate fi partajat fie cu tipul de cerere blocare partajată - citire, fie cu partajată - actualizare. Aceasta înseamnă că un alt utilizator poate cere fie starea de blocare partajată - citire, fie starea de blocare partajată - actualizare pentru același obiect. Această stare de blocare este corespunzătoare când un utilizator intenționează să modifice un obiect, dar dorește să permită altui utilizator să citească sau să modifice același obiect.

Partajat-Fără actualizare

Obiectul poate fi partajat numai cu tipuri de cerere blocare partajat - fără actualizare și cu partajat - citire. Această stare de blocare este corespunzătoare când un utilizator nu intenționează să modifice un obiect, dar dorește să se asigure că nici un alt utilizator nu modifică obiectul.

Partajat-Citire

Obiectul poate fi partajat cu toate cererile de blocare altele decât cu exclusiv - fără citire. Aceasta este când un alt utilizator poate cere o stare de blocare exclusivă - citire, partajată - actualizare, partajată - citire sau partajată - fără actualizare.

Starea blocării spune starea cererii de blocare. Diferitele stări de blocare sunt:

Reținut: Cererea de blocare a fost îndeplinită și jobul, firul de execuție sau spațiul de blocare reține blocarea.

În așteptare: Jobul sau firul de execuție așteaptă să obțină blocarea.

Cerut: Jobul sau firul de execuție a cerut blocarea.

Posesorii blocării sunt joburi, firele de execuție și spațiile blocate care sunt momentan reținute pentru un blocaj pe un anumit obiect blocat.

Tipuri de joburi

Serverul iSeries procesează mai multe tipuri de joburi diferite.

Joburi autostart:

Un job autostart este un job batch care face muncă repetitivă, muncă cu inițializare singulară care este asociată cu un anume subsistem, inițializează funcții pentru o aplicație sau furnizează funcții de service centralizat pentru alte joburi din același subsistem. Un job autostart din subsistemul de control poate fi utilizat pentru a porni alte subsisteme (precum face subsistemul de control livrat de IBM). Joburile autostart asociate cu un subsistem sunt pornite automat de fiecare dată când este pornit subsistemul.

Din moment ce toate joburile autostart sunt pornite când pornește subsistemul, valoarea specificată pentru numărul maxim de joburi din subsistem nu împiedică joburile autostart să pornească. Dacă numărul maxim de joburi este depășit, nici un alt job nu poate să înceapă. Când destule joburi autostart sunt terminate, deci numărul de joburi care rulează este mai mic decât nivelul maxim de activitate, alte joburi din sistem pot să înceapă.

Descrierea de job care este utilizată pentru un job autostart este specificată utilizând comanda (ADDAJE) (Add Autostart Job Entry - Adăugare intrare de job autostart). Când este pornit subsistemul, jobul operează sub numele de profil utilizator din descrierea jobului specificat. Puteți să nu specificați descrierea de job care conține USER(*RQD). Deoarece jobul autostart operează sub profilul utilizator care este specificat de descrierea de job, trebuie să controlați cine are voie să modifice descrierea de job.

Dacă este specificat mai mult de un job autostart pentru un subsistem, toate joburile autostart sunt pornite imediat, mai degrabă decât unul după celălalt. Dacă este depășit numărul maxim de joburi ale subsistemului nici un alt job nu poate fi pornit în subsistem până când nu sunt terminate destule joburi autostart astfel încât numărul de joburi care rulează să fie sub nivelul maxim de activitate.

Joburi batch:

Un job batch este un grup predefinit de acțiuni de procesare lansate sistemului pentru a fi realizate cu interacțiune redusă sau fără interacțiune între utilizator și sistem. Joburile care nu necesită interacțiunea utilizatorului pentru a rula pot fi procesate ca joburi batch. Un job batch de obicei este un job cu prioritate redusă și poate necesita pentru a rula un mediu de sistem special.

Joburile batch rulează în fundalul sistemului, eliberând utilizatorii care lansează jobul să facă alt lucru. Mai multe joburi batch pot fi active în același timp.

Lista următoare descrie tipurile diferite de joburi batch:

Job batch simplu

Jobul batch simplu este un job care se supune unei cozi de joburi. El așteaptă la rând cu celelalte joburi batch și este procesat în conformitate cu prioritatea sa și cu numărul de ordine.

Job batch imediat

Un job batch imediat este un job care începe cu multe din atributele jobului părinte. Jobul rulează în același subsistem ca și jobul părinte. (Aceasta se realizează utilizând API-ul spawn()). Deoarece jobul copie atributele de la jobul părinte și nu trece printr-o coadă de joburi, el poate porni mai repede decât un job care se supune unei cozi de joburi.

Job batch MRT

Un job batch MRT este un job terminal multiplu solicitant (MRT). Joburi MRT sunt mediu S/36 care acționează ca servere, permițând altor medii de job S/36 să atașeze la ele la comanda să ruleze o procedura MRT.

Tipărirea în batch a joburilor

Tipărirea în batch a joburilor urmărește fișierul de ieșire imprimantă (de asemenea fișierele de spool) care sunt create de un job al cărui profil al utilizatorului curent este diferit de profilul utilizatorului care a fost pornit dedesubt.

Joburile batch pot fi pornite când un utilizator:

- Face ca un job să fie pus într-o coadă de joburi
- Emite o cerere de pornire program de comunicație
- Porneste un subsistem cu un job prestart
- Utilizează API-ul spawn()

Cum porneste un job batch:

Când un utilizator lansează un job batch, jobul strânge informație de la mai multe obiecte de sistem înainte de a fi plasat într-o coadă de joburi.

1. Un utilizator lansează un job.
2. Jobul caută atribute de job. Dacă atributele jobului nu sunt găsite în comanda SBMJOB (Submit Job - Lansare job), jobul caută în descrierea jobului (specificat în comanda SBMJOB), profilul utilizator al utilizatorului curent, și jobul activ actual (jobul care lansează comanda SBMJOB).

Notă: Similar cu inițierea jobului interactiv, puteți specifica în descrierea jobului să se folosească profilul utilizator. Profilul utilizator poate specifica să se utilizeze o valoare de sistem pentru găsirea anumitor atribute de job.

3. O dată ce jobul are toate atributele sale, el se află în coada de joburi.
4. Când subsistemul este pregătit pentru a trata un job, el caută joburi în coada de joburi (cele pe care subsistemul le-a alocat).
5. Apoi, asemeni procesării de job interactiv, subsistemul verifică descrierea de job pentru datele de rutare.
6. Subsistemul utilizează datele de rutare pentru a găsi o intrare de rutare. Intrarea de rutare furnizează informații despre ce pool va folosi jobul, ce program de rutare va fi utilizat și din ce clasă își va obține jobul atributele de run-time.
7. După obținerea acestor informații, programul de rutare este rulat. Dacă utilizați QCMD, QCMD va executa comanda SBMJOB. El rulează comanda specificată în parametrul **CMD** sau **RQSDTA**.

Creare job batch:

Spawn este o funcție care creează un nou proces de job (proces copil) care moștenește multe atribute ale procesului de apelare (procesul părinte). Un program nou este specificat și porneste rularea în procesul copil. Când dați naștere la un job batch folosiți un job părinte pentru a transmite mai departe argumente și variabile de mediu către jobul copil. API-ul spawn() folosește joburi batch imediat, joburi prestart, sau joburi batch prestart.

Joburi de comunicație:

Un job de comunicație este un job batch care este pornit de o cerere de pornire a unui program, de la un sistem aflat la distanță. Procesarea jobului implică o cerere de comunicație și specificațiile corespunzătoare.

Pentru ca un job batch de comunicații să ruleze pe un sistem i5/OS, trebuie să existe pe sistem o descriere de subsistem conținând o intrare de lucru pentru joburi de comunicații. Intrarea de lucru pentru comunicații îi identifică subsistemului sursele pentru jobul de comunicații pe care acesta îl va procesa. Procesarea jobului începe când subsistemul recepționează o cerere de pornire a programului de comunicații de la un sistem aflat la distanță și se găsește o intrare de rutare corespunzătoare pentru cerere.

Rutarea datelor pentru joburi de comunicație

Rutarea joburilor pentru joburi de comunicații este determinată de cererea de pornire a programului care este recepționată de la sistemul aflat la distanță. Când este procesată o cerere de pornire a programului pe sistemul destinație este creat un flux de date de lungime fixă care este utilizat ca reprezentând date de rutare. Poziția 25 a datelor de rutare va conține întotdeauna PGMEVOKE pentru cereri de comunicații. Intrările de rutare ale subsistemului care specifică o

valoare de comparație a PGMEVOKE din poziția 29 au de obicei *RTGDTA ca nume de program. Aceasta înseamnă că numele de program specificat în datele de rutare (de la cererea de pornire a programului sistemului aflat la distanță) este programul de rulat.

Dacă este necesar un mediu special de procesare, pentru anumite joburi de comunicații, puteți adăuga o intrare de rutare suplimentară la descrierea subsistemului, specificând o valoare de comparație a cărei poziție de pornire este 37. Această valoare de comparație ar trebui să conțină numele de program pentru cererea de pornire a programului. Intrarea de rutare trebuie să aibă un număr de ordine mai mic decât intrarea de rutare care utilizează PGMEVOKE ca valoare de comparație. Această metodă permite anumitor joburi de comunicații să ruleze cu o clasă diferită sau cu o specificație de pool diferită.

Securitatea

Securitatea, pe sistemul i5/OS, controlează cine anume poate utiliza dispozitive de comunicație cât și cine poate accesa comenzile utilizate cu descrierile dispozitivului asociat. Ar trebui să luați în considerație măsuri suplimentare de securitate atunci când scrieți și rulați programe de aplicații atât pe sisteme aflate la distanță cât și pe sisteme destinație.

Descrierea de job pentru joburile de comunicație

Descrierea de job utilizată pentru joburi de comunicații este specificată în comanda (ADDCMNE) (Add Communications Entry - Adăugare intrare de comunicații). Utilizatorul specificat în această descriere de job este ignorat. Sistemul obține numele utilizatorului pentru joburi de comunicații din cererea de pornire a programului. Dacă cererea de pornire a programului nu specifică un nume de utilizator, sistemul utilizează valoarea implicită pentru utilizator din intrarea de comunicații. Pentru a asigura un grad mai înalt de securitate a sistemului, includeți informații ale utilizatorului despre cererea de pornire a programului în loc să specificați un utilizator implicit în intrarea de lucru a comunicațiilor.

Tipuri de joburi de comunicații:

Acest subiect descrie cele mai comune tipuri de joburi de comunicații.

Qlus (servicii de unitate logică)

Qlus tratează tratarea eveniment pentru dispozitivele de unități logice, cunoscute ca dispozitive de comunicații. Qlus este de asemenea responsabil pentru alocarea dispozitivelor subsistemului de comunicații corect.

Qcmnarbxx (arbitrii de comunicații)

Arbitrii de comunicații împreună cu Qsysarb (arbitrul de sistem) și Qtaparb (arbitrul de bandă) procesează munca pentru toate tipurile de dispozitive, nu doar pentru dispozitive de comunicații. Acest lucru include conexiunea de comunicații, deconectarea, blocarea dispozitivului și procesarea de recuperare eroare.

Valoarea de sistem a joburilor arbitru de comunicații, la repornire (QCMNARB) determină numărul de joburi arbitru de comunicații care sunt pornite. Un minim de 3 arbitri de comunicații sunt porniți pe sistemele cu un singur procesor.

Qsyscomm1 (comunicații de sistem)

Acest job tratează unele comunicații și activitate intrare/ieșire (I/O).

Q400filsvr (comunicație de sistem de fișier aflat la distanță)

Acest job execută comunicațiile interfeței de programare comune (APPN sau APPC) pentru sistemul de fișiere la distanță.

Joburi interactive:

Un job interactiv este un job care pornește atunci când un utilizator se semnează pe o stație de afișare și se sfârșește când utilizatorul renunță. Pentru ca jobul să ruleze, subsistemul caută descrierea jobului, care poate fi specificată în intrarea stației de lucru sau în profilul utilizator.

Joburile interactive necesită comunicații continue pe două căi între utilizator și iSeries pentru a realiza o operație. Un job interactiv pornește atunci când un utilizator se semnează pe un sistem. Sistemul cere informații de semnare. Dacă cererea de semnare este acceptată de sistem, atunci sistemul creează jobul interactiv. Sistemul atunci cere utilizatorului să facă o cerere. Utilizatorul introduce o cerere și sistemul răspunde procesând cererea. Acest model este repetat până când utilizatorul oprește jobul interactiv ieșind din sistem, sau jobul se sfârșește din cauza unei excepții de aplicație sau a unei recuperări de eroare de dispozitiv.

Dacă un job interactiv este parte a unui grup de joburi sau o pereche de joburi, atunci el va avea unul din următoarele tipuri de joburi:

Interactiv - Grup

Un job Interactiv - Grup este parte a unui grup de joburi care este asociat cu un singur dispozitiv de afișare.

Interactiv - Cerere de sistem

Un job Interactiv - Cerere de sistem este unul dintr-o pereche de joburi care sunt asociate între ele de funcția de cerere sistem.

Știați că? Vă puteți semna pe sistem în două moduri. Puteți intra manual în sistem utilizând un ID utilizator și o parolă. Puteți, de asemenea, crea un program care să trimită automat ID-ul utilizator și parola la server, în felul acesta ocolind ecranul de semnare.

Cum pornește un job interactiv:

Când un utilizator se înregistrează pe sistem, subsistemul adună informații de la mai multe obiecte de sistem înainte ca jobul interactiv să fie pregătit.

1. Subsistemul caută în intrarea de stație de lucru descrierea jobului pentru a obține atribute pentru jobul interactiv. Dacă intrarea de stație de lucru specifică *USRPRF pentru descrierea jobului, jobul va utiliza informațiile din profilul utilizator.

Notă: Această flexibilitate vă permite să specificați dacă atributele jobului sunt legate de stația de lucru sau de utilizatorul individual.

2. După ce subsistemul știe ce descriere de job să utilizeze, este posibil să nu găsească toate atributele jobului în descrierea de job. Unele atribute este posibil să fie în profilul utilizator. Dacă profilul utilizator nu are informațiile, subsistemul caută în valoarea de sistem.

Notă: Profilul utilizator conține atribute de job care vă permit să croiți anumite lucruri special pentru utilizator.

3. După ce subsistemul adună toate atributele jobului, el determină dacă poate porni un nou job interactiv sau dacă ar trebui afișat un mesaj de eroare pe ecranul de semnare. Subsistemul verifică dacă s-a ajuns la numărul maxim de joburi permise de către subsistem sau de către intrarea de stație de lucru. Apoi verifică dacă a fost livrat un nume de profil utilizator valid, dacă numele de profil utilizator este al unui profil utilizator activat și dacă parola livrată (dacă este necesară) este validă. Apoi, verifică dacă utilizatorul are autorizările corecte pentru descrierea jobului, descrierea subsistemului, descrierea dispozitivului stație de lucru și pentru coada de ieșire și bibliotecă. În final, subsistemul verifică dacă utilizatorul a ajuns la limita de semnări permise pentru acel profil utilizator. Dacă este întâlnită vreo eroare de validare, ecranul de semnare afișează mesajul corespunzător. Altfel, procesul de pornire a jobului interactiv continuă.
4. După ce subsistemul validează faptul că poate porni jobul interactiv, el verifică descrierea jobului pentru datele de rutare. Subsistemul utilizează datele de rutare pentru a găsi o intrare de rutare din descrierea subsistemului. Intrarea de rutare furnizează informații despre ce pool va utiliza jobul, ce program de rutare va fi folosit și din ce clasă își va lua jobul atributele de runtime.
5. Când sunt obținute toate aceste piese, programul de rutare rulează. IBM livrează un program de rutare numit QCMD, pe care îl puteți utiliza pentru toate tipurile de lucru. QCMD știe dacă jobul este un job interactiv și verifică profilul utilizator pentru ca un program inițial să ruleze. Dacă programul inițial termină de rulat, QCMD afișează meniul inițial.

Deconectarea joburilor interactive:

Când este apelată comanda DSCJOB (Disconnect Job - Deconectare job), jobul este deconectat și ecranul de semnare apare din nou. Pentru a vă conecta din nou la job, semnați-vă pe același dispozitiv de la care v-ați deconectat. Poate fi pornit alt job interactiv pe dispozitiv, sub alt nume de utilizator.

- O opțiune din meniul Cerere sistem vă permite să deconectați un job interactiv, făcând astfel să apară ecranul de semnare. Opțiunea apelează comanda DSCJOB.
- Când vă conectați din nou la un job, valorile specificate în ecranul de semnare pentru program, meniu și biblioteca curentă, sunt ignorate.
- Un job care are activă o funcție de organizare PC sau de asistare text PC nu poate fi deconectat.
- Un job TCP/IP TELNET poate fi deconectat dacă sesiunea utilizează o descriere de dispozitiv denumită specificată de utilizator. Puteți crea o descriere de dispozitiv denumită specificată de utilizator utilizând una din următoarele căi:
 - Utilizând Stații rețea cu parametrul **DISPLAY NAME**
 - Utilizând suportul Client Access PC 5250 din iSeries cu funcția de ID a stației de lucru
 - Utilizând punctul de ieșire TCP/IP TELNET Inițializare dispozitiv pentru a specifica un nume de stație de lucru
 - Client Telnet (STRTCPTLN) cu parametrul dispozitivului de la distanță

Notă: Numele de dispozitiv specificate de sistem, precum QPADEV*, nu permit ca jobul să fie deconectat deoarece este puțin probabil că același utilizator se va semna pe același dispozitiv.

- Vor fi deconectate toate joburile pentru joburile de grup. Când acestea sunt reconectate, vă întoarceți în locul unde a fost emisă deconectarea. Dacă ultimul job de grup activ se termină înainte să vă reconectați, vă întoarceți la următorul job de grup.
- Dacă jobul nu poate fi deconectat din orice motiv, acesta va fi oprit în loc să fie deconectat.
- Toate joburile deconectate din subsistem se opresc atunci când se oprește subsistemul. Dacă un subsistem se oprește, comanda DSCJOB nu poate fi lansată în oricare din joburile din subsistem.
- Valoarea de sistem QDSCJOBITV (Disconnect Job Interval - Interval deconectare job) poate fi utilizată pentru a indica un interval de timp pentru care un job poate fi deconectat. Dacă se termină intervalul de timp, jobul deconectat se oprește.
- Joburile deconectate care nu au depășit valoarea QDSCJOBITV se opresc atunci când este oprit subsistemul sau când survine un IPL.

Eroarea de I/E pentru dispozitivul solicitant de joburi:

Un dispozitiv solicitant este o stație de lucru de la care un utilizator se poate înregistra pe un domeniu și poate utiliza resursele rețelei. Atributul de job DEVRCYACN (Device Recovery Action - Acțiune de recuperare dispozitiv) specifică ce acțiuni trebuie să se întreprindă atunci când are loc o eroare de I/E pentru un dispozitiv solicitant al jobului.

Atributul DEVRCYACN are următoarele opțiuni:

***SYSVAL**

Aceasta este implicită. Se referă la valoarea de sistem QDEVRCYACN, acțiune de întreprins când are loc o eroare de dispozitiv stație de lucru. Valoarea de sistem va suporta toate valorile pe care le suportă atributul de job (cu excepția ***SYSVAL**).

***MSG** Semnalizează mesajul de eroare I/E și lasă programul de aplicație să realizeze recuperarea erorii. Aceasta Nu este setarea recomandată.

***DSCMSG**

Deconectarea jobului. Aceasta este soluția implicită livrată. După reconectare, un nou mesaj de eroare semnalează programul de aplicație a utilizatorului indicând că dispozitivul a fost pierdut și recuperat de la I/E, și conținuturile ecranului trebuie reafișate.

***DSCENDRQS**

Deconectarea jobului. După reconectare se realizează o funcție de cerere de oprire pentru a returna controlul jobului la ultimul nivel de cerere.

***ENDJOB**

Oprire job. Este posibil să fie produs un istoric de job pentru acest job. Este trimis un mesaj către istoricul jobului și către istoricul QHST indicând faptul că jobul s-a oprit din cauza erorii de dispozitiv.

***ENDJOBNO LIST**

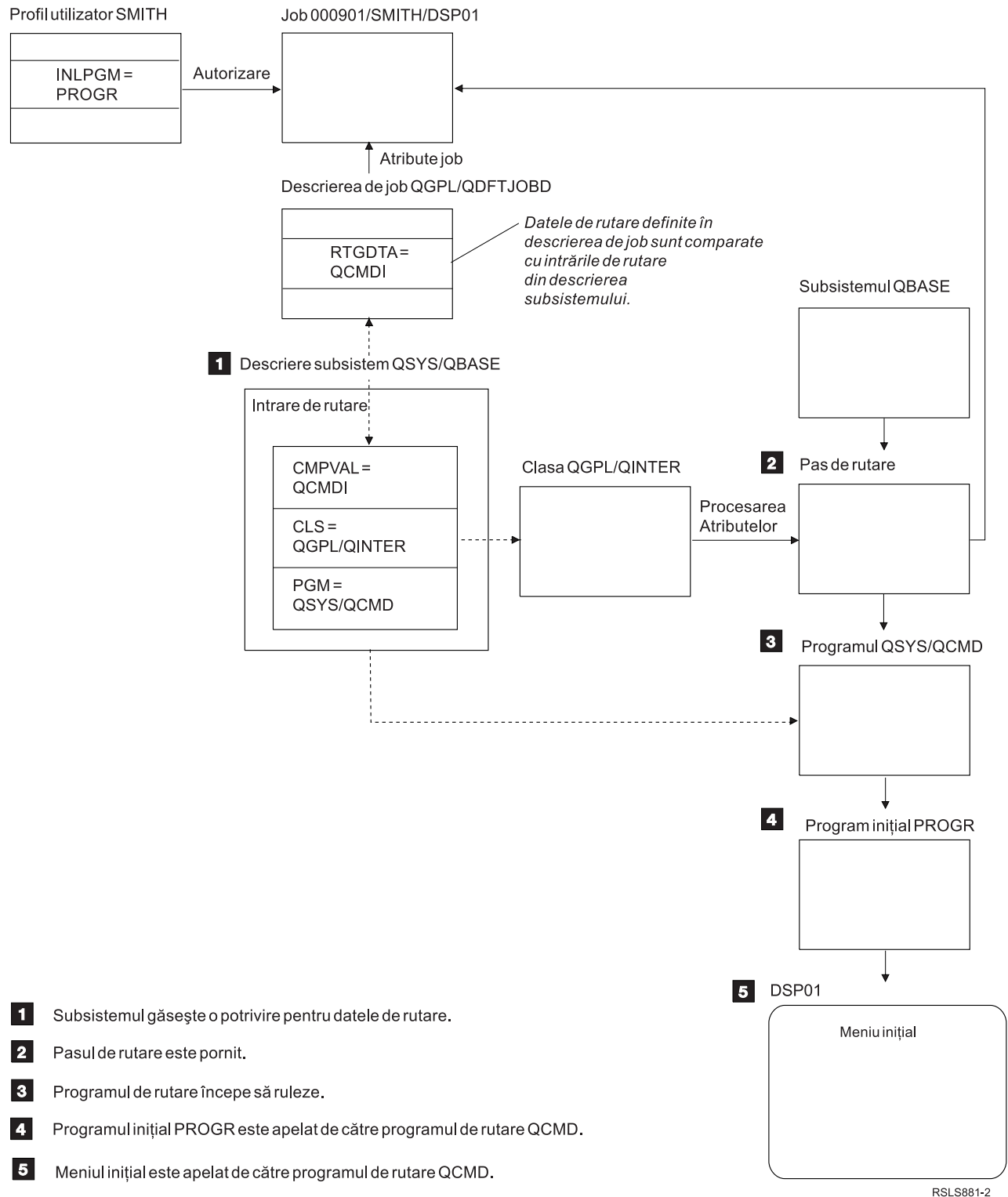
Oprire job. Nu se produce nici un istoric de job. Este trimis un mesaj către istoricul QHST indicând faptul că jobul s-a terminat din cauza erorii de dispozitiv.

Notă: Dacă se specifică ***DSCENDRQS**, ***ENDJOB**, sau ***ENDJOBNO LIST** pentru DEVRCYACN, acțiunea de recuperare intră în vigoare atunci când are loc eroarea în dispozitiv. Dacă este specificată una din celelalte valori, acțiunea de recuperare are loc la următoarea I/O asupra dispozitivului cu eroarea.

Joburi interactive și pași de rutare:

Înainte să fie apelat meniul inițial datele de rutare sunt comparate cu intrările de rutare din descrierea subsistemului. Când se face o potrivire, programul specificat în intrarea de rutare este apelat și pasul de rutare este pornit.

Următoarele ilustrează activitatea următoare care conduce la pornirea unui pas de rutare și afișarea meniului inițial pentru un profil utilizator care specifică un program inițial.



- 1** Subsistemul găsește o potrivire pentru datele de rutare.
- 2** Pasul de rutare este pornit.
- 3** Programul de rutare începe să ruleze.
- 4** Programul inițial PROGR este apelat de către programul de rutare QCMD.
- 5** Meniul inițial este apelat de către programul de rutare QCMD.

RSL881-2

Figura 1. Activitate de subsistem

Abordări ale jobului interactiv

Puteți trata un job interactiv în diferite moduri. Aceste abordări sunt dependente de cum controlați pasul de rutare. Mai întâi ar trebui să determinați următoarele:

- Ce program va controla pasul de rutare: QSYS/QCMD sau un program de utilizator?

- Rutarea va fi bazată pe utilizator sau pe stația de lucru?

Programe care controlează pasul de rutare:

Pentru a determina cea mai bună abordare pentru un anumit job, trebuie mai întâi să determinați ce program ar trebui să controleze pasul de rutare.

Utilizarea QSYS/CMD pentru joburile interactive - avantaje

Procesorul de comenzi livrat de IBM QSYS/QCMD oferă cea mai mare flexibilitate în ce privește punerea la dispoziție a funcțiilor pentru utilizatorii de stații de lucru. Utilizarea QCMD pentru controlul pasului de rutare vă oferă următoarele avantaje:

- Programul de atenție este activat dacă el este specificat în profilul utilizator.
- Este apelat programul inițial care este specificat în profilul utilizator.
- Este apelat meniul inițial care este specificat în profilul utilizator.
- Utilizatorul este pus în mediul System/36 după cum este specificat în profilul utilizator.

În plus, utilizarea implicită a QCMD vă duce în Meniul principal unde puteți introduce comenzi direct, puteți include comanda CALL, care este utilizată pentru apelarea funcțiilor scrise de utilizator. Sunt furnizate funcții de meniu cu ajutor online pentru a oferi un acces mai ușor la funcțiile sistemului. Mai sunt furnizate meniuri de selectare a comenzii, acces rapid la căutarea indexului și funcția de introducere comandă (apelată de CALL QCMD). Funcțiile de introducere comandă sunt destinate în primul rând programatorilor și operatorilor care necesită disponibilitatea întregii game de funcții prin utilizarea directă a comenzilor.

Apelarea unui program utilizator direct pentru joburi interactive - avantaje

Programele dumneavoastră pot fi apelate direct pentru a controla pașii de rutare pentru joburi interactive. Aceste programe pot fi proiectate pentru a oferi un acces mai specializat la funcțiile necesitate de utilizatorii stațiilor dumneavoastră de lucru decât oferă programele livrate de IBM. În plus, deoarece programele dumneavoastră sunt croite pentru anumite funcții, ele ar trebui să necesite chiar mai puține resurse de sistem pentru a le suporta rularea decât programele livrate de IBM. Trebuie, de asemenea, să doriți să furnizați funcții precum un program inițial sau un meniu inițial.

Stațiile de lucru față de rutarea bazată pe utilizator:

După ce ați determinat ce program controlează pasul de rutare, trebuie să determinați dacă rutarea se va baza pe stația de lucru pe care este pornit jobul sau pe utilizatorul (profilul utilizator) care s-a înregistrat.

Rutarea bazată pe stația de lucru este realizată utilizând datele de rutare specificate în descrierea jobului asociată cu intrarea stației de lucru sau cu profilul dispozitivului. Rutarea bazată pe utilizator se poate face utilizând programul inițial specificat în profilul utilizator sau descrierea de job din profilul utilizator mapând către altă intrare de rutare decât QCMD.

Utilizări program inițial

Programul inițial poate interacționa cu stații de lucru pentru a obține valori de intrare de la un utilizator de stație de lucru. Când este apelat un program inițial, el nu poate primi valori de parametru. Un program inițial poate fi utilizat în unul din cele două moduri:

- Pentru a stabili un mediu inițial pentru utilizatorul care introduce comenzile. De exemplu, lista de biblioteci poate fi modificată sau pot fi suprascrise fișiere de imprimantă sau fișiere de mesaj. Când un program inițial își efectuează funcțiile și se întoarce la QSYS/QCMD, este afișat meniul inițial.
- Ca program de control pentru job. Dacă programul inițial nu se întoarce la QSYS/QCMD, el devine program de control pentru pasul de rutare. Meniul inițial nu este afișat. Utilizatorul poate cere doar acele funcții disponibile prin programul inițial.

De exemplu, poate fi afișat un meniu cu opțiuni pentru o anumită aplicație. Utilizatorul poate realiza doar funcțiile din meniu. Un exemplu de astfel de opțiune este anularea semnăturii (sign off). Dacă este rulată comanda SIGNOFF, jobul se termină și Meniul principal al sistemului nu este afișat niciodată. Dacă utilizați această abordare, luați în considerare utilizarea opțiunii de profil utilizator INLMNU pentru a vă asigura că nu este afișat nici un meniu.

Poate fi scris un program inițial astfel încât atunci când este emisă o operație de returnare, se întoarce sau nu la QSYS/QCMD. Dacă programul inițial se întoarce la QSYS/QCMD, este afișat meniul inițial.

Când joburile se finalizează în același timp:

Câteodată, joburile se finalizează în același timp. De exemplu, o eroare de rețea apare și atributele jobului sunt setate la *ENDJOB sau *ENDJOBNO LIST. În plus față de finalizarea jobului, următoarele acțiuni de recuperare dispozitiv apar.

- Prioritatea jobului este scăzută. Aceasta apare astfel ca jobul să nu mai fie la aceeași prioritate ca alte joburi interactive active.
- Felia de timp a jobului este setată la 100 milisecunde. Aceasta apare ca să dea joburilor cu prioritate mai ridicată o șansă mai bună de a obține resurse de procesare.

Înregistrări de job pentru joburi cu atribute de job setate la *ENDJOB sau *ENDJOBNO LIST sunt în înregistrare job în așteptare. Pentru a produce ieșire imprimantă dintr-o înregistrare de job care este în înregistrare job în așteptare, folosiți comanda Afișare înregistrare job (DSPJOBLOG).

Când un job ia sfârșit puteți controla cum este scrisă înregistrarea jobului la un fișier de spool. Aceasta poate fi făcută de către jobul însuși în timp ce se finalizează, de către un job server de fundal, sau deloc. Valoarea pe care ați specificat-o poate avea un impact semnificativ pe tot timpul de recuperare când multe joburi se finalizează în același timp. Pentru informații suplimentare, vedeți conceptul înrudit Înregistrare job în așteptare.

Joburi prestart:

Un job prestart este un job batch care începe să ruleze înainte de a fi recepționată o cerere de lucru. Joburile prestart sunt pornite înainte de celelalte tipuri de joburi dintr-un subsistem. Joburile prestart sunt diferite de celelalte joburi deoarece ele utilizează intrări de job prestart (parte din descrierea subsistemului) pentru a determina ce program, clasă și pool de spațiu de stocare să fie utilizate când ele sunt pornite.

Într-o intrare de job prestart trebuie să specificați atributele pe care subsistemul le utilizează pentru a crea și gestiona un pool de joburi prestart. Utilizați joburi prestart pentru a reduce durata necesară pentru a trata o cerere de lucru. Există 2 tipuri de joburi prestart. Fiecare tip tratează tipuri diferite de cereri. Înainte ca un job să aștepte prima sa cerere, el va fi afișat ca Doar prestart pentru că sistemul nu știe încă ce tip de cereri va trata jobul.

Comunicații prestart

Jobul este un job batch de comunicații care începe să ruleze înainte ca un sistem aflat la distanță să trimită o cerere de pornire program.

Batch prestart

Jobul este un job batch care pornește înainte de a fi recepționată o cerere de lucru.

Un job prestart pornește înainte de primirea unei cereri de lucru, fie când pornește subsistemul fie ca rezultat al comenzii Pornire joburi prestart (STRPJ). Joburile prestart pornesc dintr-o intrare de job prestart (PJE) din descrierea de subsistem. Intrarea de job prestart specifică atribute precum ce program să ruleze în jobul prestart, profilul utilizator sub care jobul prestart începe să ruleze, descrierea jobului, clasa utilizată pentru a specifica atributele de run-time (timp de rulare) ale jobului și pool-ul de memorie în care rulează jobul prestart.

Joburile prestart pot porni și se pot inițializa singure înainte de primirea unei cereri de lucru. Aceasta reduce cantitatea de timp necesar pentru a trata cererile. Joburile prestart furnizează abilitatea de a inițializa o singură dată și de a trata mai multe cereri astfel încât să nu fie necesar un job nou pentru fiecare cerere. Multe aplicații server client utilizează joburi prestart pentru a trata cererile pentru utilizatorul client. A avea un job gata de rulare înseamnă o performanță mai bună pentru că jobul prestart poate porni procesarea cererii pentru utilizator imediat.

Notă: Valoarea specificată pentru numărul maxim de joburi din subsistem poate împiedica joburile prestart să pornească. Dacă numărul maxim de joburi din subsistem este depășit, nici un job prestart nu mai poate fi pornit. Când destule joburi au fost terminate astfel că numărul de joburi care rulează este sub numărul maxim de joburi din subsistem, joburile prestart din subsistem pot porni.

Cereri pornire program

O Cerere de pornire program (PSR) este o cale planificată pentru ca clienții SNA să se conecteze la un server SNA. Când un job prestart este setat pentru a trata PSR-uri, starea externă a jobului este în PSRW (Program Start Request Wait - Așteptare cerere pornire program).

Joburile prestart sunt, de asemenea, utilizate pentru servere TCP/IP livrate de IBM, cel mai remarcabil pentru serverele gazdă. Aceste joburi prestart acceptă de lucru prin interfețele interne și PSR-urile nu sunt utilizate. Totuși, joburile prestart care așteaptă de lucru, chiar dacă nu utilizează PSR-uri, afișează totuși o stare PSRW.

Nume job prestart:

Numele complet calificat, alcătuit din trei părți, al jobului prestart nu se mai modifică o dată ce jobul prestart este pornit. Numele de utilizator al numelui complet calificat, alcătuit din trei părți, conține întotdeauna profilul utilizator sub care este pornit jobul prestart.

Dacă este deschis un fișier spool înainte ca un job prestart să trateze vreo cerere de lucru, fișierul spool este asociat cu profilul utilizator al intrării jobului prestart. Altfel, el este asociat cu profilul utilizator curent al jobului.

Dacă profilul intrării jobului prestart și profilul utilizatorului curent sunt diferite, fișierele puse în spool sunt puse în spool sub un job cu numele de job QPRTJOB și numele de utilizator al profilului utilizator curent. (Aceasta este, de asemenea, adevărată pentru intrări de job prestart pentru joburi de server.)

Parametrul de clasă (**CLS**) din intrarea jobului prestart furnizează o cale de a controla caracteristicile de performanță a două clase ale joburilor prestart per intrare de job prestart.

Cum lucrează joburile prestart:

Un job prestart este un job care este pornit înainte să existe de lucru. Aceasta îi permite sistemului să manipuleze o cerere de lucru fără întârzierea cauzată de pornirea unui nou job.

Un job prestart este un tip unic de job batch. Aceasta înseamnă că jobul este de tipul 'B' și are un subtip de job 'J'. Tipul de job, îmbunătățit, definește mai departe jobul ca pe un job prestart (1610), job batch prestart (1620) sau job de comunicații prestart (1630). Tipul de job îmbunătățit descrie modul cum jobul prestart acceptă cereri de lucru. Dacă programul care rulează în jobul prestart utilizează interfața de comunicații pentru a accepta de lucru, jobul este un job de comunicații prestart. Dacă programul care rulează în jobul prestart accepta de lucru printr-o interfață de lucru batch, jobul este un job batch prestart. Dacă programul nu a ajuns încă în punctul de a accepta de lucru, jobul este doar un job prestart. Joburile batch prestart sunt des numite joburi de server deoarece ele furnizează service pentru cererile de lucru.

O cerere de lucru de comunicații este manipulată de subsistemul care alocat dispozitivul de comunicații necesar. O cerere de lucru batch este de obicei manipulată de unul din subsistemele de bază care sunt livrate cu sistemul: QSYSWRK, QUSRWRK sau QSERVER.

Joburile prestart sunt pornite pe baza informațiilor conținute în intrările jobului prestart. Parametrul Start joburi (**STRJOBS**) poate specifica faptul că joburile prestart sunt pornite atunci când subsistemul este pornit sau când este introdusă comanda (STRPJ) (Start Prestart Jobs - Pornire joburi prestart). Parametrul Număr inițial de joburi (**INLJOBS**) determină numărul de joburi prestart care pornesc inițial pentru un program.

Pe măsură ce sosesc cereri de lucru, este posibil să fie necesare mai multe joburi prestart. Parametrul prag (**THRESHOLD**) spune când să fie pornite mai multe joburi. Când numărul de joburi prestart disponibile pentru a trata

o cerere scade sub valoarea specificată de parametrul prag **THRESHOLD**, sunt pornite joburile suplimentare. Parametrul Număr de joburi suplimentare (**ADLJOBS**) spune câte joburi să mai fie pornite.

Trebuie să vă asigurați că există destule joburi pentru a trata cererile de lucru fără întârzieri necesare. Puteți utiliza informațiile oferite de comanda (DSPACTPJ) (Display Active Prestart Jobs - Afișare joburi prestart active) pentru a vedea dacă vreo cerere de lucru numită "cerere de pornire program", din informația DSPACTPJ, a trebuit să aștepte ca joburile să devină disponibile. Creșteți valoarea parametrului **THRESHOLD** până când DSPACTPJ arată că nici o cerere de lucru nu va mai fi niciodată nevoită să aștepte.

Unele joburi prestart tratează o cerere de lucru și apoi devin disponibile să trateze altă cerere de lucru. Parametrul Număr maxim de utilizări (**MAXUSE**) vă permite să specificați câte cereri de lucru tratează aceste joburi prestart. Unele joburi prestart tratează o singură cerere de lucru și apoi se termină, ignorând valoarea **MAXUSE**. Programul care rulează în jobul prestart decide dacă jobul prestart tratează mai multe cereri de lucru sau tratează doar o cerere de lucru.

Când jobul prestart se termină după ce a tratat cel puțin o cerere de lucru, subsistemul compară numărul de joburi care încă mai rulează cu numărul specificat în parametrul **INLJOBS**. Dacă numărul de joburi care rămâne este mai mic decât **INLJOBS**, subsistemul pornește alt job.

Dacă un job prestart se termină fără să fi tratat măcar o cerere de lucru și jobul nu a fost terminat de comanda (ENDJOB) (End Job - Terminare job), programul jobului prestart este considerat a fi eronat. Subsistemul termină intrarea de job prestart într-o manieră controlată. Aceasta permite joburilor care servesc o cerere de lucru să termine acea cerere, dar împiedică pornirea joburilor suplimentare de către subsistem.

Subsistemul verifică periodic numărul de joburi prestart pentru a determina dacă există joburi prestart disponibile în exces. Un job prestart este disponibil atunci când așteaptă o cerere de lucru. Dacă numărul de joburi prestart disponibile este mai mare decât parametrul **THRESHOLD** și numărul total de joburi prestart este mai mare decât parametrul **INLJOBS**, subsistemul oprește unele din joburile prestart disponibile. Pentru ca joburile prestart să trateze mai mult de o cerere de lucru, ar trebui să utilizați informația DSPACTPJ pentru a vă determina sarcina vârf, să adăugați acest număr la valoarea parametrului **THRESHOLD** și să setați parametrul **INLJOBS** la această sumă. Aceasta împiedică subsistemul să considere că joburile disponibile sunt în exces doar dacă volumul de lucru nu este mai mare decât sarcina vârf așteptată.

Intrări de job prestart:

Definiți jobul prestart utilizând o intrare de job prestart. O intrare de job prestart nu afectează alocarea dispozitivelor sau alocarea cererii de pornire program.

Atributele de job ale unui job prestart nu sunt modificate de către subsistem când o cerere de pornire program se atașează la un job prestart. Totuși, joburile de server, în general, modifică atributele de job cu cele ale profilului utilizator cu care s-a făcut schimb (swap).

Comanda CHGPJ (Change Prestart Job - Modificare job prestart) permite jobului prestart să modifice unele din atributele de job cu cele ale descrierii de job (specificate în descrierea de job asociată cu profilul utilizator al cererii de pornire program sau în descrierea de job specificată în intrarea jobului prestart).

Securitatea și joburile prestart:

Când un job prestart pornește, el rulează sub profilul utilizator al jobului prestart. Când o cerere de pornire program se atașează la un job prestart, profilul utilizator al jobului prestart este înlocuit de profilul utilizator al cererii de pornire program. Când jobul prestart a terminat tratarea unei cereri de pornire program, profilul utilizator al cererii de pornire program este înlocuit de profilul utilizator al jobului prestart. Dacă există un profil de grup asociat cu profilul utilizator, profilul de grup este, de asemenea, modificat.

Schimbul profilului utilizator este doar pentru verificarea autorizării. Nici unul din celelalte atribute asociate cu profilul utilizator nu este modificat. Bibliotecile din lista de biblioteci la care este autorizat profilul utilizator al intrării jobului

prestare continuă să fie autorizate la jobul prestare când profilul utilizator al cererii de pornire program înlocuiește profilul utilizator al intrării de job prestare. Totuși, lista de biblioteci poate fi modificată de comanda CHGLIBL (Change Library List - Modificare listă de biblioteci).

Autorizare obiect job prestare

Când un job prestare pornește, este realizată verificarea autorizării împotriva profilului utilizator al intrării jobului prestare, pe fiecare obiect care este necesar pentru pornirea unui job. Înainte ca o cerere de pornire program să aibă permisiunea de a se atașa la un job prestare, sunt verificate doar profilul utilizator/parola cererii de pornire program și autorizarea sa la dispozitive de comunicații și bibliotecă/program.

Pentru a evita evenimentele în care profilul utilizator al cererii de pornire program nu este autorizat la obiecte la care este autorizat profilul utilizator al intrării jobului prestare, trebuie să vă asigurați că profilul utilizator al cererii de pornire program este autorizat la cel puțin atâtea obiecte cât și profilul utilizator al intrării jobului prestare. Pentru a realiza aceasta, programul jobului prestare poate fi creat de utilizatorul intrării jobului prestare cu `USRPRF(*OWNER)` specificat în comanda `CRTxxxPGM` (unde xxx este limbajul programului). Autorizarea proprietarului programului va fi transferată automat oricărui program apelat de programul jobului prestare. Altfel, puteți alege să verificați în mod explicit autorizarea obiectului (`CHKOBJ`) înainte de a face referire la orice obiect.

Fișierele și obiectele la care profilul utilizator al jobului prestare nu este autorizat ar trebui să fie închise și dealocate înainte ca sfârșitul tranzacției să fie realizat pe dispozitivul solicitant. Dacă fișierele de bază de date sunt lăsate deschise în jobul prestare, pentru a garanta securitatea bazei de date, programul jobului prestare trebuie să verifice autorizarea la fișierele deschise a profilului utilizator al cererii de pornire program.

Sugestii de performanță pentru joburi prestare:

Jobul prestare ar trebui să funcționeze cât mai mult înainte de a încerca să obțină un dispozitiv cu program ICF sau să accepte o conversație de Comunicații CPI. Cu cât mai mult lucrează inițial (alocă obiecte, deschide fișiere de bază de date, și așa mai departe), cu atât mai puțin va avea de făcut când este recepționată o cerere de pornire program, dând, prin urmare, un timp de răspuns mai rapid tranzacției. Următoarele sunt câteva considerente suplimentare de performanță când se utilizează joburile prestare:

De reținut: Dacă o intrare de job prestare este în subsistem, subsistemul verifică periodic numărul de joburi prestare dintr-un pool care sunt pregătite să servească cereri de pornire program pentru a determina dacă sunt prea multe joburi prestare disponibile. Joburile prestare disponibile care sunt în exces sunt oprite de subsistem treptat. Totuși, subsistemul lasă întotdeauna într-un pool cel puțin numărul de joburi prestare specificat în atributul `INLJOBS`.

- Ar trebui doar să dealocați resursele specifice tranzacției care doriți să fie realizată. Toate resursele care sunt în mod obișnuit utilizate pentru alte tranzacții realizate de programul jobului prestare ar trebui să rămână alocate în timp ce jobul își așteaptă următoarea cerere. Ar trebui să lăsați fișiere deschise și obiecte alocate pentru a economisi timp când este recepționată următoarea cerere.

Notă: Fișierele de bază de date care sunt lăsate deschise în jobul prestare necesită, de obicei, aceleași considerente ca și fișierele bază de date care sunt partajate în același job.

- Din moment ce aceeași bibliotecă `QTEMP` este utilizată pentru întreaga viață a unui job prestare, obiectele care nu mai sunt necesare ar trebui șterse.
- Din moment ce aceeași zonă de date locală (`LDA`) este utilizată pentru întreaga viață a unui job prestare, informațiile pot fi ținute și transmise următoarei tranzacții.
- Din moment ce fiecare job prestare poate trata mai multe cereri de pornire program și are doar un istoric de job, este posibil să doriți ca aplicația dumneavoastră să trimită mesaje istoricului de job, identificând activitatea jobului prestare. Aceasta este, de asemenea, utilă deoarece istoricele joburilor batch prestare sunt curățate între utilizări.
- Atributele de job ale unui job prestare nu sunt modificate de către subsistem când se atașează o cerere de pornire program la un job prestare. Comanda `CHGPJ` (Change Prestare Job - Modificare job prestare) permite jobului prestare să modifice unele din atributele jobului cu cele din descrierea jobului (specificată în descrierea de job asociată cu profilul utilizator al cererii de pornire program sau în descrierea de job specificată în intrarea jobului prestare.)

- Dacă sistemul dumneavoastră utilizează contabilizarea joburilor, programul jobului prestart ar trebui să ruleze comanda CHGPJ cu valoarea cererii de pornire program pentru parametrul codului de contabilizare (CHGPJ ACGCDE(*PGMSTRRQS)) imediat după ce cererea de pornire program este atașată la jobul prestart. Această acțiune modifică codul de contabilizare cu valoarea specificată în profilul utilizator asociat cu cererea de pornire program. Imediat înainte ca programul să termine de tratat cererea de pornire program, programul ar trebui să ruleze comanda CHGPJ cu valoarea Prestart Job Entry - Intrare job prestart pentru parametrul codului de contabilizare (CHGPJ ACGCDE(*PJE)). Aceasta modifică codul de contabilizare înapoi cu valoarea intrării jobului prestart specificată în descrierea jobului.
- Parametrul clasă (CLS) din intrarea jobului prestart furnizează o cale de control al caracteristicilor de performanță a două clase de joburi prestart per intrare de job prestart. De exemplu, puteți furniza o prioritate de execuție mai mică pentru lucrul care sosește atunci când sistemul este deja ocupat.

Fișier pus în spool și intrarea jobului prestart:

Dacă un fișier spool este deschis înainte ca un job prestart să trateze toate cererile de pornire program, fișierul spool este asociat cu profilul utilizator al intrării i jobului prestart; altfel el este asociat cu profilul utilizator al cererii de pornire program curente.

Dacă profilul intrării jobului prestart și profilul utilizator al cererii de pornire program curente sunt diferite, fișierele spooled sunt puse în spool sub un job a cărui primă parte, din cele trei părți ale numelui de job, este QPRTJOB și a doua parte este numele profilului utilizator.

Joburile de citire și scriere:

Un *job cititor* este un job cu intrarea pusă în spool, și un *job scriitor* este un job cu ieșirea pusă în spool.

Cititor Un job cititor citește fluxuri de joburi batch din fișiere de bază de date și pune joburile într-o coadă de joburi. Jobul de citire este parte din introducerea în spool a intrării și este un program furnizat de IBM.

Scriitor

Un job de scriere scrie înregistrările din fișierele imprimantă (de asemenea numite fișiere spool) la o imprimantă. Jobul de scriere este un program furnizat de IBM care este pornit în subsistemul de spool unde selectează fișiere din coada de ieșire care trebuie să fie tipărite.

Joburi server:

Joburile de server sunt joburile care rulează continuu în fundal, în sistemul iSeries.

Lucrul poate apărea din funcțiile de rețea, funcțiile sistemului de operare, în numele unui utilizator, a altui sistem de pe rețea sau din serviciile sistem generale, precum joburile server de funcționare în cluster. Joburile de server rulează de obicei în unul din cele trei subsisteme de bază livrate cu sistemul - QSYSWRK, QSERVER, sau QUSRWRK. Joburile de server sunt de cele mai multe ori asociate cu funcții precum HTTP, Note Lotus și TCP/IP. Sistemul iSeries are trei modele de bază pentru joburile de server:

Modelul de job cu fir de execuție

În modelul de job cu fir de execuție jobul de server este un job cu mai multe fire de execuție. Un fir de execuție se comportă ca un distribuitor de lucru pentru celelalte fire de execuție. De exemplu, când serverul primește o cerere client, firul de execuție inițial citește cererea și o pasează unui alt fir de execuție pentru a îndeplini cererea. Cu acest model, cantitatea de joburi din sistem este foarte redusă pentru că lucrul este tratat în diferite fire de execuție în loc să necesite mai multe joburi. Câteva exemple de joburi de server care utilizează modelul de job cu fir de execuție sunt Domino, serverul HTTP și WebSphere.

Model de job prestart

În modelul jobului prestart există de obicei un job primar care se comportă ca un ascultător al cererilor care vin în sistem. Acest job este numit de obicei jobul demon. Jobul demon tratează cererea inițială și apoi o pasează jobului de server prestart corespunzător. Cu acest model de job, utilizarea joburilor prestart poate reduce numărul de joburi care sunt necesare deoarece după ce o cerere a fost îndeplinită jobul de server prestart așteaptă următoarea cerere. Jobul server este refolosit. De asemenea, din perspectiva performanței,

jobul prestart deja rulează și așteaptă să proceseze cererea. Unele exemple de joburi de server care utilizează modelul de job prestart sunt serverul SQL, serverele gazdă, și serverul SMTP (Simple Mail Transfer Protocol).

Notă: Pentru joburile care rulează cod utilizator, de obicei jobul nu este refolosit (ca pentru majoritatea joburilor server). Aceasta deoarece codul utilizatorului poate să fi schimbat ceva în job (cum ar fi serverul de comandă de la distanță).

Modelul de job cu ascultare multiplă

În modelul jobului cu ascultare multiplă, sunt pornite mai multe joburi de server. Când sosește o cerere, jobul care recepționează cererea tratează cererea de job, în timp ce următorul job de server disponibil așteaptă sosirea următoarei cereri. O dată ce jobul server îndeplinește cererea, închide conexiunea și se termină. Un nou job server pornește și ciclul continuă.

Cu acest model, nu trebuie să vă preocupați de intrările de job prestart. Însă uneori nu este posibilă configurarea subsistemelor în mod unic pentru mediul dumneavoastră, din cauză că acest model rulează în subsistemul implicit. O excepție este FTP (File Transfer Protocol). Pentru FTP puteți configura subsistemul în care rulează serverul FTP. Nu este nici o abilitate ca o parte din lucrul FTP să ruleze într-un subsistem iar restul lucrului să ruleze în alt subsistem. De asemenea, din perspectiva performanței, costul inițierii și terminării jobului nu poate fi evitat, pentru că după ce este rulat un job el este oprit și pornește un alt job. Dar, pentru că joburile se termină când conexiunea este terminată și este pornit următorul job, noul job în general va fi gata de execuție când este primită următoarea cerere, astfel încât costul inițierii și terminării jobului să nu afecteze timpul necesar pentru conectarea la server.

Unele exemple de joburi de server care utilizează modelul jobului cu ascultare multiplă sunt FTP și demonul imprimantei (LPD).

Pentru informații mai detaliate despre numele de job ale joburilor de server care rulează în sistem, vedeți tabela de joburi de server. Această tabelă vă arată subsistemul și numele jobului astfel încât să puteți găsi jobul activ și înregistrările lui în istoric. Tabela de asemenea arată descrierea job folosită de fiecare job server. Implicit, cele mai multe joburi server nu generează un istoric job când jobul se termină (parametrul LOG este setat la 4 0 *NOLIST), care înseamnă că istoricul job nu este creat. Dacă doriți să fie generat un istoric job cu toate mesajele trimise în istoricul job, parametrul LOG trebuie să fie 4 0 *SECLVL.

Joburi sistem:

Joburile sistem sunt create de către sistemul de operare pentru a controla resursele sistem și pentru a executa funcțiile sistem. Joburile de sistem rulează când serverul iSeries pornește sau când un pool de disc independent este în variere pe activat. Aceste joburi realizează o varietate de operații de la pornirea sistemului de operare, la pornirea și terminarea subsistemelor, la planificarea joburilor.

Joburi la pornirea sistemului:

Joburi de pornire sunt joburile de sistem care rulează la IPL. Ele tratează operațiile care pregătesc mediul sistemului de operare pentru muncă. Următoarea este o listă a diverselor joburi de pornire sistem.

Scpf (pornire funcție de control program)

Acesta este jobul central când porniți sistemul. Scpf pornește seria Qsysarb, dar Qsysarb3 pornește majoritatea celorlalte joburi de sistem (nu Qlus) și aduce sistemul într-o stare utilizabilă. Acest job rămâne activ după ce pornește sistemul, furnizând un mediu pentru rularea funcțiilor cu prioritate mică și cele sistem cu posibilă rulare îndelungată. Scpf de asemenea rulează în timpul procesării de oprire (Pwrdownsys) și este jobul care oprește procesarea mașinii.

Qwebtclnup (curățare tabelă de job)

Acest job este folosit în timpul pornirii sistemului pentru a se asigura că structurile de job sunt disponibile pentru folosire. El termină procesarea, de obicei, înaintea ca sistemul să termine de pornit, dar poate continua să ruleze după ce sistemul pornește, dacă există multe structuri de job de curățat. Acest job sistem se termină când se termină procesarea.

Qlpsvr (acceptarea acordurilor de software)

Acest job este automat pornit în timpul unui IPL dacă acordurile software online trebuie acceptate. Jobul se termină când toate acordurile sunt fie acceptate sau declinate.

Arbitri sistem:

Arbitrii sistemului (QSYSARB și de la QSYSARB2 la QSYSARB5), porniți de către un job de sistem SCPF, furnizează mediul pentru rularea funcțiilor cu prioritate ridicată. Ei permit sistemelor să pornească și să oprească și să urmărească starea sistemului (de exemplu, o stare restricționată).

Arbitrii sistemului, identificați de numele de job QSYSARB și de la QSYSARB2 la QSYSARB5, sunt joburile centrale și cu prioritatea cea mai mare în sistemul de operare. Fiecare arbitru de sistem răspunde evenimentelor din întreg sistemul care trebuie tratate imediat și acelor care pot fi tratate mai eficient de către un singur job decât de mai multe joburi.

Arbitrul sistem (QSYSARB) este, de asemenea, responsabil pentru pornirea jobului QLUS (Logical Unit Services - Servicii de unitate logică) în timpul unui IPL. Arbitrul sistem rămâne activ până când sistemul este oprit..

Următoarea este o listă de arbitrii sistem.

Qsysarb (arbitru sistem)

Arbitrul sistem furnizează mediul pentru rularea funcțiilor cu prioritate înaltă. El manipulează resursele sistem și ține evidența stării sistemului. Arbitrul sistem răspunde evenimentelor de la nivelul sistemului care trebuie să fie tratate imediat și acelea care pot fi tratate mai eficient de către un singur job. Qsysarb, Qtaparb (arbitru de bandă), și Qcmnarbxx (arbitrii de comunicații) sunt responsabili de procesarea cererilor de comunicație, blocarea dispozitivelor, configurarea liniei, a controler-ului și a dispozitivului și tratarea resurselor din răspândite în întregul sistem.

Qsysarb2 (arbitrul sistem 2)

Acest job este responsabil de gestionarea resurselor bandă, pentru tratarea spațiilor analizatorului de comenzi pentru procesarea de comenzi și pentru alte procesări la nivelul sistemului pentru sistemul de operare.

Qsysarb3 (arbitru sistem 3)

Acest job este responsabil pentru crearea și întreținerea structurilor job din sistem. Oricând sunt necesare structurile job permanente sau temporare pentru inițierea jobului, cererea este procesată de Qsysarb3. Qsysarb3, de asemenea, pornește și oprește multe dintre joburile de sistem..

Qsysarb4 (arbitru sistem 4)

Acest job este responsabil pentru pornirea și terminarea subsistemelor. Aceasta include procesarea de oprire (Pwrdownsys).

Qsysarb5 (arbitru sistem 5)

Acest job este responsabil pentru procesarea evenimentelor mașină. Aceasta include tratarea evenimentelor pentru a suporta alimentare auxiliară, pool-uri de memorie auxiliară de sistem (ASP-uri), prag de spațiu de stocare și limitele tabeli de blocare. De obicei, evenimentele mașină sunt tratate și mesajele corespunzătoare CPF sunt trimise la Qsysopr și Qhst.

Joburi de comunicație de sistem:

Subiectul conține o listă de joburi de comunicație de sistem.

Qlus (servicii unitate logică)

Qlus se ocupă de tratarea evenimentelor pentru dispozitive de unitate logică, cunoscute ca dispozitive de comunicație. Qlus este de asemenea responsabil pentru alocarea dispozitivelor subsistemului de comunicații corect.

Qcmnarbxx (arbitri de comunicație)

Arbitrii de comunicații cu Qsysarb (arbitrul de sistem) și Qtaparb (arbitrul de bandă) procesează lucrul pentru toate tipurile de dispozitiv, nu doar pentru dispozitive de comunicații. Acest lucru include conexiunea de comunicații, deconectarea, blocarea dispozitivului și procesarea de recuperare eroare. La repornire, valoarea

de sistem QCMNARB (joburi de arbitru de comunicație) determină numărul de joburi de arbitru de comunicații care sunt pornite. Un minim de 3 arbitri de comunicații sunt porniți pe sistemele cu un singur procesor.

Qsyscomm1 (comunicații de sistem)

Acest job tratează unele comunicații și activitate intrare/ieșire (I/O).

Q400filsvr (comunicații de sistem de fișiere aflate la distanță)

Acest job realizează comunicațiile interfeței comune de programare (APPN sau APPC) pentru aceste sisteme de fișiere aflate la distanță.

Joburile bază de date:

Următoarea este o listă de joburi de bază de date.

Qdbfstccol (colecție statistici de fișiere de bază de date)

Acest job colectează statistici de fișiere de bază de date. Aceste statistici au o importanță crucială la corectarea optimizării interogării de baze de date.

Qdbsrvxr (referință încrucișată de bază de date) și Qdbx####xr pentru grup de pool-uri de disc independente ###

Acest job întreține fiecare fișier referință încrucișată sistem nivel fișier din Qsys. Aceste fișiere conțin informații referință încrucișată despre fișierele bază de date și informații SQL din sistem. Fișierele încep toate cu prefixul Qadb din biblioteca Qsys. Fișierul primar care trebuie întreținut este Qdbxref, fișierul referință încrucișată. Acest fișier conține o înregistrare a fiecărei baze de date fizice, baze de date logice, și fiecărui fișier Alias din sistem. Qdbsrvxr se activează când un fișier este creat, modificat, șters, restaurat, redenumit sau dreptul de proprietate este modificat.

Qdbsrvxr2 (referință încrucișată de bază de date 2) și Qdbx####xr2 pentru grup de pool-uri de disc independente ###

Acest job întreține cele 2 fișiere referință încrucișată nivel. Qadbifld din biblioteca Qsys este fișierul referință încrucișată câmp. Qadbkfld din biblioteca Qsys este fișierul de referințe încrucișate pentru câmpul de chei. Qdbsrvxr2 este activat când un fișier este creat, modificat sau șters.

Qdbsrv01 (server de bază de date) și Qdbs####v01 pentru grup de pool-uri de disc independente ###

Acest job poate fi văzut ca dispecer de task de întreținere bază de date. Numărul de joburi server bază de date din sistem este cu unul în plus față de dublul numărului de procesoare sau cu unul în plus față de dublul numărului de ASP-uri, care este mai mare. Minimul de plecare este 5. Qdbsrv01 este jobul sistem principal care alocă lucrul celorlalte joburi. De obicei, Qdbsrv01 va fi cel mai activ imediat după restaurarea unei biblioteci care conține fișiere bază de date. Funcțiile sale includ:

- Semnalarea task-urilor LIC (Licensed Internal Code - Cod intern licențiat) cu SMAPP (system-managed access path protection - protecție cale de acces administrată de sistem) faptul că noile căi de acces au fost restaurate. SMAPP apoi determină dacă aceste căi de acces trebuie să fie protejate.
- Prepararea listei de căi de acces care sunt necesare a fi reconstruite din cauza căilor de acces care nu au fost restaurate.

Din joburile server bază de date rămase, jumătate procesează cereri cu prioritate înaltă și cealaltă jumătate procesează cereri cu prioritate mică. (Exemplu: de la Qdbsrv02 la Qdbsrv05 sunt cu prioritate înaltă, de la Qdbsrv06 la Qdbsrv09 sunt cu prioritate redusă.)

Qdbsrvxx (server de bază de date, prioritate înaltă) și Qdbs####vxx pentru grup de pool-uri de disc independente ###

Acest joburi execută întreținerea de control lansare și jurnalizare pentru sistem și sunt considerate ca fiind lucru rapid sau de scurtă durată.

Qdbsrvxx (server de bază de date, prioritate înaltă) și Qdbs####vxx pentru grup de pool-uri de disc independente ###

Aceste joburi execută întreținerea căii de acces la fișierele de date utilizator. De obicei, aceste joburi sunt inactive, dar în anumite cazuri, este posibil să se activeze pentru a realiza reconstruirea de căi de acces. Unele din motivele pentru care aceste joburi este posibil să fie active sunt:

- Restaurarea fișierelor bază de date care nu au fost salvate cu căi de acces
- Restaurarea fișierelor logice fără fișierul fizic pe care se bazează
- Anularea unei comenzi Rgzpfm în timpul procesării

- Invalidarea unui index din cauza unei deteriorări găsite în index
- Activitate post-instalare iServer pentru a termina referințele încrucișate sau alte activități de modernizare bază de date.
- Verificarea de constrângeri

Qqtemp1 și Qqtemp2 (paralelism de bază de date)

Joburile de sistem de paralelism de bază de date realizează procesarea asincronă de baze de date pentru DB2 Multisystem. Dacă utilizatorii cer fișiere distribuite, joburile sunt folosite pentru a accelera cererile prin realizarea anumitor task-uri în paralel.

Alte joburi de sistem:

Acest articol conține o listă a altor feluri de joburi de sistem.

Qalert (managerul de alertă)

Acest job realizează operațiile necesare pentru a procesa alerte. Aceasta include activități precum procesarea de alerte primite de la alte sisteme, procesarea de alerte create local și întreținerea sferei de control.

Qdcobjx (obiect de decompresie sistem)

Acest job decompromă obiecte ale sistemului de operare proaspăt instalate după cum este necesar. Există o cerință de spațiu de stocare pentru ca aceste joburi să ruleze. Dacă spațiul de stocare disponibil din sistemul dumneavoastră scade sub o anumită limită, aceste joburi se vor termina. Numărul joburilor de decompresie obiecte sistem este numărul de procesoare plus unu.

Qfilesys1 (sistem de fișiere)

Acest job suportă procesare în fundal a sistemului de fișiere integrat. El asigură că aceste modificări asupra fișierelor sunt scrise în spațiul de stocare și mai realizează, de asemenea, mai multe activități de curățare a sistemului de fișiere.

Qjobscd (planificare job)

Acest job controlează funcțiile de planificare ale jobului sistemului. Qjobscd monitorizează cronometrele pentru intrările de planificare job și ale joburilor planificate.

Qli###cl pentru grupul de pool-uri de discuri independente ### (curățare bibliotecă)

Acest job curăță bibliotecile din pool-uri de discuri independente.

Qli###rp pentru grupul de pool-uri de discuri independente ### (curățare obiect)

Această intrare curăță obiectele înlocuite din bibliotecile pool-urilor de discuri independente.

Qlur (LU 6.2 resincronizare)

Qlur tratează procesarea de resincronizare de lansare în 2 faze.

Qpfradj (ajustarea performanțelor)

Acest job gestionează modificările aduse dimensiunilor pool-ului de stocare și nivelurilor de activitate. Toate cererile pentru modificarea pool-urilor de stocare sunt procesate de acest job. În plus, dacă valoarea de sistem Qpfradj, ajustare automată a pool-urilor de memorie și a nivelurilor de activitate, este setată pe valoarea 2 sau 3, acest job modifică în mod dinamic dimensiunile și nivelurile de activitate a pool-urilor de stocare pentru a îmbunătăți performanțele sistemului.

Qsplmaint (întreținere pool de sistem) și Qspmn##### pentru grupul de pool-uri de discuri independente #####

Acest job execută funcțiile de introducere în spool ale sistemului care includ:

- Curățarea fișierelor spool după un IPL sau după ce este activat un grup de pool-uri de discuri independente
- Mutarea fișierele spool eșuate sau cozilor de ieșire utilizator deteriorate din spațiul de stocare auxiliar al subsistemului sau dintr-un pool de memorie auxiliar utilizator de bază în coada de ieșire QSPRCLOUTQ din biblioteca QRCL
- Curăță membrul de bază de spool care conține datele și atributele unui fișier spool șters
- Șterge membrii bazei de date spool care nu au fost reutilizați în perioada specificată în valoarea de sistem QRCLSPLSTG (curățare automată a spațiului de stocare neutilizat pentru ieșiri imprimantă)

Qspdf##### pentru grupul de pool-uri de discuri independente ##### (actualizator spool de sistem PRTQ)

Acest job realizează operații de fișier spool pentru grupuri specifice de pool-uri de discuri independente.

Qtaparb (dispozitiv de bandă)

Acest job procesează munca legată de dispozitivele cu bandă inclusiv blocarea dispozitivelor și procesarea recuperării erorilor.

| Qnwharbxx

Aceste joburi de sistem tratează evenimente legate de dispozitivele Adaptor gazdă server de rețea (NWSH). Întotdeauna va fi cel puțin unul din aceste joburi pornit în timpul IPL-ului curent.

| Qwcpjobs

Acest job tratează curățarea din fundal a structurilor de job permanente.

| Qwctjobs

Acest job tratează curățarea din fundal a structurilor de job temporare.

Opțiunile de planificare job

Funcția de planificare job permite la planificarea dependență-timp a joburilor batch iSeries. Puteți planifica joburi să fie eliberate din coada de joburi la un moment anume sau puteți folosi o intrare de planificare job pentru a vă trimite jobul la coada de joburi automat la momentul pe care l-ați specificat. Planificarea jobului vă permite să controlați data și ora la care un job batch este lansat sau devine eligibil să pornească dintr-o coadă de joburi. Această flexibilitate vă poate ajuta în timp ce echilibrați sarcina de lucru pe sistemul dumneavoastră.

De exemplu, puteți folosi planificarea jobului pentru a delega operația care se repetă a observațiilor de îndeplinire a lansării repetate, ștat de plată, sau rapoarte săptămânale și lunare de la planificarea dumneavoastră la planificarea sistemului. Există patru metode pentru a planifica un job batch.

Planificatorul din Administrare centrală

Navigatorul iSeries furnizează un planificator integrat, planificatorul din Administrare centrală, pentru a organiza momentele când doriți ca joburile dumneavoastră să proceseze. Aveți opțiunea de a alege să realizați o operație imediat sau să alegeți o oră mai târziu. Puteți utiliza planificatorul din Administrare centrală pentru a planifica aproape orice operație din Administrarea centrală.

Fereastra Planificator Administrare centrală este disponibilă oricând vedeți un buton de **Planificare** într-o fereastră de Navigator iSeries.

Notă: Dacă ați instalat Advanced Job Scheduler pe serverul de Administrare centrală, butonul **Planificare** va porni Advanced Job Scheduler în locul planificatorului din Administrare centrală.

Advanced Job Scheduler

Programul cu licență Advanced Job Scheduler (5722-JS1) este un planificator robust, care permite procesarea nesupravegheată a joburilor 24 de ore pe zi, 7 zile pe săptămână. Această unealtă de planificare furnizează mai multe caracteristici de calendar și oferă un control mai mare asupra evenimentelor planificate decât planificatorul Administrare centrală. Puteți, de asemenea, vizualiza istoricul efectuării unui job și puteți gestiona notificări despre starea unui job.

Nu este necesar să instalați programul cu licență Advanced Job Scheduler pe fiecare sistem punct final din rețeaua dumneavoastră de Administrare centrală. Când instalați Advanced Job Scheduler pe sistemul central, joburile sau operațiile pe care le definiți pe un sistem punct final vor aduna informații despre job care sunt necesare sistemului central. Însă trebuie să setați pe sistemul central toate informațiile definiției de job.

Dacă sistemele din rețeaua dumneavoastră au Advanced Job Scheduler instalat local, puteți planifica task-uri în afara rețelei de Administrare centrală. Sub **Conexiunile mele** din Navigatorul iSeries, aveți acces la Advanced Job Scheduler pe acel sistem local când expandați **Control funcționare**.

Advanced Job Scheduler for Wireless:

Advanced Job Scheduler for Wireless este o aplicație software care vă permite să accesați Advanced Job Scheduler pe mai multe dispozitive cu acces la Internet, precum un telefon cu funcții de Internet, un browser Web PDA sau un browser Web pentru PC.

Caracteristica de comunicație fără fir Advanced Job Scheduler se află pe sistemul dumneavoastră iSeries, unde Advanced Job Scheduler este instalat și vă permite să accesați joburile și activitatea dumneavoastră, precum și să trimiteți mesaje către destinatari de pe sistemul dumneavoastră și să opriți sau să porniți monitorul Advanced Job Scheduler. Advanced Job Scheduler for Wireless permite fiecărui utilizator să personalizeze setările și preferințele răsfoirii sale. De exemplu, un utilizator poate afișa activitatea și joburile și poate personaliza joburile afișate.

Advanced Job Scheduler for Wireless vă permite să vă accesați joburile când, normal, nu puteți să accesați un terminal sau emulator iSeries. Conectați-vă la Internet cu dispozitivul dumneavoastră mobil și introduceți URL-ul pentru servlet-ul Advanced Job Scheduler for Wireless. Aceasta va lansa un meniu care vă oferă acces în timp real la Advanced Job Scheduler.

Advanced Job Scheduler for Wireless lucrează pe două tipuri de dispozitive. Un dispozitiv WML (Wireless Markup Language) este un telefon celular pentru Internet. Un dispozitiv HTML (Hypertext Markup Language) este un browser de Web pe PDA sau pe PC. În acest subiect dispozitivele sunt numite WML și HTML.

Intrări de planificare job

Dacă sistemul dumneavoastră nu are Planificatorul din Administrare centrală sau Advanced Job Scheduler, puteți totuși să planificați joburi utilizând o intrare de planificare job, care este accesată de la interfața bazată pe caractere. Utilizând această metodă puteți planifica joburi să se repete sau să ruleze doar o dată.

Din moment ce intrările de planificare job sunt intrări într-un obiect permanent, ele nu rămân în coada de joburi precum joburile planificate și de aceea ele nu se pierd când este curățată coada de joburi. Mai puteți, de asemenea, salva și restaura obiectul de planificare job. Acesta furnizează o metodă de copiere de rezervă a informațiilor dumneavoastră de planificare job.

Când doriți ca un job să se proceseze la intervale regulate, creați o intrare de planificare job pentru job. Intrarea de planificare job conține toate informațiile care sunt necesare pentru a lansa un job și informațiile sale de planificare. Fiecare intrare din obiect este identificată unic de către numele jobului pe care îl furnizați și de către un număr de intrare de 6 cifre care este alocat de către sistem. Nu există două intrări care să aibă aceeași combinație de nume de job și număr de intrare.

Intrarea de planificare job mai conține, de asemenea, informații utilizate de către sistem pentru a gestiona intrarea în anumite situații. Informațiile care definesc jobul sunt similare cu parametrii specificați într-o comandă SBMJOB (Submit Job - Lansare job), inclusiv numele jobului, descrierea jobului, coada de joburi, profilul utilizator și coada de mesaje. Zona de date locale (LDA), a jobului lansat din intrarea de planificare job, este goală atunci când jobul pornește.

Toate intrările de planificare job sunt conținute în obiectul de planificare job. Obiectul de planificare job, QDFTJOBSCD este în biblioteca QUSRSYS și are un tip de obiect *JOBSCD. Nu puteți crea, redenumi sau duplica obiectul de planificare job. Nu puteți să îl mutați în nici o altă bibliotecă. Obiectul de planificare job este livrat cu autorizarea publică de *CHANGE. Aceasta este autorizarea minimă necesară pentru a adăuga, modifica, reține, elibera și înlătura intrările de planificare job.

Notă: Puteți, de asemenea, planifica joburile care reapar utilizând Planificatorul din Administrare centrală sau Advanced Job Scheduler.

Exemple: intrarea de planificare job:

Acest subiect furnizează exemple pentru utilizarea comenzii ADDJOBSCDE (Add Job Schedule Entry - Adăugare intrare de planificare job).

Planificarea unui job lunar: Acest exemplu vă arată cum să lansați un job să ruleze programul INVENTORY la 23:30 în ultima zi a fiecărei luni cu excepția ajunului Anului Nou.

```
ADDJOBSCDE JOB(MONTHEND)
CMD(CALL INVENTORY)
SCDDATE(*MONTHEND)
SCDTIME('23:30:00')
FRQ(*MONTHLY)
OMITDATE('12/31/05')
```

Planificarea unui job zilnic: Acest exemplu vă arată cum să lansați un job să ruleze programul DAILYCLEAN în fiecare zi la ora 18:00. Jobul rulează sub profilul utilizator SOMEPMGR. Acest job nu este lansat dacă sistemul este oprit sau este într-o stare restricționată la acea oră.

```
ADDJOBSCDE JOB(*JOB)
CMD(CALL DAILYCLEAN)
SCDDAY(*ALL)
SCDTIME('18:00:00')
SCDDATE(*NONE)
USER(SOMEPMGR)
FRQ(*WEEKLY)
RCYACN(*NOSBM)
```

Planificarea unui job săptămânal: Acest exemplu vă arată cum să lansați un job să ruleze programul PGM1 în fiecare săptămână începând de pe 12/17/05 la ora curentă. Deoarece 12/17/05 este o zi de Sâmbătă, jobul este lansat în fiecare Sâmbătă și rulează sub profilul utilizator

```
PGM1. ADDJOBSCDE JOB(*JOB)
CMD(CALL PGM1)
SCDDATE('12/17/05')
FRQ(*WEEKLY)
USER(PGM1)
```

Planificarea unui job în fiecare a treia zi de luni și de miercuri: Acest exemplu vă arată cum să lansați un job să ruleze programul PGM2 în fiecare a treia zi de luni și a treia zi de miercuri la 23:30. Acest job va fi lansat în următoarea a treia zi de luni sau de miercuri la ora 23:30, depinzând dacă aceste zile au trecut deja în această lună. Dacă ieri a fost a treia zi de luni, astăzi este a treia marți și mâine va fi a treia miercuri, el va fi lansat mâine și apoi abia luna viitoare.

```
ADDJOBSCDE JOB(*JOB)
CMD(CALL PGM2)
SCDDAY(*MON *WED) FRQ(*MONTHLY)
SCDDATE(*NONE)
RELDAYMON(3) SCDTIME('23:30:00')
```

Planificarea unui job în fiecare prima și a treia zi de luni: Acest exemplu vă arată cum să lansați un job să ruleze programul PAYROLL în prima și a treia zi de luni a fiecărei luni calendaristice la 9:00. Jobul rulează sub profilul utilizator PAYROLLMGR.

```
ADDJOBSCDE JOB(PAYROLL)
CMD(CALL PAYROLL)
SCDDAY(*MON) FRQ(*MONTHLY)
SCDDATE(*NONE)
RELDAYMON(1 3) SCDTIME('09:00:00')
USER(PAYROLLMGR)
```

Planificarea unui job în fiecare zi lucrătoare: Acest exemplu vă arată cum să lansați un job pentru a rula programul PGM4 în fiecare zi din săptămână la 23:00.

```
ADDJOBSCDE JOB(*JOB)
CMD(CALL PGM4)
SCDDAY(*MON *TUE *WED *THU *FRI)
SCDDATE(*NONE)
SCDTIME('19:00:00') FRQ(*WEEKLY)
```

Salvarea unei intrări de planificare job: Acest exemplu vă arată cum să lansați un job o dată și să salvați intrarea.

```
ADDJOBSCDE JOB(*JOBID)
CMD(CALL SAVED)
FRQ(*ONCE)
SAVE(*YES)
```

Comanda de lansare în execuție job

Această comandă a interfeței bazate pe caractere controlează ora la care un job este eliberat din coada de joburi. Există o metodă ușoară de planificare a unui job care trebuie să ruleze doar o dată. Ea vă permite să utilizați multe din atributele de job definite pentru jobul dumneavoastră curent.

Când planificați un job să ruleze doar o dată (comanda bazată pe caractere SBMJOB), jobul este eliberat din coada de joburi la momentul planificat. Următorul este un rezumat al operațiilor de sistem care apar când utilizați SBMJOB pentru a planifica un job batch.

1. Dumneavoastră planificați un job utilizând fie interfața Navigatorului iSeries (**Operații de bază → Joburi → Faceți clic dreapta pe job → Proprietăți → fișa Coadă de joburi**) sau interfața bazată pe caractere (comanda SBMJOB cu parametrii **SCDATE** și **SCDTIME** specificați).
2. Jobul rămâne în coada de joburi în starea planificat (SCD status) până la data și ora indicate de parametrii.
3. La ora planificată, jobul este eliberat din coada de joburi. Starea jobului se modifică din planificat (SCD) în eliberat (RLS), doar dacă jobul nu este reținut (SCDHLD), în acest caz starea modificându-se din planificat în reținut (HLD).
4. Jobul este procesat ca orice alt job din coada de joburi.
5. Jobul pornește dacă există condiții normale (cum ar fi o coadă de joburi alocată unui subsistem activ și nu sunt încă active numărul maxim de joburi).

Notă: Această metodă pune jobul în coada de joburi imediat, astfel, dacă coada de joburi este curățată înainte de data și ora planificate, vă veți pierde jobul.

Considerații pentru planificatorul de joburi

Când alegeți un produs planificator de joburi, trebuie să luați în considerare o varietate de caracteristici diferite. Următoarea este o listă de caracteristici de luat în considerare atunci când se determină ce fel de planificator de joburi se folosește:

- **Planificare automatizată a joburilor**
 - Flexibilitate în planificarea joburilor
 - Procesare de job nesupraveheat (sau supraveheat) 24 ore pe zi, 7 zile pe săptămână, în compatibilitate totală pentru planificările pe care le-ați setat
 - Extensia naturală a sistemului de operare iSeries
 - Controlul complet a cum, când și unde este lansat un job
 - Dependente de job extensive cum ar fi obiecte (existența unui fișier sau a unor înregistrări într-un fișier fizic), activitatea sau inactivitatea altor joburi, sau starea unei linii, a unui controler, sau a unui subsistem
 - Terminații funcțiile legate de calendar, inclusiv calendarele de vacanță și fiscal
 - Rulări multiple pe zi
- **Parametri de sistem și definiții de utilizator**
 - Data curentă, data de lansare, data anterioară și ora curentă pot fi transmise în programe de aplicație
 - Valorile parametrului definit de utilizator pot fi create, modificate și transmise în programe aplicații
- **Difuzare încărcare de lucru/istoric**
 - Difuzarea tuturor joburilor planificate pentru a rula săptămâna următoare, luna următoare, sau ziua următoare
 - Optimizare cerințe de producție
 - Urmărirea istorică și înregistrarea tuturor activităților Advanced Job Scheduler.
- **Gestiune de rețea**
 - Joburile pot fi setate să ruleze pe orice server iSeries din rețea sau orice alt server iSeries din rețea
 - Furnizează istoricul de job complet a jobului de pe sistemul lansat

- Joburile dependente și de grup pot fi lansate în rețea
- **Distribuție de raport și gestiune**
 - Rutarea, monitorizarea și controlul tuturor rapoartelor de ieșire generate de Advanced Job Scheduler sau sistemul de operare iSeries
 - Distribuția fișierului spool la cozi de ieșire multiple sau la sisteme la distanță cu pagini de banner opțional
 - Ieșirea spool poate fi duplicată sau trimisă oricărui utilizator din rețeaua iSeries
- **Securitate**
 - În Advanced Job Scheduler poate fi utilizată securitatea iSeries existentă.
 - Specificați cine din organizația dumneavoastră are autorizarea de setare sau modificare a informațiilor despre joburile planificate
 - Autorizarea poate fi specificată pentru funcții individuale Advanced Job Scheduler sau pentru joburi specifice
- **Interfața grafică utilizator**
 - Indicați și faceți clic pe capabilități când planificați un job
 - Gestionare joburi
 - Menținere dependențe
 - Urmărire activitate planificator și informații de istoric
- **Alte caracteristici cheie**
 - Comenzi multiple per job
 - Definiție pentru jobul LDA (Zonă de date locală)
 - Monitorizare consolă pentru rularea joburilor în stare restricționată
 - Verificarea timpului de rulare maxim pentru fiecare job
 - Interfațare direct la un sistem de paginare terță parte bazat pe mesaj
 - Aprovizionarea din plin cu documentație online a fiecărui job
 - Text de ajutor sensibil la cursor extensiv pe toate ecranele

Planificare job și disponibilitate sistem

Dacă este oprită alimentarea sistemului sau sistemul este într-o stare restricționată când s-a ajuns la orele planificate, joburile nu pot fi lansate în execuție din intrările de planificare job și starea joburilor planificate nu poate fi modificată. Totuși, puteți controla modul în care sistemul tratează această situație supă IPL-ul sistemului sau după ce acesta iese din starea restricționată.

Intrările de planificare job și joburile planificate sunt procesate în ordinea în care aparițiile scăpate ar fi fost tratate manual. Este posibil ca în sistem să intre lucru din alte surse în timp ce sunt procesate intrările de planificare job și joburile planificate.

- **Intrări de planificare job:** Puteți controla modul în care fiecare intrare este tratată după valoarea pe care o specificați pentru acțiunea de recuperare a intrării. Puteți specifica un job să fie încă lansat utilizând intrarea, ca un job să fie lansat și reținut într-o coadă de joburi sau că un job nu ar trebui să fie lansat. Dacă cereți ca un job să fie lansat, este lansat doar un job de la fiecare intrare, indiferent câte lansări au fost scăpate în timpul cât sistemul nu era disponibil.
- **Job planificat:** Sistemul verifică pentru a determina dacă a trecut vreo oră planificată în timpul cât sistemul nu era disponibil. Dacă este găsit un job planificat a cărui oră a trecut, starea jobului este actualizată.

Cozile de joburi

O coadă de joburi conține o listă ordonată de joburi care așteaptă să fie procesate de un subsistem. Coada de joburi este primul loc în care merge un job batch lansat în execuție, înainte de a deveni activ într-un subsistem. Jobul este ținut aici până este întrunit un număr de factori.

Pentru ca joburile dintr-o coadă de joburi să fie procesate, trebuie să fie un subsistem activ care acceptă lucrul de la acea coadă de joburi. Când pornește un subsistem, încearcă să aloce cozile de joburi pe care este configurat să le

accepte, și trebuie să aloce cu succes o coadă de joburi pentru a procesa joburi de la acea coadă de joburi. De aceea, în timp ce un subsistem poate procesa joburi din mai multe cozi de joburi, doar un subsistem poate procesa joburi dintr-o anumită coadă de joburi la un moment dat.

Subsistemele selectează joburi din cozile de joburi în ordinea priorității, între limite care pot fi configurate pentru fiecare prioritate. Fiecare job are o prioritate de coadă de joburi care poate fi gestionată, când jobul este în coada de joburi, prin proprietățile jobului. Un set de bază de cozi de joburi este furnizat cu sistemul dumneavoastră. În plus, puteți crea cozile suplimentare de joburi de care aveți nevoie.

Notă: Pot fi apelate API-uri, precum Open List of Job Queues (Deschide listă de cozi de joburi) (QSPOLJBQ) și Retrieve Job Queue Information (Extrage informații coadă joburi) (QSPRJOBQ), pentru a obține informații despre cozi de joburi.

Listă ordonată

Lista ordonată se referă la ordinea în care joburile apar în coada de joburi. Disponibilitatea, prioritatea, și valorile de dată și ora ajută să determinați ordinea joburilor din coada de joburi.

Numărul jobului nu este folosit pentru a determina unde jobul apare în coada de joburi, nici nu afectează când jobul rulează.

Disponibilitatea

Referire la starea jobului pe coada de joburi. Valorile de comandă posibile sunt în așteptare, planificate, și ținute.

Prioritate

Se referă la prioritatea jobului pe care o are în coada de joburi. Valorile de prioritate posibile sunt între 0-9, 0 fiind prioritatea cea mai înaltă. În cazuri unde joburile sunt joburi planificate, prioritatea nu joacă un rol în stabilirea joburilor în coada de joburi. De exemplu, dacă două joburi sunt planificate să ruleze la 12:00:00, joburile sunt comandate de către poziția lor în tabela de joburi.

Data și ora

Referire la data și ora jobului:

- Dacă jobul este planificat, data și timpul se referă la momentul în care este planificată rularea.
- Dacă jobul nu este planificat, data și timpul se referă la momentul în care jobul a intrat în sistem.

Notă: Sunt unele cazuri în care data și timpul ajung să fie valori setate manual pentru a poziționa corespunzător un job mutat într-o anumită coadă de joburi.

Cum lucrează o coadă de joburi

Cozile de joburi sunt alocate de către un subsistem prin intrarea în coada de joburi. Joburile pot fi puse într-o coadă de joburi chiar dacă subsistemul nu a fost pornit. Când subsistemul este pornit, el procesează joburile din coadă.

Descrierea subsistemului specifică numărul maxim de joburi (batch sau interactive) care pot fi active simultan. Numărul de joburi care pot fi active pentru orice coadă de joburi este specificat în intrarea în coada de joburi.

Nu toate joburile dintr-o coadă de joburi sunt în mod necesar disponibile pentru procesare atunci când este pornit subsistemul. Joburile planificate pot fi puse în coada de joburi. Joburile pot fi reținute într-o coadă de joburi până când operatorul de sistem le eliberează. Dacă subsistemul este oprit înainte să fie procesate toate joburile, joburile rămân în coadă până când subsistem este repornit, până când sunt mutate de operatorul de sistem în altă coadă, până când sunt șterse de operatorul de sistem sau până când alt subsistem alocă aceeași coadă de joburi.

La aceeași coadă de joburi se pot referi mai multe descrieri de subsistem, dar numai un subsistem activ poate utiliza la un moment dat coada de joburi ca sursă de joburi batch. De aceea, dacă un subsistem se oprește și mai sunt încă joburi în coada de joburi, alt subsistem care se referă la acea coadă de joburi poate fi pornit pentru a procesa joburile. Dacă este deja pornit alt subsistem și așteaptă aceeași coadă de joburi, subsistemul alocă automat coada de joburi când aceasta devine disponibilă.

Cum sunt luate joburile dintr-o coadă de joburi

Diferiți factori determină modul în care sunt selectate și pornite joburile dintr-o coadă de joburi.

Numărul maxim de joburi active pentru subsisteme

Acesta reprezintă numărul maxim de joburi care pot să ruleze într-un subsistem. După ce această limită este atinsă, nici un job nu mai poate porni în subsistem.

Numărul maxim de joburi active pentru cozile de joburi

Acesta reprezintă numărul maxim de joburi din coada de joburi care pot rula într-un subsistem în același timp. După ce această limită este atinsă, nici un job nu mai poate porni de la acea coadă de joburi.

Prioritate în coada de joburi

Joburile care sunt așteptate să ruleze sunt selectate bazate pe prioritatea cozii de joburi. Subsistemul încearcă să ruleze întâi joburile cu prioritate mai mare (prioritatea cozii de joburi este până de la 0 la 9, unde 0 este prioritatea mai mare), dar dacă numărul joburilor care rulează de la un nivel de prioritate atinge valoarea Maximul joburilor active pe nivel de prioritate, următorul nivel de prioritate este procesat. (Dacă joburile cu aceeași prioritate intră în coada de joburi, primul job lansat va rula primul, apoi următorul și așa mai departe.)

Secvența

Specificați secvența în coada de joburi care intră în descrierea subsistemului. Numărul secvenței definește ordinea în care subsistemul va procesa cozile de joburi. Subsistemul ia joburile din coada de joburi începând cu cel mai mic număr de ordine. Dacă nu mai sunt joburi în coada de joburi, sau dacă una din valorile maxime asociate cu coada de joburi este atinsă, subsistemul va procesa coada de joburi începând cu succesivul celui mai mare număr de ordine.

Intrare coadă job

O intrare în coada de joburi identifică o coadă de joburi din care joburile sunt selectate pentru a rula în subsistem. Există cinci parametri în intrarea în coada de joburi care controlează felul în care coada de joburi ar trebui manipulată.

Descriere subsistem (SBSD)

Acesta este numele și biblioteca descrierii subsistemului la care este adăugată intrarea în coada de joburi.

Coadă de joburi (JOBQ)

Specifică numele și biblioteca cozii de joburi care este o sursă a joburilor batch care sunt pornite de către subsistem.

Joburi maxime active (MAXACT)

Specifică numărul maxim de joburi care pot fi active în același timp de la această coadă de joburi.

Număr de ordine (SEQNBR)

Specifică un număr de ordine pentru această coadă de joburi, care este folosită de către subsistem pentru a determina ordinea în care cozile de joburi sunt procesate.

Prioritate maximă activă 1 (prin 9) (MAXPTYx)

Specifică numărul de joburi care pot fi pornite pentru un nivel de prioritate de job specificat.

Cum sunt alocate cozile de joburi la un subsistem

O coadă de joburi poate fi asociată cu mai multe subsisteme dar poate fi alocată numai o dată la un subsistem. Când subsistemul este pornit, monitorul de subsistem încearcă să aloce fiecare coadă de joburi definită în intrările coadă de joburi ale subsistemului.

Dacă o coadă de joburi a fost deja alocată de către un alt subsistem, primul subsistem trebuie să se oprească și să dezalocă coada de joburi înainte ca cel de-al doilea subsistem să-l poată aloca. După ce este pornit, acest al doilea subsistem alocă cozile de joburi cei sunt alocate de îndată ce devin disponibile.

Dacă o coadă de joburi nu există când subsistemul este pornit, coada de joburi este alocată subsistemului când apare una din următoarele situații:

- Coada de joburi este creată.
- O coadă de joburi este redenumită cu numele definit la subsistem.

- O coadă de joburi este mutată la o altă bibliotecă și numele calificat rezultat se potrivește cu numele din descrierea subsistemului.
- Biblioteca care conține coada de joburi este redenumită numele calificat rezultat se potrivește cu numele din descrierea subsistemului.

Cozi de joburi multiple

În multe cazuri, utilizarea QBATCH ca singura coadă de joburi, cu valoarea de joburi active setată implicit pe unu, va fi adecvată necesităților dumneavoastră. Dacă acesta nu este suficientă, este posibil să doriți să aveți mai multe cozi de joburi astfel încât unele cozi de joburi să fie active în timpul orelor normale de lucru, unele să fie pentru scopuri speciale și altele să fie active după orele de lucru normale.

De exemplu, puteți desemna diferite cozi de joburi pentru:

Joburi cu durată mare de rulare astfel încât să puteți controla câte joburi sunt active simultan

Este posibil, de asemenea, să doriți ca aceste joburi să utilizeze o prioritate mai mică decât alte joburi batch.

Joburile de peste noapte, pentru care reprezintă un inconvenient să ruleze în timpul orelor normale de lucru

De exemplu, pentru a rula o comandă RGZPFM (Reorganize Physical File Member - Reorganizare membrii fișier fizic) pe un fișier bază de date mare necesită o privire exclusivă a fișierului. Aceasta înseamnă că alți utilizatori nu pot accesa fișierul în timp ce are loc această operație. În plus, această operație poate să dureze mult timp. Poate să fie mai eficient să fie pus acest job într-o coadă de joburi care rulează în timpul orelor de ieșire din tură.

Joburi cu prioritate mare

Este posibil să doriți să aveți o coadă de joburi în care să fie trimis tot lucrul care are o prioritate mare. Puteți atunci să vă asigurați că acest lucru este terminat rapid și nu este întârziat de joburile cu prioritate mai mică.

Joburi care sunt direcționate către cerințe de resurse particulare, precum dischetă sau casetă

Un astfel de job necesită un parametru **MAXACT** de valoare 1 în intrarea în coada de joburi a descrierii de subsistem astfel încât, la un moment dat, doar un job să utilizeze aceste resurse.

De exemplu, dacă este utilizată o casetă pentru mai multe joburi, toate joburile care utilizează caseta sunt puse într-o singură coadă de joburi. Apoi sunt selectate joburile din coada de joburi pe rând, câte unul. Aceasta asigură că nu vor exista două joburi în competiție pentru un dispozitiv în același timp. Dacă se întâmplă acest lucru, unul din joburi se termină cu o eroare de alocare.

Notă: Ieșirea de bandă nu poate fi pusă în spool.

Munca de programator

Este posibil să doriți să aveți o coadă de joburi care să se ocupe de munca de programator sau de tipuri de lucru care pot fi reținute în timp ce se rulează munca de producție.

Rularea secvențială a unei serii de joburi

Puteți avea o aplicație în care un job este dependent de terminarea altui job. Dacă puneți aceste joburi într-o coadă de joburi care selectează și rulează un job o dată, aceasta asigură ordinea de rulare a acestor joburi.

Dacă un job necesită controlul exclusiv al unui fișier, este posibil să doriți să îl puneți într-o coadă de joburi când coada este singura activă pe server, cum ar fi în timpul nopții sau la sfârșitul săptămânii.

Dacă utilizați cozi de joburi multiple, veți vedea că cel mai important considerent este controlul diverselor cozi de joburi. De obicei, veți dori să controlați:

- Câte cozi de joburi există
- Câte cozi de joburi sunt active într-un anumit subsistem în același timp
- Câte joburi active pot fi selectate dintr-o anumită coadă de joburi la un anumit moment dat
- Câte joburi pot fi active într-un subsistem la un anumit moment dat

Cum sunt luate joburile din mai multe cozi de joburi

Un subsistem procesează joburile dintr-o coadă de joburi pe baza unui număr de ordine. Un subsistem poate avea mai mult de o intrare în coada de joburi și de aceea poate aloca mai mult de o coadă de joburi.

Numărul maxim de joburi dintr-o coadă este specificat de parametrul **MAXACT** (Maximum active jobs - Maxim de joburi active) din comenzile ADDJOBQE (Add Job Queue Entry - Adăugare intrare în coada de joburi) sau CHGJOBQE (Change Job Queue Entry - Modificare intrare în coada de joburi). Puteți, de asemenea, controla câte joburi din fiecare prioritate pot fi active utilizând parametrii **MAXACTx** (Maximum active priority - Maxim de prioritate activă). De exemplu, dacă MAXACT=10, MAXACT5=2, și există trei joburi în coada de joburi cu nivelul de prioritate 5, atunci doar două dintre ele pot deveni active la orice moment dat.

Subsistemul procesează joburile din coada de joburi începând cu cele cu numărul de ordine cel mai mic. Atunci când toate joburile care sunt în coada de joburi au fost procesate sau când s-a ajuns la numărul maxim de joburi din coada de joburi, subsistemul procesează joburile din coada cu următorul cel mai mare număr de ordine.

Ordinea continuă până când subsistemul a procesat toate intrările în coadă de joburi disponibile sau până când subsistemul a ajuns la limita sa de joburi care pot rula sau aștepta în subsistem. Numărul de joburi care pot rula sau aștepta este determinat de parametrul **MAXACT** (Maximum active jobs - Maxim de joburi active) din descrierea subsistemului. În unele cazuri secvența este întreruptă pe măsură ce joburile se opresc sau sunt transferate. Crearea, reținerea și eliberarea cozilor de joburi poate, de asemenea, modifica ordinea cozilor de joburi procesate.

Securitatea cozii de joburi

Puteți menține un nivel de securitate cu coada dumneavoastră de job prin autorizarea numai a anumitor persoane (profiluri de utilizatori) la acea coadă de joburi. În general, există trei moduri prin care un utilizator poate deveni autorizat să controleze o coadă de joburi (de exemplu, reținerea sau eliberarea cozii de joburi).

- Utilizatorului i se alocă o autorizare de control spool (SPCAUT(*SPLCTL)) din profilul utilizator.
- Utilizatorului i se alocă o autorizare de control job (SPCAUT(*JOBCTL)) din profilul utilizator și coada de joburi poate fi controlată de către operator (OPRCTL(*YES)).
- Utilizatorul are autorizarea necesară de obiect asupra cozii de joburi. Autorizarea de obiect necesară este specificată de parametrul **AUTCHK** din comanda CRTJOBQ. Valoarea *OWNER indică faptul că doar proprietarul cozii de joburi este autorizat, prin intermediul autorizării de obiect, pentru coada de joburi. Valoarea *DTAAUT indică faptul că utilizatorii cu autorizarea *CHANGE pentru coada de joburi sunt autorizați să controleze coada de joburi.

Notă: Autorizările specifice, necesare pentru *DTAAUT sunt autorizările *READ, *ADD, și *DLT pentru date.

Aceste trei metode de autorizare se aplică doar cozii de joburi, nu și joburilor din coada de joburi. Regulile normale de autorizare pentru controlul joburilor se aplică dacă jobul este într-o coadă de joburi sau dacă el rulează momentan.

Cozile de ieșire

Cozile de ieșire sunt zone în care fișierele de ieșire imprimantă (de asemenea numite fișiere spool) așteaptă să fie procesate și trimise la imprimantă. Ieșirea imprimantă este creată fie de către sistem fie de către utilizatorul care folosește un fișier de tipărit.

Un fișier de imprimantă este similar cu un șablon sau o indicație unde valorile implicite pentru atributele ieșirii imprimantei sunt setate. Este începutul ciclului de viață al ieșirii imprimantei.

Fișierul de imprimantă conține atributele pentru coada de ieșire (OUTQ) și dispozitivul de tipărire (DEV), care dictează modul în care ieșirea imprimantă va condusă. Setările implicite sunt, de obicei *JOB, ceea ce înseamnă că atributele jobului pentru coada de ieșire și dispozitivul imprimantă determină modul cum este direcționată ieșirea imprimantei. Atributele job ale setărilor cozii de ieșire și dispozitivului imprimantă sunt bazate pe informațiile obținute la crearea unui job. Aceasta se bazează pe informațiile din profilul utilizator sub care rulează jobul, din descrierea de job, din descrierea dispozitivului de stație de lucru și din valoarea de sistem QPRTDEV (descriere dispozitiv imprimantă).

Când ieșirea imprimantă este gata să fie creată, sistemul verifică fișierul de imprimantă și atributele de job (în această ordine) pentru a vedea care coadă de ieșire va procesa ieșirea imprimantă și care dispozitiv de imprimantă va fi folosit de către sistem. Puteți modifica parametrii cozii de ieșire (OUTQ) și dispozitivului de imprimantă (DEV) în momentul în care jobul este lansat sau la momentul rulării jobului pentru a evita procesarea extinsă. De exemplu, utilizatorul poate seta coada de ieșire a fișierului de imprimantă cu o anumită coadă și poate seta dispozitivul de imprimantă cu imprimanta specifică din fișierul de imprimantă la inițializarea de job pentru ca modificările să aibă efect imediat. Când

face astfel, ieșirea imprimantei nu trebuie să parcurgă atributele de job pentru a găsi coada de ieșire și dispozitivul de tipărire pe care le va folosi. Dacă o coadă de ieșire specificată nu poate fi găsită, ieșirea imprimantă va fi condusă la QGPL/QPRINT. Pentru informații suplimentare despre modul cum este creată ieșirea imprimantei, vedeți Capitolul 1 din manualul Programarea dispozitivului de tipărire.

Fișierele de ieșire imprimantă sunt fișiere care dețin informații care așteaptă să fie tipărite sau procesate. Fișierul ieșire imprimantă deține atribute importante care definesc poziția ieșirii imprimantă în coadă față de altă ieșire imprimantă. Poziția este definită de atributele de prioritate, stare și planificare.

Coadă de ieșire

O **coadă de ieșire** este un obiect care conține o listă de fișiere ieșire imprimantă de scris la un dispozitiv de imprimantă. Coada de ieșire are atribute importante care determină ordinea în care este procesată ieșirea imprimantei și autorizarea necesară pentru a face modificări asupra fișierului de ieșire imprimantă.

Prioritate

Ieșirea imprimantei care așteaptă să fie procesată este mutată în coada de ieșire pe baza priorității sale (în intervalul 1-9 unde 1 este cea mai mare prioritate).

Stare Starea curentă a ieșirii imprimantei. Puteți vizualiza această stare din pagina Generală a ferestrei Proprietăți ieșire.

Planificare

Atributul de planificare menționează când fișierul ar trebui să pornească tipărirea fizică a datelor de ieșire.

Imediat

Tipărește imediat, chiar dacă fișierul de ieșire imprimantă nu este închis.

Sfârșit fișier (implicit)

Tipărirea începe imediat cum fișierul de ieșire imprimantă este închis.

Opre job

Tipărirea începe când jobul este oprit.

După ce fișierul de ieșire imprimantă este gata de a fi tipărit, un job scriitor, un job care procesează ieșirea imprimantei dintr-o coadă de ieșire până la dispozitivul de tipărire, ia datele din fișierul de ieșire a imprimantei și le trimite imprimantei desemnate.

Atributele unei cozi de ieșire

Coadă de ieșire controlează modul în care fișierele de ieșire imprimantă (de asemenea numite fișiere spool) sunt procesate și cine are autorizarea de a executa acțiuni în coada de ieșire și ieșirea imprimantă asociată.

Deoarece majoritatea informației pe care o tipăriți pe sistemul iSeries este creată ca ieșire de imprimantă, securitatea este necesară pentru împiedica accesul utilizatorilor neautorizați la materialele confidențiale sau sensibile. Dacă aveți autorizarea de verificare, autorizarea de date, controlul operator, controlul spool sau dacă sunteți proprietar, puteți să accesați și să faceți modificări într-o coadă de ieșire sau fișier de ieșire imprimantă. Trebuie să aveți una din următoarele autorizări pentru a executa orice acțiune într-o coadă de ieșire sau ieșire imprimantă:

Autorizare la verificare

Trebuie să fiți proprietarul cozii sau să aveți autorizație de date.

Afișare date

Când această autorizare este setată la *YES, vă permite să executați acțiuni precum vizualizare, mutare, trimitere ieșire către un alt sistem și copierea ieșirii imprimantă.

Control operator

Dacă acest atribut este setat la *YES, utilizatorii cu autorizare specială *JOBCTL sunt autorizați să execute acțiuni precum reținere, eliberare și ștergere ieșire imprimantă din coada de ieșire. Alte acțiuni pe ieșire imprimantă, cozi de ieșire, și scrieri sunt permise la fel de bine și sunt documentate în Referințe de securitate iSeries.

Control de spool

Permite utilizatorului să execute toate operațiile asupra ieșirii imprimantei. Utilizatorul trebuie să aibă autorizare *EXECUTE la biblioteca unde este localizată coada de ieșire pentru a executa orice acțiune asupra cozii de ieșire.

Proprietar

Aceasta permite utilizatorului care deține coada de ieșire să modifice sau să șteargă ieșirea imprimantă.

Notă: Autorizarea implicită la coada de ieșire este autorizarea publică *USE. Autorizarea Afișare date este setată la *NO (care înseamnă că nu oricine poate vizualiza ieșirea imprimantă). Autorizare pentru a verifica este *OWNER (astfel ca proprietarul cozii de ieșire să poată manipula ieșirea de imprimantă). Controlul operator este setat la *YES (care înseamnă că un utilizator cu *JOBCTL poate reține, elibera și șterge ieșirea imprimantă).

Pentru informații suplimentare despre autorizațiile necesare să lucreze cu cozi de ieșire, vedeți Anexa D din Manual de referințe de securitate.

Comandarea fișierelor

Ordinea fișierelor din coada atribut (SEQ) determină cum ieșirea de imprimantă va lăsa coada de ieșire să fie procesată.

Acest atribut are două valori:

- *FIFO: Coada este primul-sosit primul-plecat din cadrul priorității pentru fiecare fișier. Aceasta este, noile fișiere de spool sunt așezate după toate celelalte intrări din coadă cu aceeași prioritate.
- *JOBNBR : Intrările în coadă pentru fișierele de spool sunt sortate în secvență de prioritate folosind numărul de job (de fapt, data și ora la care jobul a intrat în sistem este folosită) al jobului care a creat fișierul de spool.

Notă: Puteți modifica doar ordinea în coada de ieșire a atributelor fișierelor când nici un fișier de ieșire imprimantă nu este în coadă.

Fișiere puse în spool

Punerea în spool este o funcție de sistem care salvează date pentru o procesare sau tipărire ulterioară. Aceste date sunt memorate într-un fișier pus în spool. Fișierele puse în spool funcționează într-o manieră asemănătoare fișierelor de bandă sau altor fișiere de dispozitiv. Fișierele puse în spool vă permit să vă gestionați datele destinate dispozitivelor atașate extern cum ar fi o imprimantă.

Funcțiile de punere în spool ajută utilizatorii de server să gestioneze operațiile de intrare sau ieșire mai eficient. Serverul suportă două tipuri de punere în spool, punere în spool de ieșire și de intrare. Punerea în spool de ieșire poate fi utilizată pentru dispozitive imprimante. Punerea în spool de intrare se aplică intrării de fișier bază de date.

Punerea în spool a ieșirii:

Punerea în spool a ieșirii poate fi utilizată pentru dispozitive imprimante sau cu dischetă. Punerea în spool a ieșirii trimite ieșirea jobului în spațiul de stocare în loc să îl trimită direct la un dispozitiv de ieșire imprimantă sau dischetă. Punerea în spool a ieșirii permite jobului care produce ieșirea să continue procesarea fără a lua în considerare viteza sau disponibilitatea dispozitivelor de ieșire.

În plus, punerea în spool a ieșirii permite serverului să producă ieșire pe mai multe dispozitive de ieșire, precum dispozitive imprimantă și dispozitive cu dischetă, într-o manieră eficientă. Face acest lucru trimițând ieșirea unui job, destinată unei imprimante, pe discul de stocare. Acest proces anulează o potențială limitare a jobului impusă de disponibilitatea sau viteza dispozitivelor de ieșire.

Principalele elemente ale punerii în spool a ieșirii sunt:

- **Descrierea dispozitivului:** O descriere a unui dispozitiv imprimantă.
- **Fișierul pus în spool:** Un fișier care conține înregistrări de ieșiri puse în spool care vor fi procesate pe un dispozitiv de ieșire.

- **Coadă de ieșire:** O listă ordonată de fișiere puse în spool.
- **Scriitor:** Un program care trimite fișierele dintr-o coadă de ieșire la un dispozitiv.
- **Programul de aplicație:** Un program cu limbaj de nivel înalt care creează un fișier pus în spool utilizând un fișier de dispozitiv cu atributul de punere în spool specificat ca fiind SPOOL(*YES).
- **Fișier de dispozitiv:** O descriere a formatului ieșirii și o listă de atribute care descriu modul cum serverul ar trebui să proceseze fișierul pus în spool.

Funcțiile de punere în spool a ieșirii sunt realizate de server fără a necesita operații speciale de către programul care produce ieșirea. Când un dispozitiv este deschis de către un program, sistemul de operare determină dacă ieșirea va fi pusă în spool. Când este deschis un fișier de imprimantă care specifică punerea în spool, fișierul pus în spool, care conține ieșirea programului, este pus în coada de ieșire corespunzătoare din server.

Un fișier pus în spool poate fi pus la dispoziție pentru tipărire când este deschis fișierul de imprimantă, când este închis fișierul de imprimantă, sau la sfârșitul jobului. Un scriitor de imprimantă este pornit în subsistemul depunere în spool pentru a trimite înregistrări către imprimantă. Fișierul pus în spool este selectat dintr-o coadă de ieșire.

Descrieri dispozitiv de punere în spool

Descrierile de dispozitiv trebuie să fie create pentru fiecare dispozitiv imprimantă și de dischetă pentru a defini acel dispozitiv pe server. Descrierile dispozitivului imprimantă sunt create utilizând comanda CRTDEVPR (Create Device Description for Printer - Creare descriere de dispozitiv pentru imprimantă); descrierile de dispozitive cu dischetă sunt create utilizând comanda CRTDEVDKT (Create Device Description for Diskette - Creare descriere dispozitiv pentru dischetă).

Redirecționarea fișierelor pentru fișiere puse în spool

Redirecționarea fișierelor are loc atunci când un fișier pus în spool este trimis la alt dispozitiv de ieșire decât cel pentru care a fost intenționat de fapt. Redirecționarea fișierelor poate implica dispozitive care procesează alte medii de stocare (precum o ieșire de imprimantă trimisă la un dispozitiv cu dischetă) sau dispozitive care procesează același tip de medii de stocare dar sunt tipuri diferite de dispozitive (precum o ieșire de imprimantă 5219 trimisă la o imprimantă 4224).

În funcție de noul dispozitiv de ieșire pentru fișierele puse în spool, fișierul poate fi procesat ca și cum ar fi fost pe dispozitivul specificat la început. Totuși, diferențele dintre dispozitive pot deseori cauza formatarea diferită a ieșirii. În aceste cazuri, expeditorul trimite un mesaj de interogare către coada de mesaje a scriitorului pentru a vă informa de situație și pentru a vă permite să specificați dacă doriți ca tipărirea să continue.

Cozile de ieșire și fișiere spool:

Procesarea joburilor interactive și batch poate avea ca rezultat înregistrări de ieșiri puse în spool care vor fi procesate pe un dispozitiv de ieșire, precum o imprimantă sau o unitate de dischetă. Aceste înregistrări de ieșiri sunt stocate în fișiere puse în spool până când vor putea fi procesate. Pot exista mai multe fișiere puse în spool pentru un singur job.

Când este creat un fișier spool, fișierul este pus într-o coadă de ieșire. Fiecare coadă de ieșire conține o listă ordonată de fișiere puse în spool. Un job poate avea fișiere puse în spool în una sau mai multe cozi de ieșire. Toate fișierele puse în spool dintr-o anumită coadă de ieșire ar trebui să aibă un set comun de atribute de ieșire, precum dispozitivul, tipul formularului și numărul de linii per inch. Utilizarea atributelor comune într-o coadă de ieșire reduce cantitatea de intervenție necesită și crește debitul dispozitivului.

În continuare vor fi listați unii din parametrii din comanda CRTOUTQ (Create Output Queue - Creare coadă de ieșire) și ceea ce ei specifică:

- **MAXPAGES:** Specifică dimensiunea maximă a fișierului pus în spool în numărul de pagini care sunt permise a fi tipărite între o oră de început și o oră de sfârșit din zi.
- **AUTOSTRWTR:** Specifică numărul de scriitori care sunt porniți automat la această coadă de ieșire.

- **DSPDTA:** Dacă utilizatorii care nu au o autorizare specială dar care au autorizare *USE pentru coada de ieșire pot afișa, copia sau trimite conținuturile fișierelor puse în spool altele decât cele proprii. Specificând *OWNER pentru DSPDTA, doar proprietarul fișierului sau un utilizator cu autorizare specială *SPLCTL poate afișa, copia sau trimite un fișier.
- **JOBSEP:** Câte pagini de separator de job, dacă există, vor fi tipărite între ieșirile fiecărui job când ieșirea este tipărită.
- **DTAQ:** Coadă de date asociată cu această coadă de ieșire. Dacă este specificat, o intrare este trimisă în coada de date de câte ori un fișier pus în spool intră în starea Pregătit în coadă.
- **OPRCTL:** Dacă un utilizator care are autorizare de control job poate controla coada de ieșire (de exemplu, dacă utilizatorul poate reține coada de ieșire).
- **SEQ:** Controlează ordinea în care fișierele puse în spool vor fi sortate în coada de ieșire.
- **AUTCHK:** Specifică ce tip de autorizare pentru coada de ieșire va permite unui utilizator să controleze fișierele puse în spool aflate într-o coadă de ieșire (de exemplu, poate permite unui utilizator să rețină fișierele puse în spool din coada de ieșire).
- **AUT:** Autorizare publică. Specifică ce control au utilizatorii asupra cozii de ieșire însăși.
- **TEXT:** Descriere text. Până la 50 caractere de text care descrie coada de ieșire.

Valori implicite ale cozilor de ieșire ale serverului:

Serverul este livrat cu valori implicite pe comenzi pentru a folosi cozile de ieșire implicite pentru imprimanta serverului la fel ca coada de ieșire implicită pentru toate ieșirile de spool. Imprimanta serverului este definită de către valoarea QPRTDEV a serverului.

Când un fișier de spool este creat prin deschiderea unui fișier dispozitiv și coada de ieșire specificată pentru fișier nu poate fi găsită, serverul va încerca să amplaseze fișierul de spool în coada de ieșire QPRINT în biblioteca QGPL. Dacă pentru oricare motiv fișierul de spool nu poate fi plasat în coada de ieșire QPRINT, un mesaj de eroare va fi trimis și ieșirea nu va fi pusă în spool.

Următoarele cozi de ieșire sunt livrate cu serverul:

- **QDKT:** Coadă de ieșire de dischetă implicită
- **QPRINT:** Coadă de ieșire de imprimantă implicită
- **QPRINTS:** Coadă de ieșire de imprimantă pentru forme speciale
- **QPRINT2:** Coadă de ieșire de imprimantă pentru hârtie 2-părți

Scriitor de punere în spool:

Un scriitor este un program i5/OS care ia fișierele puse în spool dintr-o coadă de ieșire și le produce pe un dispozitiv de ieșire. Fișierele puse în spool care au fost puse într-o anumită coadă de ieșire vor rămâne memorate în server până când este pornit un scriitor la coada de ieșire.

Scriitorul ia fișierele puse în spool câte unul pe rând din coada de ieșire, pe baza priorității lor. Scriitorul procesează un fișier pus în spool doar dacă intrarea sa în coada de ieșire indică faptul că acesta are este în starea pregătit (RDY). Puteți afișa starea unui anumit fișier pus în spool utilizând comanda WRKOUTQ (Work with Output Queue - Gestionare coadă de ieșire).

Dacă un fișier pus în spool este în starea pregătit, scriitorul ia intrarea din coada de ieșire și tipărește jobul specificat sau separatorii de fișier sau ambele, urmate de datele de ieșire din fișier. Dacă fișierul pus în spool nu este în starea pregătit, scriitorul lasă intrarea în coada de ieșire și trece la următoarea intrare. În cele mai multe cazuri scriitorul va continua să proceseze fișiere puse în spool (precedate de separatori de job și fișier) până când toate fișierele aflate în starea pregătit au fost luate din coada de ieșire.

Parametrul **AUTOEND** din comenzile de pornire scriitor determină dacă scriitorul continuă să aștepte ca noi fișiere puse în spool să devină disponibile pentru a fi scrise, dacă să se oprească după procesarea unui fișier sau să se oprească după ce toate fișierele puse în spool cu starea pregătit au fost luate din coada de ieșire.

Rezumatul comenzilor scriitorului de punere în spool:

Există multe comenzi pe care le puteți utiliza pentru a controla scriitori de punere în spool.

- **STRDKTWTR** (Start Diskette Writer - Pornire scriitor pe dischetă): Pornește un scriitor de punere în spool pe un dispozitiv cu dischetă specificat pentru a procesa fișierele spool pe acel dispozitiv.
- **STRPRTWTR** (Start Printer Writer - Pornire scriitor imprimantă): Pornește un scriitor de punere în spool pe un dispozitiv imprimantă specificat pentru a procesa fișierele spool pe acel dispozitiv.
- **STRRMTWTR** (Start Remote Writer - Pornire scriitor la distanță): Pornește un scriitor de punere în spool care trimite fișierele spool dintr-o coadă de ieșire către un server aflat la distanță.
- **CHGWTR** (Change Writer - Modificare scriitor): Vă permite să modificați unele din atributele scriitorului, cum ar fi tipul formularului, numărul de pagini de separator de fișier sau atributele cozii de ieșire.
- **HLDWTR** (Hold Writer - Reținere scriitor): Oprește un scriitor la sfârșitul unei înregistrări, la sfârșitul unui fișier pus în spool sau la sfârșitul unei pagini.
- **RLSWTR** (Release Writer - Eliberare scriitor): Eliberează un scriitor reținut anterior, pentru procesare suplimentară.
- **ENDWTR** (End Writer - Opreire scriitor): Oprește un scriitor de punere în spool și face disponibil pe server dispozitivul de ieșire asociat.

Notă: Puteți defini unele funcții pentru a furniza suport de punere în spool suplimentar. Surse de exemplu și documentație pentru comenzi, fișiere și programe pentru aceste funcții sunt parte din biblioteca QUSRTOOL, care este o parte instalată opțional a i5/OS.

Intrare în spool:

Intrarea în spool preia informațiile de la dispozitivul de intrare, pregătește jobul pentru planificare și pune o intrare într-o coadă de joburi. Utilizând spooling-ului de intrare, puteți scurta run time-ul jobului, crește numărul de joburi care pot rula secvențial și îmbunătăți debitul dispozitivului.

Elementele principale ale intrării în spool sunt:

- **Coadă de joburi:** O listă ordonată de joburi batch lansate în execuție pe server pentru a rula și din care sunt selectate joburi batch pentru a rula.
- **Cititor:** O funcție care preia joburile de la un dispozitiv de intrare sau de la un fișier bază de date și le pune într-o coadă de joburi.

Când un job batch este citit de la o sursă de intrare de către un cititor, comenzile din fluxul de intrare sunt memorate în server ca cereri pentru job, datele inline sunt puse în spool ca fișiere de date inline și o intrare pentru job este pusă într-o coadă de joburi. Informațiile jobului rămân memorate în server unde a fost pus de către cititor până ce este selectată intrarea de job din coada de joburi pentru a fi procesată de către un subsistem.

Pentru joburi cu fluxuri de intrare mici, este posibil să îmbunătățiți performanța serverului dacă nu folosiți intrarea în spool. Comanda SBMJOB (Submit Job - Lansare job) citește fluxul de intrare și pune jobul în coada de joburi din sistemul corespunzător, ocolind subsistemul de punere în spool și operațiile cititorului.

Dacă jobul dumneavoastră necesită citirea unui flux de intrare mare, ar trebui să utilizați intrarea în spool (comanda STRDKTRDR (Start Diskette Reader - Pornire cititor dischetă) sau STRDBRDR (Start Database Reader - Pornire cititor bază de date)) pentru a permite jobului să aibă intrarea independentă de când anume este de fapt jobul procesat.

Rezumatul comenzilor de intrare job:

Următoarele comenzi pot fi folosite la lansarea joburilor pe server. Comenzile de pornire cititor pot fi utilizate pentru punerea în spool a intrării de job; comenzile de lansare job nu utilizează punerea în spool. Pentru descrieri detaliate ale acestor comenzi, vedeți subiectul CL din Centrul de informare iSeries.

- Job batch (BCHJOB): Marchează începutul unui job într-un flux de intrare batch și definește caracteristicile de operare ale jobului.
- Date (DATA): Marchează începutul unui fișier de date inline.
- Oprire job batch (ENDBCHJOB): Marchează sfârșitul unui job într-un flux de intrare batch.
- Oprire intrare (ENDINP): Marchează sfârșitul unui flux de intrare batch.
- Lansare joburi de bază de date (SBMDBJOB): Citește un flux de intrare dintr-un fișier bază de date și pune joburile din fluxul de intrare în cozile de joburi corespunzătoare.
- Lansare joburi de pe dischetă (SBMDKTJOB): Citește un flux de intrare de pe dischetă și pune joburile din fluxul de intrare în cozile de joburi corespunzătoare.
- Pornire cititor de bază de date (STRDBRDR): Porneste un cititor pentru a citi un flux de intrare dintr-un fișier bază de date și pune jobul din fluxul de intrare în coada de joburi corespunzătoare.
- Pornire cititor dischetă (STRDKTRDR) : Porneste un cititor pentru a citi un flux de intrare de pe dischetă și pune jobul din fluxul de intrare în coada de joburi corespunzătoare.

Utilizarea unui fișier de date inline:

Un fișier de date inline este un fișier de date care este inclus ca parte a unui job batch atunci când jobul este citit de un cititor sau de către o comandă de lansare joburi. Utilizați SBMDBJOB sau STRDBRDR pentru a pune în coada un flux batch CL (flux de comenzi CL pentru a fi executate sau rulate). Acel flux batch CL poate include date care să fie puse în fișiere "temporare" (fișiere inline). Când un job se termină, fișierele inline sunt șterse.

Un fișier de date inline este delimitat în job de către o comandă //DATA la începutul fișierului și de un delimitator sfârșit-de-date la sfârșitul fișierului.

Delimitatorul sfârșit-de-date poate fi un șir de caractere definit de utilizator sau implicit // . // trebuie să apară în pozițiile 1 și 2. Dacă datele dumneavoastră conțin // în pozițiile 1 și 2, ar trebui să utilizați un set unic de caractere precum: // *** END OF DATA Pentru a specifica acesta ca fiind un delimitator de sfârșit-de-date unic, parametrul **ENDCHAR** din comanda //DATA ar trebui codat astfel:

```
ENDCHAR('// *** END OF DATA')
```

Notă: Fișierele de date inline pot fi accesate doar în timpul primului pas de rutare a jobului batch. Dacă un job batch conține o comandă TFRJOB (Transfer Job - Transfer job), RRTJOB (Reroute Job - Rerutare job) sau TFRBCHJOB (Transfer Batch Job - Transfer job batch), fișierele de date inline nu pot fi accesate în noul pas de rutare.

Un fișier de date inline poate fi fie denumit fie nedenumit. Pentru un fișier de date inline nedenumit, fie este specificat QINLINE ca numele fișierului din comanda //DATA sau se specifică fără nume. Pentru un fișier de date inline denumit, se specifică un nume de fișier.

Un fișier de date inline denumit are următoarele caracteristici:

- Are un nume unic într-un job. Nici un alt fișier de date inline nu poate avea același nume.

- Poate fi utilizat de mai multe ori într-un job.
- De fiecare dată când este deschis, el este poziționat în prima înregistrare.

Pentru a utiliza un fișier de date inline denumit, trebuie fie să specificați numele fișierului în program sau să utilizați o comandă de suprascriere pentru a modifica numele fișierului specificat în program cu numele fișierului de date inline. Fișierul trebuie să fie deschis doar pentru intrare.

Un fișier de date inline nedenumit are următoarele caracteristici:

- Numele său este QINLINE. (Într-un job batch, toate fișierele de date inline nedenumite primesc același nume.)
- El poate fi utilizat doar o dată într-un job.
- Când sunt incluse mai multe fișiere de date inline nedenumite într-un job, fișierele trebuie să fie în fluxul de intrare în aceeași ordine ca atunci când sunt deschise.

Pentru a utiliza un fișier de date inline nedenumit, faceți una din următoarele:

- Specificați QINLINE în program.
- Utilizați o comandă de suprascriere fișier pentru a modifica numele de fișier specificat în program cu QINLINE.

Dacă limbajul dumneavoastră de nivel înalt necesită nume de fișier unice într-un program, puteți utiliza QINLINE ca nume de fișier doar o singură dată. Dacă aveți nevoie să utilizați mai mult de un fișier de date inline nedenumit, puteți utiliza o comandă de suprascriere fișier în program pentru a specifica QINLINE pentru filiere de date inline nedenumite suplimentare.

Notă: Dacă rulați comenzi condițional și procesați mai mult de un fișier de date inline nedenumit, rezultatele nu pot fi precise dacă este utilizat fișierul de date inline nedenumit greșit.

Considerente deschise pentru fișiere de date inline:

Următoarele considerente se aplică la deschiderea fișierelor de date inline:

- Lungimea înregistrării specifică lungimea înregistrărilor de intrare. (Lungimea înregistrării este opțională.) Când lungimea înregistrării depășește lungimea datelor, este trimis un mesaj programului dumneavoastră. Datele sunt completate cu blancuri. Când lungimea înregistrării este mai mică decât lungimea datelor, înregistrările sunt trunchiate.
- Când este specificat un fișier într-un program, serverul caută fișierul ca pe un fișier de date inline denumit, înainte să caute fișierul într-o bibliotecă. De aceea, dacă un fișier de date inline denumit are același nume ca și fișierul care nu este un fișier de date inline, fișierul de date inline este întotdeauna utilizat, chiar dacă numele fișierului este calificat de un nume de bibliotecă.
- Fișierele de date inline denumite pot fi partajate între programele din același job specificând SHARE(*YES) într-o comandă de creare fișier sau suprascriere fișier. De exemplu, dacă o comandă de suprascriere fișier specificând un fișier numit INPUT și SHARE(*YES) este într-un job batch cu un fișier de date inline numit INPUT, toate programele care rulează în job, care specifică numele de fișier INPUT vor partaja același fișier de date inline denumit. Fișierele de date inline nedenumite nu pot fi partajate între programele din același job.
- Când utilizați fișiere de date inline, ar trebui să vă asigurați că este specificat tipul corect de fișier în comanda //DATA. De exemplu, dacă fișierul va fi utilizat ca un fișier sursă, tipul fișierului din comanda //DATA trebuie să fie sursă.
- Fișierele de date inline trebuie să fie deschise doar pentru intrare.

Istoricul joburilor

Un istoric de joburi conține informații referitoare la cererile introduse pentru un job. Un istoric de job are două formulare, un formular de așteptare și unul pus în spool.

În formularul său de așteptare, un istoric de job pentru un job terminat se poate modifica pe măsură ce alte joburi (subsistem, operator sistem și așa mai departe) interacționează cu el. În formularul pus în spool, un istoric de job este

un instantaneu (un moment de timp) și nu se modifică (precum fișierele puse în spool care sunt create de comanda DSPJOBLOG (Display Job Log - Afișare istoric job), sau create după ce jobul își termină activitatea).

Fiecare job are un istoric de job asociat care poate conține următoarele informații pentru job:

- Comenzile din job
- Comenzile dintr-un program CL (dacă programul CL a fost creat cu opțiunea LOG(*YES) sau cu opțiunea LOG(*JOB) și a fost rulată o comandă CHGJOB (Change Job - Modificare job) cu opțiunea LOGCLPGM(*YES))
- Toate mesajele (mesajul și textul de ajutor pentru mesaj) trimise la solicitant și care nu au fost înlăturate din coada de mesaje a programului

La sfârșitul jobului, istoricul de job poate fi scris în fișierul pus în spool QPJOBLOG așa încât să poată fi tipărit. Totuși, producerea unui istoric de job nu înseamnă neapărat tipărirea lui sau crearea unui fișier pus în spool. (De exemplu, API-ul QMHCTLJL (Control Job Log - Control istoric job) poate fi utilizat pentru a specifica faptul că istoricul de job va fi scris într-un fișier de ieșire la terminarea jobului.)

- | Puteți reduce numărul de istorice de joburi produse și puteți reduce conflictul pentru resurse (precum cozile de ieșire).
- | Aceasta reduce conflictul pentru resurse cauzat de producerea istoricelor de joburi.

Cum sunt create istoricele de joburi

Ediția V5R4 aduce istoricele de job în zona "la cerere". Istorigele de job sunt disponibile atunci când sunt necesare dar nu se face nici un efort pentru a produce istorice de joburi de care nu este nevoie.

Parametrul **LOG** are trei elemente: nivelul de mesaj (sau de înregistrare în istoric), gravitatea mesajului și nivelul de text de mesaj. Fiecare din aceste elemente au valori anumite care atunci când sunt combinate determină cantitatea și tipul de informație trimisă în istoric, de către job.

De exemplu, valoarea *NOLIST a elementului Text face să nu fie produs nici un istoric de job dacă jobul se termină anormal. (Istorigul de job nu intră în starea de așteptare.) Dacă jobul se termină anormal (dacă codul de terminare al jobului este 20 sau mai mare), este produs un istoric de job. Mesajele care apar în istoricul de job conțin atât textul de mesaj cât și ajutorul mesajului.

Puteți controla ceea ce produce istoricul de job. Aceasta se face cu parametrul **LOGOUTPUT**. Când un job se termină, are loc una din cele trei acțiuni care afectează modul cum este creat istoricul de job. Următoarele sunt valori ale parametrului **LOGOUTPUT**:

- **Serverul de istorice de job produce istoricul de job:** (*JOBLOGSVR)
- **Jobul însuși produce istoricul de job:** Dacă jobul nu își poate produce propriul istoric de job, istoricul de job va fi produs de un server de istorice de joburi. (*JOBEND)
- **Istorigul de job nu este produs:** Istorigul de job rămâne în starea de așteptare până când este înlăturat. (*PND)

Notă: Aceste valori nu afectează istoricele de job care sunt produse când coada de mesaje este plină și acțiunea cozii de mesaje a jobului specifică *PRTWRAP. Mesajele din coada de mesaje a jobului sunt scrise într-un fișier spool, din care poate fi tipărit istoricul de job, doar dacă nu a fost utilizat în job API-ul QMHCTLJL (Control Job Log Output - Control ieșire istoric de job) pentru a specifica faptul că mesajele din istoricul de job vor fi scrise într-un fișier bază de date.

Ce controlează parametrii istoricului de job?

Când un job pornește, el își obține valoarea **LOGOUTPUT** din descrierea jobului. Dacă descrierea de job specifică *SYSVAL (valoarea implicită pentru CRTJOB), jobul utilizează valoarea ieșirii istoricului de job care este specificată în valoarea de sistem QLOGOUTPUT pentru ieșire istoric job. (Deși valoarea livrată pentru valoarea de sistem QLOGOUTPUT (ieșire istoric job) este *JOBEND, valoarea recomandată este *JOBLOGSVR.) După ce jobul și-a stabilit atributul **LOGOUTPUT**, orice modificări aduse descrierii jobului sau valorii de sistem nu afectează jobul activ. Modificările asupra valorii de sistem sau asupra descrierii jobului intră în vigoare pentru joburile care intră în sistem după modificare.

Puteți utiliza comanda CHGJOB (Change Job - Modificare job) sau API-ul QWTCHGJB pentru a modifica atributul de job **LOGOUTPUT** după ce acesta a fost deja setat în job. Modificările asupra jobului au efect imediat.

Indiferent de metoda pe care o alegeți, opțiunile pentru tratarea istoricelor de job sunt aceleași. Puteți seta jobul să nu producă un istoric de job (*PND), puteți face ca jobul să producă istoricul de job (*JOBEND) sau puteți face ca serverul de istorice de job să producă istoricul de job (*JOBLOGSVR).

Istoric job în așteptare

Starea istoric job în așteptare a fost disponibilă mulți ani. Când un atribut al istoricului unui job este *PND, nu este produs nici un istoric de job. O dată cu lansarea ediției V5R4 au fost aduse îmbunătățiri atât Navigatorului iSeries și cât și interfeței bazate pe caractere, astfel încât puteți acum controla cum și în ce circumstanțe va fi produs istoricul unui anumit job.

Această nouă caracteristică este utilă atunci când plasați sistemul într-o stare restricționată. Când sistemul intră într-o stare restricționată, subsistemele se opresc și este posibil să se încheie o mie de joburi o dată. Aceasta în schimb poate crea o încărcare mare a resurselor de ieșire. Prevenind producerea acestor istorice de job, puteți reduce semnificativ impactul asupra acestor resurse.

Alt exemplu de când puteți utiliza această nouă caracteristică este în timpul eșecului de comunicații. Poate există multe joburi similare care produc același mesaj de eroare a istoricului de job. Puteți seta istoricul jobului să nu producă un fișier spool pentru toate joburile. Atunci, dacă se întâmplă să existe un eșec de comunicații, puteți utiliza comanda (WRKJOBLOG) (Work with Job Log - Gestionare istoric de job) pentru a determina ce istorice să fie tipărite. Puteți, de asemenea, utiliza ecranul (WRKJOBLOG) (Work with Job Logs - Gestionare istoric de job) pentru a gestiona istorice de job.

Joburile pot fi într-o stare de istoric job în așteptare din cauza lucrărilor realizate de comanda PWRDWNSYS (Power Down System - Oprire sistem din alimentare). Interfața cu utilizatorul a Navigatorului iSeries arată starea "Terminat - Istoric job în așteptare" pentru aceste joburi. Acesta este un subset al stării interfeței bazate pe caractere a *OUTQ.

Dacă veți profita de aceste îmbunătățiri ele vă vor ajuta să reduceți numărul de istorice de job produse și de aceea veți reduce conflictul pentru resurse. Aceasta poate duce la o performanță îmbunătățită a sistemului.

Server de istorice de joburi

În mod obișnuit, serverul de istorice de joburi scrie istoricul de job al unui job într-un fișier pus în spool. Puteți ruta istoricul de joburi către o imprimantă sau către un outfile, (dacă este specificat să faceți astfel utilizând API-ul QMHCTLJL (Control job log - Control istoric joburi)), totuși nu este metoda recomandată pentru producerea istoricelor de joburi.

Puteți vizualiza informații despre serverul de istorice de joburi prin Navigator iSeries din ecranul **Control funcționare** → **Joburi de server** sau ecranul **Control funcționare** → **Joburi active**. (Pentru a ușura identificarea joburilor care rulează pe serverul de istorice de joburi, asigurați-vă că includeți coloana Server în ecranul dumneavoastră.)

Numărul maxim de servere de istorice de joburi care pot fi active la un moment dat este 30. Porniți servere suplimentare de istorice de joburi și le gestionați în același mod ca și pe celelalte servere din sistemul dumneavoastră. Aceasta se face utilizând comanda de interfață bazată pe caracter STRLOGSVR.

Cum pornește serverul de istorice de joburi

În mod implicit, serverul de istorice de joburi va porni automat când pornește subsistemul QSYSWRK. Serverul se oprește de câte ori este oprit subsistemul QSYSWRK.

Comanda STRLOGSVR (Start Job Log Server - Pornire server de istorice de job) pornește serverul de istorice de joburi. Serverul de istorice de joburi scrie istorice de joburi pentru joburile al căror istoric de job se află în stare de așteptare și nu au atributul *PND. Serverul de istorice de joburi scrie istoricul de job al unui job fie într-un fișier pus în spool, la o imprimantă sau la un outfile, (dacă este specificat să se facă astfel utilizând API-ul QMHCTLJL de Control istoric de joburi).

Caracteristici istoric job

Navigatorul iSeries vă furnizează o interfață cu utilizatorul prietenoasă, ușor de citit, de la care puteți vizualiza istoricele de joburi și mesajele lor. Puteți, de asemenea, vizualiza istorice de job utilizând interfața bazată pe caractere.

Puteți controla ce coloane să apară în lista de istorice de joburi utilizând fereastra Istoric job - Coloane. (**Control funcționare** → **Joburi active** → **Faceți clic dreapta pe un job și selectați Istoric job** → **Vizualizare meniu** → **Personalizarea acestei vizualizări** → **Coloane**) Coloanele pe care alegeți să le afișați din lista de istorice de joburi sunt:

ID mesaj	Din program
Mesaj	Nivel cerere
Trimis	Gravitate
Fir de execuție	Către program
Tip	

Interfața bazată pe caractere

Când utilizați comanda DSPJOBLOG (Display Job Log - Afișare istoric job), vedeți ecranul Istoric job. Acest ecran arată nume de programe cu simboluri speciale, după cum urmează:

>>	Comanda care rulează sau următoarea comandă care va fi rulată. De exemplu, dacă a fost apelat un program CL sau un program cu limbaj de nivel înalt, este arătat apelul programului.
>	Comanda a terminat procesarea.
..	Comanda nu a fost încă procesată.
?	Mesaj de răspuns. Acest simbol marchează atât acele mesaje care necesită un răspuns cât și pe acele mesaje cărora li s-a răspuns.

Anteturi de istoric job:

Anteturile de istoric job sunt localizate la începutul fiecărei pagini a istoricului jobului tipărit. Aceste anteturi identifică jobul cu fiecare din aplicațiile istoricului jobului și cu caracteristicile fiecărei intrări. Următoarea este o listă a posibilelor intrări în antetul istoricului jobului.

- Numele complet calificat al jobului (nume job, nume utilizator, și numărul jobului)
- Numele descrierii de job folosit pentru a porni jobul
- Data și ora la care jobul a pornit
- Identificatorul mesajului
- Tipul mesajului
- Gravitatea mesajului
- Data și ora la care fiecare mesaj a fost trimis
- Mesajul. Dacă nivelul de înregistrare specifică că textul de nivel secund este a fi inclus, textul de nivel secund apare pe liniile următoare sub mesaj
- Programul din care mesajul sau cererea a fost trimisă
- Numărul de instrucțiune al interfeței mașinii sau offset-ul la programul la care mesajul a fost trimis

Notă: Numerele de instrucțiune ale interfeței mașinii apar numai pentru scăpare, notificare, și mesaje de diagnostică. Pentru celelalte tipuri de mesaj, numărul de instrucțiune al interfeței mașinii este setat la zero.

- Dacă jobul utilizează APPC, antetul conține o linie care arată unitatea de identificator de lucru pentru APPC.

Mesajele:

Mesajele conțin numele jobului, tipul mesajului, data și ora la care a fost trimis, acțiunea care apare și acțiunile necesare pentru a corecta o problemă. Acesta este util atunci când încercați să depanați problemele care pot surveni pe servere. Puteți accesa istorice de joburi, pentru joburile de server, prin Navigatorul iSeries. Mesajele se încadrează în două categorii, mesaje de alertă și mesaje înregistrate într-un istoric de job.

Mesaje de alertă - Aceste mesaje sunt trimise la QSYSOPR deoarece ei au nevoie de acțiune imediată. Mesajul conține problema, cauza, și acțiunile de recuperare necesare. De exemplu, serverul eșuează să înceapă sau să sfârșească serverul dintr-o dată. Unele servere trimit mesaje de alertă la QSYSOPR. Aceste mesaje au definită opțiunea de alertă (ALROPT) în descrierea mesajului. Puteți folosi alertele ca să furnizeze centralizat manipulând mesaje de alertă.

Mesajele înregistrate într-un istoric al jobului - Aceste mesaje sunt diagnosticate în natură, semnificând că nu sunt critice dar alertează utilizatorul despre unele acțiuni care au fost luate. Aceste mesaje pot fi sisteme generate precum utilizatorul creat.

Nivelul de înregistrare în istoric a mesajelor

Nivelul înregistrării în istoric a mesajelor determină ce mesaje și ce tipuri de mesaje ar trebui să fie înregistrate în istoric pentru job. Următoarea este o explicație a ceea ce reprezintă fiecare nivel.

Nivelul 1	Toate mesajele trimise cozii de mesaje externe a jobului cu o gravitate mai mare sau egală cu valoarea de gravitate a mesajului. (În Navigatorul iSeries, valoarea de Gravitate a mesajului (0-99) poate fi găsită în fereastra Proprietăți job - Istoric job. Aceasta este o valoare pe care o puteți controla.)
Nivelul 2	Toate mesajele care întrunesc calificările Nivelului 1 și toate mesajele de cerere care rezultă într-un mesaj de nivel înalt mai mare sau egal cu valoarea de gravitate a mesajului. Notă: Un mesaj de nivel înalt este un mesaj care este trimis cozii de mesaje a programului care recepționează mesajul de cerere. (De exemplu, QCMD este un program de procesare a cererilor livrat de IBM care recepționează mesaje de cerere.)
Nivelul 3	Toate mesajele care întrunesc calificările de Nivel 1 sau Nivel 2 și toate mesajele de cerere. În plus, toate comenzile din programele CL sunt incluse dacă este bifat Comenzi istoric din caseta Programe CL (Proprietăți job - fereastra Istoric job). Notă: Comenzi istoric din caseta Programe CL este echivalent cu atributul de istoric al programului CL.
Nivelul 4	Toate mesajele de cerere și toate mesajele cu o gravitate mai mare sau egală cu gravitatea de înregistrare în istoric a mesajelor, inclusiv mesajele de urmărire. În plus, toate comenzile din programele CL sunt incluse dacă este bifată Comenzi istoric din caseta Programe CL Proprietăți job - fereastra Istoric job). Notă: Comenzi istoric din caseta Programe CL este echivalent cu atributul de istoric al programului CL.

Istoric de joburi interactive

Descrierile de job livrate de IBM QCTL, QINTER și QPGMR toate au un nivel de istoric de LOG(4 0 *NOLIST); de aceea toate textele de mesaj de ajutor sunt scrise în istoricul jobului. Totuși, istoricele de job nu sunt tipărite dacă jobul se oprește normal decât dacă specificați *LIST în comanda SIGNOFF.

Dacă un utilizator de stație de afișare folosește un meniu livrat de IBM sau ecranul de Introducere comenzi, sunt afișate toate mesajele de eroare. Dacă utilizatorul de stație de afișare utilizează un program inițial scris de utilizator, toate mesajele nemonitorizate fac ca programul inițial să se termine și să fie produs un istoric de job. Totuși, dacă programul inițial monitorizează mesaje, el primește controlul atunci când este recepționat un mesaj. În acest caz, este important să vă asigurați că este produs istoricul de job astfel încât să puteți determina respectiva eroare survenită.

De exemplu, presupuneți că programul inițial afișează un meniu care include o opțiune de anulare semnare, care are valoarea implicită *NOLIST. programul inițial monitorizează toate excepțiile li include o comandă CHGVAR (Change Variable - Modificare variabilă) care modifică opțiunea de anulare semnare cu *LIST dacă are loc o excepție:

```
PGM
DCLF MENU
DCL &SIGNOFFDPT TYPE(*CHAR) LEN(7)
VALUE(*NOLIST)
.
```

```

.
.
MONMSG MSG(CPF0000) EXEC(GOTO ERROR)
PROMPT: SNDRCVF RCDfmt (PROMPT)
CHGVAR &IN41 '0'
.
.
.
IF (&OPTION *EQ '90') SIGNOFF
LOG(&SIGNOFFOPT);
.
.
.
GOTO PROMPT
ERROR: CHGVAR&SIGNOFFOPT '*LIST'
CHGVAR &IN41 '1'
GOTO PROMPT
ENDPGM

```

Dacă are loc o excepție, comanda CHGVAR modifică opțiunea din comanda SIGNOFF cu *LIST și setează un indicator. Acest indicator poate fi utilizat pentru a condiționa o constantă care afișează un mesaj care explică faptul că a survenit o eroare neașteptată și care spune utilizatorului stației de afișare ce să facă.

Istoric sistem QHST

Istoricul de sistem (QHST) conține o coadă de mesaje și un fișier fizic cunoscut ca o versiune de istoric. Mesajele trimise cozii de mesaje a istoricului sunt scrise de sistem în fișierul fizic al versiunii de istoric curente.

Istoricul sistemului (QHST) conține o urmă de nivel înalt de activități de sistem precum mesajele de sistem, subsistem, informații de job, stare dispozitiv și operator de sistem. Coada sa de mesaje este QHST.

Versiune istoric

Fiecare versiune de istoric este un fișier fizic care este numit în următorul mod:

Qxxxxydddn

Unde:

xxx este o descriere de 3 caractere a tipului de istoric (HST)

yyddd este data Iuliană în care a fost creată versiunea de istoric

n este un număr de ordine din data Iuliană (de la 0 la 9 sau de la A la Z)

Când o versiune de istoric este plină, o nouă versiune de istoric este creată automat.

Notă: Numărul de înregistrări din versiunea de istoric a istoricului de sistem este specificat în valoarea de sistem QHSTLOGSIZ (maxim de înregistrări din istoricul de sistem). Această valoare de sistem suportă, de asemenea, o opțiune *DAILY care creează o versiune nouă în fiecare zi.

Formatul istoricului sistemului:

Pentru a memora mesajul trimis unui istoric de sistem este utilizat un fișier bază de date. Deoarece toate înregistrările dintr-un fișier fizic au aceeași lungime și mesajele trimise unui istoric au lungimi diferite, mesajele se pot extinde la mai mult de o înregistrare. Fiecare înregistrare pentru un mesaj are trei câmpuri:

- Data și ora sistemului (un câmp de caractere de lungimea 8). Acesta este un câmp intern. De asemenea, data și ora convertite sunt tot în mesaj.
- Număr înregistrare (un câmp de 2 octeți). De exemplu, câmpul conține hex 0001 pentru prima înregistrare, hex 002 pentru a doua înregistrare și așa mai departe.

- Date (un câmp de caractere cu lungimea 132).

Formatul pentru al treilea câmp (date):

Tabela 1. Formatul pentru Al treilea câmp al Primei înregistrări

Conținuturi	Tip	Lungime	Poziții din înregistrare
Nume job	Caracter	26	11-36
Data și ora convertite	Caracter	13	37-49
ID mesaj	Caracter	7	50-56
Nume fișier mesaje	Caracter	10	57-66
Nume bibliotecă	Caracter	10	67-76
Tip mesaj	Caracter	2	77-78
Cod de gravitate	Caracter	2	79-80
Nume program trimitere	Caracter	12	81-92
Nume program recepție	Caracter	10	97-106
Număr instrucțiune program recepție	Caracter	4	107-110
Lungime text mesaj	Binar	2	111-112
Lungime date mesaj	Binar	2	113-114
Rezervat	Caracter	28	115-142

Tabela 2. Formatul câmpului al treilea (date) din înregistrările rămase

Conținuturi	Tip	Lungime
Mesaj	Caracter	Variabilă (Această lungime este specificată în prima înregistrare (pozițiile 111 și 112) și nu poate depăși 132.)
Date mesaj	Caracter	Variabilă (Această lungime este specificată în prima înregistrare (pozițiile 113 și 114).)

Un mesaj nu este niciodată divizat când este pornită o nouă versiune de istoric. Prima și ultima înregistrare a unui mesaj sunt întotdeauna în aceeași versiune QHST.

Procesare fișier QHST

Dacă utilizați un program de limbaj înalt pentru a procesa fișierul QHST, țineți minte că datele de mesaj încep la o locație variabilă pentru fiecare utilizare a aceluiași mesaj. Motivul pentru acest lucru este că acest mesaj include variabile înlocuibile astfel încât lungimea reală a mesajului variază.

Totuși, pentru mesajul CPF1124 (pornire job) și mesajul CPF1165 (terminare job) datele de mesaj încep întotdeauna în poziția 11 a celei de a treia înregistrări.

Informații despre performanță și QHST:

Informațiile despre performanță nu sunt afișate ca text în mesajul CPF1164. Deoarece mesajul nu este în istoricul QHST, utilizatorii pot scrie programe de aplicații pentru a extrage aceste date.

Informațiile de performanță sunt transmise ca o valoare de text de înlocuire de lungime variabilă. Aceasta înseamnă că datele sunt într-o structură din prima intrare care este lungimea datelor. Dimensiunea câmpului lungime nu este inclusă în lungime.

Ora și data: Primele câmpuri de date din structură sunt orele și datele când jobul a intrat în sistem și când a fost pornit primul pas de rutare pentru job. Orele sunt în formatul 'oo:mm:ss'. Separatorii de oră în acest exemplu sunt două puncte. Acest separator este determinat de valoarea specificată în valoarea de sistem pentru dată și oră, QTIMSEP. Data este într-un format definit în valoarea de sistem pentru dată și oră QDATFMT și separatorii sunt în valoarea de sistem QDATSEP. Ora și data când jobul a intrat în sistem este precedentă în structură orei și datei de pornire a jobului. Ora și data la care jobul a intrat în sistem sunt atunci când sistemul devine conștient că un job trebuie inițiat (o structură de job este pusă de-o parte pentru job). Pentru un job interactiv, ora de intrare a jobului este ora la care parola a fost recunoscută de către sistem. Pentru un job batch, este ora la care este procesată comanda BCHJOB (Batch Job - Job batch) sau comanda SBMJOB (Submit Job - Lansare job). Pentru un job monitor, cititor sau scriitor, este ora la care este procesată comanda de pornire corespunzătoare iar pentru joburi autostart este în timpul pornirii subsistemului.

Timpul total de răspuns și Numărul de tranzacții: În continuarea orelor și datelor sunt timpul de răspuns și numărul de tranzacții. Timpul total de răspuns este în secunde și conține valoarea acumulată a tuturor intervalelor în care jobul era în curs se procesare, între apăsarea tastei Enter de la stația de lucru și momentul când este afișat următorul ecran. Această informație este similară cu aceea arătată în ecranul WRKACTJOB (Work with Active Jobs - Gestionare joburi active). Acest câmp are sens doar pentru joburile interactive.

Este, de asemenea, posibil în cazul unui eșec al sistemului sau unei opriri de job anormale ca ultima tranzacție să nu fie inclusă în total. Codul de oprire job în acest caz ar fi 40 sau mai mare. Numărul tranzacției are, de asemenea, sens, doar pentru joburile interactive în afara de joburile de consolă și reprezintă numărul de intervale de timp de răspuns numărate de sistem în timpul jobului.

Numărul de operații auxiliare sincrone de I/E: Numărul de operații de I/E auxiliare sincrone urmează după numărul de tranzacții. Pentru un job cu mai multe fire de execuție, această valoare include doar operațiile de I/E auxiliare sincrone pentru firul de execuție inițial. Acesta este la fel cu câmpul AUXIO care apare în ecranul WRKACTJOB cu excepția următoarelor diferențe:

- Ecranul WRKACTJOB afișează valoarea pentru firul de execuție inițial al pasului de rutare curent.
- Mesajul QHST conține totalul cumulat pentru firul de execuție inițial al fiecărui pas de rutare din job.

Dacă jobul se termină cu un cod de sfârșit 70, această valoare este posibil să nu conțină numărul pentru pasul de rutare final. În plus, dacă un job există peste un IPL (utilizând o comandă TFRBCHJOB (Transfer Batch Job - Transfer job batch) el este oprit înainte de a deveni activ în urma unui IPL, valoarea este 0.

Fișiere spool

Un fișier spool reține datele de ieșire până când acestea pot fi tipărite. Fișierul spool colectează date de la un dispozitiv până când un program sau un dispozitiv este capabil să proceseze datele. Un program utilizează un fișier spool ca și când ar citi sau ar scrie la un dispozitiv real. Acesta este spooling de intrare și ieșire.

Spooling de intrare se face de către sistem pentru fișiere bază de date sau pe dischetă. În subsistemul de spooling este pornit un program livrat de IBM, numit cititor, care citește fluxurile jobului batch de la dispozitiv și pune joburile într-o coadă de joburi.

Spooling de ieșire se realizează pentru imprimante. În subsistemul de spooling este pornit un program livrat de IBM, numit scriitor de imprimantă, care selectează fișiere spool din coada sa de ieșire și scrie la imprimantă înregistrările din fișierul spool de ieșire.

La sfârșitul unui job, istoricul jobului poate fi scris în fișierul spool QPJOBLOG, astfel încât acesta să poată fi tipărit.

Contabilizarea jobului

Funcția de contabilizare job adună date astfel încât să puteți determina cine vă utilizează sistemul și ce resurse de sistem utilizează. De asemenea, ea vă asistă la evaluarea utilizării generale a sistemului dumneavoastră. Contabilizarea jobului este opțională. Trebuie să faceți anumiți pași pentru a seta contabilizarea jobului. Puteți cere sistemului să adune date despre contabilizarea resurselor jobului, date despre contabilizarea fișierelor de imprimantă sau ambele. Puteți, de asemenea, alocă coduri de contabilizare profilurilor utilizator sau anumitor joburi.

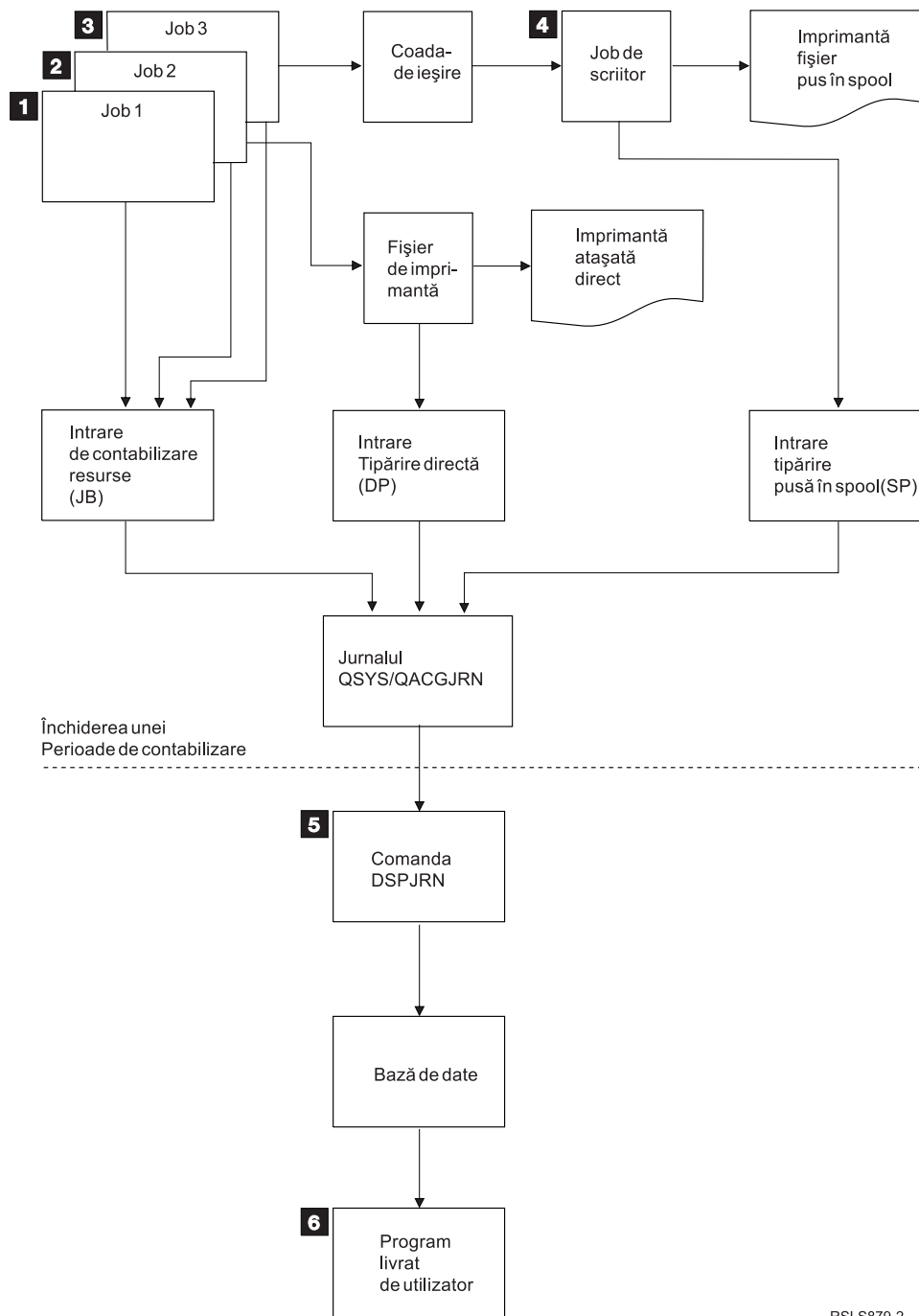
Datele tipice de contabilizare job vă detaliază joburile care rulează în sistemul dumneavoastră și resursele pe care le utilizează precum utilizarea unității de procesare, a imprimantei, a stațiilor de afișare, a bazei de date și a funcțiilor de comunicații.

Statisticile de contabilizare job sunt păstrate utilizând intrările în jurnal făcute de jurnalul de contabilizare a sistemului QSYS/QACGJRN. Ar trebui să știți cum să realizați operațiile de gestionare jurnal, cum ar fi salvarea unui receptor de jurnal, modificarea receptorilor de jurnal și ștergerea vechilor receptori de jurnal.

Când doriți să analizați datele de contabilizare job, ele trebuie extrase din jurnalul QACGJRN cu ajutorul comenzii DSPJRN (Display Journal - Afișare jurnal). Cu această comandă puteți scrie intrările într-un fișier bază de date. Trebuie să scrieți programe de aplicație sau să scrieți o utilitate, cum ar fi utilitatea de interogare, pentru a analiza datele.

Cum funcționează contabilizarea de job

Pentru această privire generală a modului cum funcționează contabilizarea de joburi, presupuneți că trei joburi diferite intră în sistem.



RSL5879-2

Figura 4. Privire generală asupra contabilizării de job

1. Când Job1 este terminat, sistemul face un rezumat al resurselor utilizate și scrie intrarea de jurnal JB în jurnalul QACGJRN. Dacă codul de contabilizare a fost modificat în timpul jobului, va fi scrisă o intrare de jurnal JB pentru fiecare dată când codul de contabilizare a fost modificat și la sfârșitul jobului. Job1 nu produce nici o ieșire de imprimantă și nu este produs nici un istoric de job. De aceea, nu sunt produse nici o intrare de jurnal de tipărire directă (DP) sau tipărire spool (SP) pentru Job1.
2. Job2 tipărește un fișier direct la o imprimantă. Când fișierul este terminat se scrie o intrare de jurnal DP cu sumarul datelor tipărite. Când Job2 s-a terminat, sistemul face un rezumat al resurselor utilizate și scrie intrarea de jurnal JB. Job2 nu produce nici o ieșire de imprimantă spool și nu se produce nici un istoric de job. De aceea, nu este făcută nici o intrare de jurnal SP pentru Job2.

3. Job3 tipărește la un fișier care este spool. Intrarea de jurnal SP nu este scrisă decât dacă un scriitor de imprimantă nu tipărește fișierul. Când Job3 este terminat, sistemul face un rezumat al resurselor utilizate și scrie intrarea de jurnal JB. Dacă este făcut un istoric de job la terminarea jobului, el este considerat un fișier spool normal și este creată o intrare de jurnal SP dacă fișierul este tipărit.
4. Este pornit un scriitor de imprimantă și el tipărește fișierele făcute de unul sau mai multe joburi. Când scriitorul termină un fișier, el produce o intrare de jurnal SP. Intrarea de jurnal SP nu este făcută dacă fișierul este anulat înainte ca tipărirea să înceapă.
5. La închiderea perioadei de contabilizare, comanda DSPJRN (Display Journal - Afișare jurnal) poate fi utilizată pentru a scrie intrările de jurnal acumulate în fișierul bază de date.
6. Programele scrise de utilizator sau utilitatea de interogare pot fi utilizate pentru a analiza datele de contabilizare. Rapoarte cum ar fi cel de resurse utilizate vor compila date după un cod de contabilizare, utilizator sau tip de job anumite.

Caracteristicile de operare ale contabilizării de job:

Încercările sistemului iSeries de a aloca memorie principală cât mai eficient posibil. Este posibil ca un job să nu folosească aceeași cantitate de resurse la fiecare rulare.

De exemplu, dacă există mai multe joburi active pe sistemul dumneavoastră, un job petrece mai mult timp restabilind resursele necesare pentru rulare decât dacă este folosit un mediu de sistem dedicat. Sistemul folosește jobul și rulează prioritățile alocate către joburi diferite pentru a ajuta la gestionarea memoriei principale. De aceea, joburi cu prioritate ridicată pot folosi mai puține resurse de sistem decât joburile cu prioritate scăzută.

Din cauza acestor caracteristici ale sistemului de operare, ați putea dori să aplicați interpretarea sau algoritmul dumneavoastră propriu la datele de contabilizare job colectate. Dacă încasați bani pentru utilizarea sistemului dumneavoastră ați putea dori să încasați mai mult pentru joburi cu prioritate ridicată, lucru realizat în timpul orei de vârf a sistemului, sau folosirea resurselor critice.

Procesarea jurnalului de contabilizare:

Jurnalul de contabilizare QSYS/QACGJRN este procesat ca orice alt jurnal. Fișierele pot fi, de asemenea, înregistrate în acest jurnal deși, pentru simplificare, este recomandat să îl păstrați doar pentru informații de contabilizare.

Puteți utiliza comanda SNDJRNE (Send Journal Entry - Trimitere intrare de jurnal) pentru a îi trimite alte intrări acestui jurnal. În timp ce există considerente operaționale suplimentare implicate în utilizarea mai multor jurnale, există avantaje să NU se permită nici o intrare de fișier în jurnalul QACGJRN. Este de obicei mai ușor să se controleze jurnalul QACGJRN separat astfel încât toate intrările de contabilizare job pentru o anumită perioadă de contabilizare să fie într-un număr minim de receptori de jurnal și să fie pornit un nou receptor de jurnal, la începutul unei perioade de contabilizare. Intrările de sistem apar, de asemenea, în jurnalul QACGJRN. Acestea sunt intrările cu un cod de jurnal J, care se leagă de operațiile de IPL și cele generale realizate în receptorii de jurnal (de exemplu, o salvare a receptorului).

Intrări de contabilizare job

Intrările de contabilizare job se află în receptorul care începe o dată cu următorul job care intră în sistem după ce intră în vigoare comanda CHGSYSVAL (Change System Value - Modificare valoare sistem). Nivelul de contabilizare a unui job este determinat când acesta intră în sistem. Dacă valoarea de sistem QACGLVL, informații contabilizare jurnal, este modificată după ce jobul a pornit, ea nu are nici un efect asupra tipului de contabilizare care este realizată pentru acel job. Intrările DP (tipărire directă) și SP (tipărire spool) apar dacă jobul care a creat fișierul operează sub contabilizare și valoarea de sistem este setată pe *PRINT. Dacă fișierele spool sunt tipărite după ce a fost setat nivelul de contabilizare pe *PRINT sau dacă jobul care a creat fișierul a fost pornit înainte ca nivelul de contabilizare să fie modificat, nu se realizează nici o jurnalizare pentru aceste fișiere spool.

Când să utilizați contabilizarea joburilor

Ar trebui să utilizați funcția de contabilizare joburi din mesajele QHST din moment ce mesajele cu numerele CPF1124 și CPF1164 sunt întotdeauna disponibile în istoricul QHST? Sau ar trebui să utilizați contabilizarea de job? Utilizați următoarele informații pentru a vă ajuta să determinați ce metodă este cea mai bună pentru organizația dumneavoastră.

Informații suplimentare furnizate de contabilizarea de job

Contabilizarea de job are toate informațiile livrate de CPF1164 plus:

- Cod de contabilizare
- Numărul de fișiere, linii și pagini de imprimantă create de către programe
- Numărul de operații de citire, scriere și actualizare bază de date
- Numărul de operații de scriere și citire comunicații
- Linii și pagini reale tipărite
- Timpul cât jobul a fost activ și suspendat
- Numărul real de octeți de informații de control și date de tipărire trimise la imprimantă

Funcția de contabilizare job este mai eficientă pentru adunarea statisticilor de contabilizare job dacă:

- Informațiile resurselor referitoare la utilizarea bazei de date, a imprimantei și a comunicațiilor sunt importante.
- Codurile de contabilizare sunt alocate utilizatorilor sau joburilor.
- Informațiile pentru ieșirea tipărită sunt importante.
- Contabilizarea joburilor trebuie făcută pe baza unui segment de contabilizare dintr-un job decât pe baza unui job complet.
- Sunt necesare informațiile despre timpul activ și suspendat.

Mesajele QHST sunt mai eficiente pentru a aduna statistici de contabilizare job dacă:

- Nu doriți să gestionați obiectele suplimentare incluse în jurnalizare.
- Nu aveți nevoie de nici o altă informație despre resurse decât cele furnizate în mesajele CPF1124 și CPF1164, care sunt trimise automat către istoricul QHST.
- Nu aveți nevoie de informații de contabilizare a imprimantei.

Notă: Unele statistici înregistrate în mesajul CPF1164 și intrările de jurnal JB nu se vor potrivi perfect. Aceasta este, în principal, din cauza a doi factori: (1) Statisticile CPF1164 sunt înregistrate puțin înaintea statisticilor de jurnal JB și (2) de fiecare dată când este modificat un cod de contabilizare, are loc rotunjirea pentru unele câmpuri, în timp ce pentru mesajele CPF1164 rotunjirea are loc o singură dată.

Securitatea și contabilizarea joburilor

Doar responsabilul cu securitatea (sau un program care îi adoptă autorizarea) sau un utilizator cu autorizările *ALLOBJ și *SECADM pot modifica valoarea de sistem QACGLVL (informații contabilizare jurnal).

Modificarea intră în vigoare atunci când un job nou intră în sistem. Această restricție asigură că, dacă contabilizarea de job este în vigoare și responsabilul cu securitatea realizează un IPL de sistem, este scrisă o intrare de contabilizare pentru jobul responsabilului cu securitatea.

Autorizare la alocarea codurilor de contabilizare job

Puteți alocă coduri de contabilizare doar dacă aveți autorizarea să utilizați comenzile CRTUSRPRF (Create User Profile - Creare profil utilizator), CHGUSRPRF (Change User Profile - Modificare profil utilizator) sau CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare). Aceasta restricționează utilizarea codurilor de contabilizare și furnizează o bază pentru verificarea validității oricărei modificări.

Doar un utilizator cu autorizarea specială *SECADM are permisiunea de a utiliza comenzile CRTUSRPRF și CHGUSRPRF. Totuși, responsabilul cu securitatea poate delega această autorizare creând un program CL, care permite altui utilizator să adopte profilul responsabilului cu securitatea și să modifice parametrul **ACGCDE** din profilul utilizator. Individul ar putea atunci avea autorizare la unul sau mai multe programe CL.

Parametrul **ACGCDE** mai există, de asemenea, și în obiectele descrierii de job dar trebuie să aveți autorizare să utilizați comanda CHGACGCDE pentru a introduce o altă valoare decât cea implicită *USRPRF. CHGACGCDE este livrată cu autorizarea PUBLIC de *USE.

Autorizare la comanda CHGACGCDE

Dacă permiteți unui utilizator să utilizeze comanda CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare), acel utilizator poate:

- Să creeze sau să modifice parametrul **ACGCDE** din descrierile de job. (Este, de asemenea, necesară autorizarea pentru crearea sau modificarea descrierilor de job.)
- Să modifice codul de contabilizare din jobul său curent.
- Să modifice codul de contabilizare din alt job decât cel propriu dacă mai are și autorizarea specială *JOBCTL.

Puteți furniza securitate suplimentară utilizând comanda CHGACGCDE într-un program CL, care adoptă autorizarea proprietarului programului. Aceasta îi permite utilizatorului care rulează o funcție externă să realizeze o funcție sensibilă la securitate fără să aibă autorizare directă la comanda CHGACGCDE.

Jurnalul de contabilizare și receptorii săi sunt tratați ca orice alte obiecte de jurnal din punctul de vedere al securității. Trebuie să decideți ce autorizare ar trebui să existe pentru jurnalul de contabilizare și receptorul de jurnal.

Intrări de jurnal pentru contabilizare de job

Sistemul furnizează intrări de jurnal diferite pentru diferitele tipuri de date care pot fi adunate:

- Contabilizare resurse job: Intrarea de jurnal de job (JB) conține date care fac un rezumat al resurselor utilizate pentru un job sau pentru diferitele coduri de contabilizare utilizate într-un job.
- Contabilizare fișier imprimantă:
 - Intrare de jurnal Tipărire directă (DP): Conține date despre fișierele de imprimantă produse pe dispozitivele de tipărire (fără a fi puse în spool).
 - Intrare de jurnal Tipărire spool (SP): Conține date despre fișierele de imprimantă făcute de un scriitor de tipărire (spool).

Informații câmp de intrare jurnal de contabilizare job:

Acest subiect conține o listă de câmpuri care sunt în intrarea jurnalului JB. Informații suplimentare despre diferitele câmpuri se găsesc în fișierele de referință de câmp QSYS/QAJBACG4 și QSYS/QAJBACG44.

Tabela 3.

Nume câmp (Caracter 14)	Descriere	Atribute câmp	Comentarii
JAJOB	Nume job	Caracter (10)	
JAUSER	Utilizator job	Caracter (10)	
JANBR	Număr job	Aflat în zona (6,0)	
JACDE	Cod de contabilizare	Caracter (15)	
JACPU	Timp de utilizare unitate de procesare (în milisecunde)	Zecimal împachetat(11,0)	Timpul unității de procesare nu include statisticile de utilizare a unității de procesare și ale imprimantei pentru crearea istoricelor de job.
JARTGS	Număr de pași de rutare	Zecimal împachetat(5,0)	

Tabela 3. (continuare)

Nume câmp (Caracter 14)	Descriere	Atribute câmp	Comentarii
JAEDTE	Jobul a intrat în sistem - Data de intrare a jobului (în format llzxaa)	Caracter (6)	
JAETIM	Jobul a intrat în sistem - Ora de intrare job (în format oomsss)	Caracter (6)	
JASDTE	Ora și data de pornire job - Data de pornire job (format llzxaa)	Caracter (6)	Pentru data și ora de terminare a jobului din intrările de jurnal, utilizați câmpurile JODATE și JOTIME care sunt parte din informația de prefix a intrării de jurnal standard. (Vedeți cartea Copiere de rezervă și Recuperare pentru informații suplimentare despre aceste câmpuri.) După o oprire anormală a sistemului, aceste câmpuri conțin data și ora curentă și nu (ca la mesajele CPF1164) ora reală a opririi sistemului.
JASTIM	Ora și data de pornire job - Ora de pornire job (în format oomsss)	Caracter (6)	Pentru ora și data de terminare job din intrările de jurnal, utilizați câmpurile JODATE și JOTIME care sunt parte a informațiilor de prefix de intrare jurnal standard. (Vedeți cartea Copiere de rezervă și Recuperare pentru informații suplimentare despre aceste câmpuri.) După o oprire anormală a sistemului, aceste câmpuri conțin data și ora curentă și nu (ca la mesajele CPF1164) ora reală a opririi sistemului.
JATRNT	Timp total de tranzacție (în secunde)	Zecimal împachetat(11,0)	Timpul total de tranzacție este setat la -1 când: <ul style="list-style-type: none"> • Timpul este setat înapoi. • A survenit o depășire într-un fișier la o numărare. • Sistemul a căzut în timp ce jobul era activ.
JATRNS	Număr de tranzacții	Zecimal împachetat(11,0)	Ultima tranzacție (SIGNOFF) nu este numărată.
JAAUX	Operații auxiliare sincrone de I/E și operații de bază de date (includ pagini lipsă din orice motiv)	Zecimal împachetat (11,0)	
JATYPE	Tip job	Caracter (1)	Tipurile de job înregistrate sunt următoarele: <p>A Job autostart B Job batch (include comunicații și MRT) I Job interactiv M Monitor subsistem R Cititor de punere în spool W Scriitor de punere în spool Notă: Acestea sunt aceleași cu cele utilizate în mesajul CPF1164, cu excepția că mesajul CPF1164 include unele informații de job de sistem care nu sunt incluse în intrările de jurnal.</p>

Tabela 3. (continuare)

Nume câmp (Caracter 14)	Descriere	Atribute câmp	Comentarii
JACCDE	Cod de terminare	Zecimal împachetat (3,0)	Codurile de terminare, care sunt similare cu cele utilizate pentru mesajul CPF1164, sunt: 000 Terminare normală 010 Terminare normală în timpul opririi controlate sau oprii controlate a subsistemului 020 Jobul a depășit gravitatea de oprire 030 Jobul s-a oprit anormal 040 Jobul s-a oprit înainte de a deveni activ 050 Jobul s-a oprit în timp ce era activ 060 Subsistemul s-a oprit anormal în timp ce jobul era activ 070 Sistemul s-a oprit anormal în timp ce jobul era activ 080 Jobul s-a terminat în limita de timp 090 Jobul a forțat terminarea după ce limita de timp se terminase 099 Intrare de contabilizare cauzată de comanda CHGACGCDE
JALINE	Număr de linii de tipărire	Zecimal împachetat (11,0)	Numărul de linii de tipărire nu reflectă ceea ce este de fapt tipărit. Fișierele spool pot fi anulate sau tipărite cu multiple copii. Informația din intrarea jurnalului JB reflectă doar ceea ce a fost scris de către program. Aceasta exclude orice linie scrisă pentru istoricul de job. Vedeți discuția despre datele de contabilizare a fișierului de imprimantă DP și SP mai târziu în acest capitol.
JAPAGE	Număr de pagini tipărite	Zecimal împachetat (11,0)	
JAPRTF	Număr de fișiere de imprimantă	Zecimal împachetat(11,0)	
JADBPT	Număr de operații de scriere în bază de date	Zecimal împachetat (11,0)	Numerele înregistrate pentru operațiile de I/E bază de date nu includ operațiile de I/E ale cititorilor și scriitorilor, sau operațiile de I/E cauzate de comenzile CL CPYSPLF, DSPSPLF, sau WRKSPLF. Dacă SEQONLY(*YES) este în vigoare, aceste numere arată fiecare bloc de înregistrări citit și nu numărul de înregistrări individuale citite.
JADBGT	Număr de operații de citire bază de date	Zecimal împachetat (11,0)	Numerele înregistrate pentru operațiile de I/E bază de date nu includ operațiile de I/E de la cititori și scriitori sau operațiile de I/E cauzate de comenzile CL CPYSPLF, DSPSPLF sau WRKSPLF. Dacă SEQONLY(*YES) este în vigoare, aceste numere arată fiecare bloc de înregistrări citit și nu numărul de înregistrări individuale citite.
JADBUP	Număr de operații de actualizare bază de date, ștergere FEOD, eliberare, comitere și derulare înapoi	Zecimal împachetat(11,0)	Numerele înregistrate pentru operațiile de I/E bază de date nu includ operațiile de I/E de la cititori și scriitori sau operațiile de I/E cauzate de comenzile CL CPYSPLF, DSPSPLF sau WRKSPLF. Dacă SEQONLY(*YES) este în vigoare, aceste numere arată fiecare bloc de înregistrări citit și nu numărul de înregistrări individuale citite.
JACMPT	Numărul de operații de scriere de comunicații	Zecimal împachetat(11,0)	Numerele înregistrate pentru operații de I/E de comunicații nu includ activitatea stației de lucru aflată la distanță. Când I/E este pentru un dispozitiv de comunicații, numerele includ doar activitatea legată de fișierele ICF.

Tabela 3. (continuare)

Nume câmp (Caracter 14)	Descriere	Atribute câmp	Comentarii
JACMGT	Numărul de operații de citire de comunicații	Zecimal împachetat (11,0)	Numerele înregistrate pentru operații de I/E de comunicații nu includ activitatea stației de lucru aflată la distanță. Când I/E este pentru un dispozitiv de comunicații, numerele includ doar activitatea legată de fișierele ICF.
JAACT	Timpul cât jobul a fost activ (în milisecunde)	Zecimal împachetat (11,0)	
JASPN	Timpul cât jobul a fost suspendat (în milisecunde)	Zecimal împachetat (11,0)	
JAEDTL	Amprenta de timp când jobul a intrat în sistem (llzaaaaoommss)	Caracter (14)	
JAESTL	Amprentă de timp pornire job (llzaaaaoommss)	Caracter (14)	
JAAIO	I/E asincronă pentru operații de bază de date și non-bază de date	Zecimal împachetat(11,0)	
JAXCPU	Timp utilizare CPU expandat	Zecimal împachetat (29,0)	
JAXSIO	Operații auxiliare sincrone I/E expandate	Zecimal împachetat (29,0)	
JAXAIO	Operații auxiliare asincrone I/E expandate	Zecimal împachetat (29,0)	
JAXDBP	Număr expandat de puneri în baza de date	Zecimal împachetat (29,0)	
JAXDBG	Număr expandat de luări din baza de date	Zecimal împachetat(29,0)	
JAXDBU	Număr expandat de actualizări și ștergeri de bază de date	Zecimal împachetat(29,0)	
1 JAXLIN	Număr expandat de linii tipărite	Zecimal împachetat (29,0)	
1 JAXPAG	Număr expandate de pagini tipărite	Zecimal împachetat (29,0)	
1 JAXPRT	Număr de fișiere de imprimantă	Zecimal împachetat (29,0)	

Date de contabilizare fișier de imprimantă Tipărire directă (DP) și tipărire spool (SP):

Codul de contabilizare utilizat pentru intrările de jurnal DP sau SP este codul de contabilizare a jobului la momentul când fișierul este închis. Uneori este creată o intrare DP sau SP înainte ca fișierul să fie închis (ca atunci când este oprit un scriitor care creează un fișier SCHEDULE(*IMMED)). Când se întâmplă acest lucru este utilizat codul curent de contabilizare a jobului.

Este creată o intrare de jurnal DP sau SP pentru fiecare fișier tipărit. Dacă istoricul de job este pus în spool și apoi tipărit, este creată o intrare SP pentru el. De asemenea, o intrare SP este scrisă pentru fișierele spool de pe dischetă redirecționate către o imprimantă de către scriitorul de tipărire.

Informații jurnal de contabilizare DP:

Fișierul QSYS/QAPTACG5 conține câmpuri care sunt utilizate în intrarea jurnalului DP. Tabela din acest subiect listează aceste câmpuri și atributele lor.

Tabela 4.

Nume câmp	Descriere	Atribute câmp
JAJOB	Nume job	Caracter (10)
JAUSER	Utilizator job	Caracter (10)
JANBR	Număr job	Aflat în zona (6,0)
JACDE	Cod de contabilizare	Caracter (15)
JADFN	Nume fișier dispozitiv	Caracter (10)
JADFNL	Biblioteca în care fișierul de dispozitiv este memorat	Caracter (10)
JADFN	Nume dispozitiv	Caracter (10)
JADVT	Tip dispozitiv	Caracter (4)
JADVM	Model dispozitiv	Caracter (4)
JATPAG	Număr total de pagini de tipărire produse	Zecimal împachetat (11,0)
JATLIN	Număr total de linii de tipărire produse	Zecimal împachetat (11,0)
JASPFN	Întotdeauna blank	Caracter (10)
JASPNB	Întotdeauna blank	Caracter (4)
JAOPY	Întotdeauna blank	Caracter (1)
JAFMTP	Întotdeauna blank	Caracter (10)
JABYTE	Întotdeauna zero	Zecimal împachetat(15,0)
JAUSR	Date utilizator	Caracter (10)
JALSPN	Întotdeauna blank	Caracter (6)
JASPSY	Întotdeauna blank	Caracter (8)
JASPDT	Întotdeauna blank	Caracter (7)
JASPTM	Întotdeauna blank	Caracter (6)
JADFASP	Întotdeauna blank	Caracter (10)

Informații ale jurnalului de contabilizare SP:

Acest subiect conține o tabelă care menționează câmpurile (aflate în fișierul QSYS/QAPTACG5) din intrarea de jurnal SP.

Notă: Informațiile jurnalului de contabilizare SP sunt similare cu cele furnizate în datele jurnalului de contabilizare DP cu excepția că sunt incluse numele fișierului spool, numărul fișierului spool, prioritatea de ieșire, tipul formularului și numărul total de octeți de informații de control și date de imprimantă trimise către imprimantă. O intrare de jurnal SP nu este scrisă dacă este șters un fișier spool înainte ca un scriitor să înceapă să scrie fișierul la dispozitiv.

Tabela 5.

Nume câmp	Descriere	Atribute câmp
JAJOB	Nume job	Caracter (10)
JAUSER	Utilizator job	Caracter (10)
JANBR	Număr job	Aflat în zona (6,0)

Tabela 5. (continuare)

Nume câmp	Descriere	Atribute câmp
JACDE	Cod de contabilizare	Caracter (15)
JADFN	Nume fișier dispozitiv	Caracter (10)
JADFNL	Biblioteca în care sunt memorate fișierele de dispozitiv	Caracter (10)
JADDEVN	Nume dispozitiv	Caracter (10)
JADEVT	Tip dispozitiv	Caracter (4)
JADEVM	Model dispozitiv	Caracter (4)
JATPAG	Numărul total de pagini de imprimantă produse	Zecimal împachetat (11,0)
JATLIN	Numărul total de linii de imprimantă produse	Zecimal împachetat (11,0)
JASPFN	Nume fișier spool	Caracter (10)
JASPNB	Număr fișier spool	Caracter (4)
JAOPTY	Prioritate ieșire	Caracter (1)
JAFMTP	Tip formular	Caracter (10)
JABYTE	Numărul total de octeți trimiși la imprimantă	Zecimal împachetat (15,0)
JAUSRD	Date utilizator	Caracter (10)
JALSPN	Număr fișier spool	Caracter (6)
JASPSY	Nume de sistem al jobului de fișier spool	Caracter (8)
JASPDT	Data creație fișier spool (în format caallzz)	Caracter (7)
JASPTM	Oră creație fișier spool (în format oommss)	Caracter (6)
JADFASP	Nume ASP pentru biblioteca de fișiere de dispozitiv	Caracter (10)

Notă:

- Sistemul încearcă să înregistreze numărul real de pagini, linii și octeți tipăriți, dar când un scriitor este anulat *IMMED sau se recuperează dintr-o eroare de dispozitiv (precum terminarea formularelor), nu este posibil să determinați numărul exact de pagini, linii și octeți tipăriți.
- Paginile și liniile suplimentare produse cu rândul de aliniere nu sunt incluse în numerele de pagini, linii și octeți.
- Dacă un fișier spool intră în starea WTR (dar este setat pe MSGW) sau dacă un fișier este șters în timp ce se afla în starea MSGW, va apărea o intrare de jurnal SP în jurnalul de contabilizare DP care va indica faptul că sunt 0 pagini și 0 linii tipărite.
- În timpul utilizării unei imprimante configurate AFP(*YES), dacă ștergeți sau rețineți un fișier imediat după ce acesta a tipărit pagini, intrarea SP pentru acel fișier este posibil să indice 0 pagini și 0 linii tipărite, deși au fost tipărite unele pagini.
- Numerele de pagini, linii și octeți pentru job și separatorii de fișier sunt incluse cu numerele pentru fișierul cu care sunt asociate.
- Când un fișier IPDS conține grafice și coduri de bare și este trimis la o imprimantă IPDS care nu suportă grafice sau coduri de bare, numerele de pagini, linii și octeți includ graficele și codurile de bare care nu au fost tipărite.
- Dacă configurația imprimantei este AFP(*YES), câmpul pentru numărul total de linii de imprimantă produse este zero. Câmpul Număr total de pagini produse este corect.

Despre codul de contabilizare

Codul inițial de contabilizare (de până la 15 caractere lungime) pentru un job este determinat de valoarea parametrului **ACGCDE** (cod de contabilizare) din descrierea jobului și profilul utilizator pentru job.

Când este pornit un job, acestuia îi este alocată o descriere de job. Obiectul de descriere job conține o valoare pentru parametrul **ACGCDE**. Dacă este utilizată valoarea implicită *USRPRF, este folosit codul de contabilizare din profilul utilizator al jobului.

Notă: Când este pornit un job utilizând comanda SBMJOB (Submit Job - Lansare job), codul său de contabilizare este același cu cel al jobului lansatorului.

Puteți modifica codul de contabilizare după ce jobul a intrat în sistem utilizând comanda CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare).

Comenzile CRTUSRPRF și CHGUSRPRF suportă parametrul **ACGCDE**. Valoarea implicită este *BLANK. Dacă toată munca pentru un anumit utilizator va fi înregistrată sub un cod de contabilizare, doar profilul utilizator trebuie schimbat. Puteți modifica codurile de contabilizare pentru anumite descrieri de job specificând codul de contabilizare pentru parametrul **ACGCDE** din comenzile CRTJOBDD și CHGJOBDD. Comanda CHGACGCDE permite, de asemenea, diferite coduri de contabilizare într-un singur job.

Comanda RTVJOBA (Retrieve Job Attributes - Extragere attribute job) și API-urile care extrag attribute de job vă permit să accesați codul curent de contabilizare dintr-un program CL.

Contabilizare resurse

Datele de contabilizare a resurselor jobului sunt rezumate intrarea de jurnal a jobului (JB) la finalizarea unui job. În plus, sistemul creează o intrare de jurnal a jobului JB rezumând resursele folosite de fiecare dată când apare comanda Modificare a codului de contabilizare (CHGACGCDE). Intrarea de jurnal JB include:

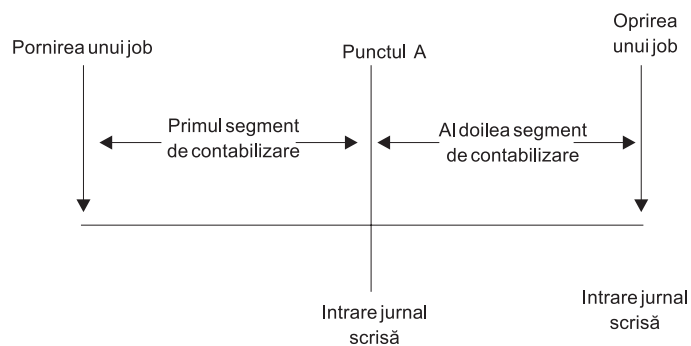
- Nume de job complet calificat
- Cod de contabilizare pentru segmentul de contabilizare ce doar s-a terminat
- Timpul unității de procesare
- Număr de pași de rutare
- Data și ora la jobul a intrat în sistem
- Data și ora la care jobul a pornit
- Timpul total de tranzacție (include timp de serviciu, timp ineligibil, și timp activ)
- Număr de tranzacții pentru toate joburile interactive
- Operații auxiliare I/O
- Tip job
- Cod de finalizare job
- Număr de linii de imprimantă, pagini, și fișiere create dacă sunt puse în spool sau tipărite direct
- Număr de citiri, scrieri, actualizări, și ștergeri ale fișierului de bază de date
- Număr de operațiuni de citire și scriere ale fișierului ICF

Notă: O parte din informația de contabilizare a jobului poate fi accesată de asemenea folosind mesajele CPF1124 și CPF1164 localizate în înregistrarea QHST.

Date de contabilizare a resurselor

La analizarea intrărilor de jurnal, este important de înțeles cum și când intrările de jurnal sunt scrise. O intrare de jurnal JB este scrisă la jurnalul de contabilizare a jobului pentru un job de câte ori codul de contabilizare a jobului este modificat și când jobul ia sfârșit. De aceea, un singur job poate avea intrări de jurnal multiple.

Fiecare contabilizare de resurse a intrării de jurnal conține informații despre resurse folosite în timp ce codul de contabilizare anterior era efectiv. Considerați următoarele exemple:



RZAKS550-0

Figura 5. Exemplu de contabilizare a resurselor de date

La un punct A, comanda CHGACGCDE a fost lansată. Codul de contabilizare este modificat și intrarea de jurnal JB este trimisă la jurnal. Intrarea de jurnal JB conține date pentru primul segment de contabilizare. Când jobul se finalizează, o a doua intrare JB este făcută pentru jobul care conține date pentru al doilea segment de contabilizare.

Dacă codul de contabilizare al jobului nu a fost modificat în timpul existenței jobului, intrarea JB singulară însumează resursele totale folosite de către job. Dacă codul de contabilizare al jobului a fost modificat în timpul existenței jobului, atunci trebuie să adăugați sus câmpurile în intrările multiple JB pentru a putea determina resursele totale folosite de către job. Crearea unei înregistrări de job nu se pune la socoteală la folosirea unității de procesare pentru un job sau pentru ieșirea sa tipărită în intrările de contabilizare JB. Totuși, dacă folosiți contabilizare de fișier de imprimantă, înregistrarea de job tipărită este inclusă în intrările de jurnal ale fișierului de imprimantă.

Job prestart și contabilizare job

Dacă sistemul dumneavoastră folosește contabilizare job, programul jobului prestart ar trebui să ruleze comanda Modificare job prestart (CHGPJ) cu valoarea cererii de pornire a programului pentru parametrul codului de contabilizare (CHGPJ ACGCDE(*PGMSTRRQS)) imediat după ce cererea de pornire a programului se atașează la jobul prestart.

Această acțiune modifică codul de contabilizare la valoarea specificată profilul utilizatorului asociat cu cererea de pornire a programului. Imediat înainte ca programul să termine manipularea cererii de pornire a programului, programul ar trebui să ruleze comanda Modificare job prestart (CHGPJ) cu valoarea Intrare job prestart pentru parametrul codului de contabilizare (CHGPJ ACGCDE(*PJE)). Aceasta modifică codul de contabilizare înapoi la valoarea specificată în descrierea de job a intrării jobului prestart.

Procesare job sistem pentru contabilizare job

Joburile de sistem pe care le controlați (de exemplu, cititori și scriitori) sunt alocate un cod de contabilizare *SYS. Alte joburi de sistem pe care nu le controlați (de exemplu, QSYSARB, QLUS,SCPF) nu primesc o intrare de jurnal.

Notă: Nu puteți utiliza comanda CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare) pentru a modifica codul de contabilizare a monitorului subsistemului sau un cititor sau un scriitor. Puteți, totuși, modifica codul de contabilizare al unui cititor sau scriitor, modificând descrierile de job corespunzătoare livrate de IBM și apoi pornindu-le din nou.

Procesarea batch și contabilizarea jobului

Orice job batch care este lansat utilizând comanda SBMJOB (Submit Job - Lansare job) utilizează automat același cod de contabilizare ca și jobul care a lansat jobul batch. Când este utilizată comanda SBMJOB, codurile de contabilizare nu pot fi suprascrise indiferent de modul cum este codată intrarea de descriere job.

Dacă doriți ca jobul batch să opereze sub alt cod de contabilizare decât cel al jobului lansator, ar trebui emisă o comandă CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare) fie:

- Înainte și după ce este lansată comanda SBMJOB
- Imediat de către jobul batch.

Joburile batch lansate utilizând un cititor sau o comandă SBMDBJOB (Submit Database Job - Lansare job de bază de date) utilizează codul de contabilizare specifică în descrierea de job pentru jobul batch. Dacă descrierea jobului specifică ACGCDE(*USRPRF), codul de contabilizare este luat din profilul utilizator folosit pentru job.

Procesare interactivă și contabilizare job

Dacă un job interactiv are un set fixat de opțiuni pentru un utilizator și fiecare opțiune are alocat un cod de contabilizare, ar putea fi dezirabil să se aloce automat un nou cod când utilizatorul cere să lucreze pe o funcție nouă.

O abordare tipică este aceea ca o opțiune de meniu să ceară o nouă zonă funcțională. Comanda CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare) este atunci emisă într-un program CL și valorile jobului utilizate pentru codul de contabilizare anterior sunt rezumate în intrarea jurnalului de contabilizare JB.

Dacă un utilizator are mai multe alocări pentru care doar utilizatorul cunoaște codurile de contabilizare care trebuie utilizate, puteți:

- Să dați autorizare utilizatorului să introducă comanda CHGACGCDE.
- Să scrieți un program care să prompteze utilizatorul pentru codul de contabilizare.

Notă: Pentru joburi sursă pass-through, informațiile de contabilizare job nu includ jobul destinație pass-through. Pentru joburile destinație pass-through, informațiile de contabilizare job nu includ jobul batch de comunicații asociat.

Contabilizare fișier imprimantă

Există două tipuri de intrări de jurnal pentru contabilizare fișier imprimantă; DP pentru fișiere de imprimantă ne puse în spool și SP pentru fișiere de imprimantă puse în spool. Aceste două tipuri de intrări de jurnal împart un format de intrare de jurnal deși o parte din informație este disponibilă numai în intrarea SP. Intrările de jurnal DP și SP includ informații cum ar fi:

- Nume de job complet calificat
- Cod de contabilizare
- Nume fișier și bibliotecă dispozitiv
- Nume dispozitiv
- Tip și model dispozitiv
- Numărul total de pagini și linii tipărite. Dacă apar copii multiple, aceasta este suma tuturor copiilor
- Nume fișier de spool (numai în intrarea SP)
- Număr fișier de spool (numai în intrarea SP)
- Prioritate de ieșire (numai în intrarea SP)
- Tip formular (numai în intrarea SP)
- Tip formular (numai în intrarea SP)
- Numărul total de octeți de informație de control și date de tipărire trimise dispozitivului de tipărire. Dacă apar copii multiple, aceasta este suma tuturor copiilor. (Aceasta se aplică numai la intrarea SP.)

Intrările de jurnal DP și SP apar atunci când fișierul este tipărit. Dacă un fișier de spool nu este tipărit niciodată, nici-o intrare de jurnal SP nu va apărea.

Control funcționare

Ca operator sau administrator de sistem, una dintre sarcinile dumneavoastră este să întrețineți buna funcționare a serverului. Aceasta presupune să monitorizați, gestionați, și să vă asigurați ca joburile, cozile de joburi, pool-urile de memorie, istoricul joburilor, și cozile de ieșire funcționează corect.

Subiectele din această secțiune vă oferă informații despre diferitele tipuri de operații de Control funcționare zilnice precum și alte operații pe care este posibil să trebuiască să le realizați pe serverul dumneavoastră iSeries. Fiecare subsubiect explică de ce este important ca să realizați aceste operații, precum și cum să le efectuați.

Modificarea programului de pornire IPL

Creați un program de pornire care va modifica resursele sistemului și resursele și atributele alocate lor care sunt pornite în timpul unui IPL. De obicei, subsistemele, scriitorii și Asistentul operațional sunt lansate în execuție de acest program.

Jobul autostart din subsistemul de control transferă controlul programului care este specificat în valoarea de sistem Program de pornire pentru setare sistem (QSTRUPPGM). Puteți ajusta acest program.

Puteți crea propriul dumneavoastră program și modifica valoarea de sistem Program de pornire pentru setare sistem (QSTRUPPGM) cu acel nume de program. Sau, puteți utiliza programul livrat QSTRUP din QSYS ca o bază pentru a vă crea propriul program. Pentru a face aceasta:

1. Extrageți sursa programului livrat utilizând comanda RTVCLSRC (de exemplu, **RTVCLSRC PGM(QSYS/QSTRUP) SRCFILE(YOURLIB/YOURFILE)**).
2. Modificați programul.
3. Creați programul utilizând comanda CRTCLPGM, punându-l în propria dumneavoastră bibliotecă.
4. Testați programul pentru a vă asigura că funcționează.
5. Modificați valoarea de sistem QSTRUPPGM, program de pornire pentru setarea sistemului, cu numele programului și biblioteca pe care le-ați specificat în comanda CRTCLPGM.

Declinarea responsabilității pentru exemplele de cod

IBM vă acordă o licență de copyright neexclusivă pentru a folosi toate exemplele de cod de programare din care puteți genera funcții similare, adaptate nevoilor dumneavoastră specifice.

| EXCEPTÂND GARANȚIILE OBLIGATORII, CARE NU POT FI EXCLUSE, IBM, DEZVOLTATORII DE
| PROGRAME ȘI FURNIZORII SĂI NU ACORDĂ NICI O GARANȚIE SAU CONDIȚIE, EXPRESĂ SAU
| IMPLICITĂ, INCLUZÂND, DAR FĂRĂ A SE LIMITA LA ELE, GARANȚIILE SAU CONDIȚIILE IMPLICITE
| DE VANDABILITATE, DE POTRIVIRE PENTRU UN ANUMIT SCOP SAU DE NEÎNCĂLCARE A UNUI
| DREPT, REFERITOARE LA PROGRAM SAU LA SUPORTUL TEHNIC, DACĂ ESTE CAZUL.

| ÎN NICI O ÎMPREJURARE IBM, DEZVOLTATORII SĂI DE PROGRAME SAU FURNIZORII NU VOR FI
| RESPONSABILI PENTRU ORICARE DINTRE URMĂTOARELE PAGUBE, CHIAI DACĂ AU FOST
| INFORMAȚI ÎN LEGĂTURĂ CU POSIBILITATEA PRODUCERII LOR:

1. PIERDEREA SAU DETERIORAREA DATELOR;
2. PAGUBE DIRECTE, SPECIALE, ACCIDENTALE SAU INDIRECTE SAU PREJUDICIILE ECONOMICE DE
| CONSECINȚĂ; SAU
3. PIERDERI REFERITOARE LA PROFIT, AFACERI, BENEFICII, REPUTAȚIE SAU ECONOMII
| PLANIFICATE.

| UNELE JURISDICȚII NU PERMIT EXCLUDEREA SAU LIMITAREA PREJUDICIILOR DIRECTE,
| INCIDENTALE SAU DE CONSECINȚĂ, CAZ ÎN CARE ESTE POSIBIL CA UNELE SAU TOATE LIMITĂRILE
| SAU EXCLUDERILE DE MAI SUS SĂ NU FIE VALABILE PENTRU DUMNEAVOASTRĂ.

Sursă pentru program de pornire CL

Obiect	Comandă	Sursă program CL
QSTRUP	CRTCLPGM	<pre> PGM DCL VAR(&STRWTRS) TYPE(*CHAR) LEN(1) DCL VAR(&CTLSBSD) TYPE(*CHAR) LEN(20) DCL VAR(&CPYR) TYPE(*CHAR) LEN(90) VALUE('+ 5722-SS1 (C) COPYRIGHT IBM CORP 1980, 2000. + MATERIAL CU LICENȚĂ - PROGRAM PROPRIETATE A IBM') QSYS/STRSBS SBSD(QSERVER) MONMSG MSGID(CPF0000) QSYS/STRSBS SBSD(QUSRWRK) MONMSG MSGID(CPF0000) QSYS/RLSJOBQ JOBQ(QGPL/QS36MRT) MONMSG MSGID(CPF0000) QSYS/RLSJOBQ JOBQ(QGPL/QS36EVOKE) MONMSG MSGID(CPF0000) QSYS/STRCLNUP MONMSG MSGID(CPF0000) QSYS/RTVSYSVAL SYSVAL(QCTLSBSD) RTNVAR(&CTLSBSD) IF ((&CTLSBSD *NE 'QCTL QSYS ') + *AND (&CTLSBSD *NE 'QCTL QGPL ')) GOTO DONE QSYS/STRSBS SBSD(QINTER) MONMSG MSGID(CPF0000) QSYS/STRSBS SBSD(QBATCH) MONMSG MSGID(CPF0000) QSYS/STRSBS SBSD(QCMN) MONMSG MSGID(CPF0000) DONE: QSYS/STRSBS SBSD(QSPL) MONMSG MSGID(CPF0000) QSYS/RTVSYSVAL SYSVAL(QSTRPRTWTR) RTNVAR(&STRWTRS) IF (&STRWTRS = '0') GOTO NOWTRS CALL PGM(QSYS/QWCSWTRS) MONMSG MSGID(CPF0000) NOWTRS: RETURN CHGVAR VAR(&CPYR) VALUE(&CPYR) ENDPGM </pre>

Valoarea de sistem QSTRUPPGM - Program de pornire (start-up) setare sistem

Programul de pornire pentru setarea sistemului (QSTRUPPGM) este programul start-up. Această valoare specifică numele programului care este chemat de la un job autostart când este pornit subsistemul de control. Acest program realizează funcții de setare, cum este pornirea subsistemelor și a imprimantelor. Această valoare de sistem poate fi modificată doar de către un responsabil cu securitatea sau de cineva cu autorizare de responsabil cu securitatea. O modificare asupra acestei valori de sistem intră în vigoare următoarea dată când este realizat un IPL. QSTRUPPGM poate avea valorile:

- 'QSTRUP QSYS': Programul specificat este rulat ca rezultat al unui transfer de control către el de la jobul autostart din subsistemul de control.
- '*NONE': Jobul autostart se termină normal fără a apela un program.

Programul de pornire implicit QSYS/QSTRUP face următoarele:

- Pornește subsistemul QSPL pentru lucrul pus în spool.
- Pornește subsistemul QSERVER pentru lucrul cu serverul de fișiere
- Pornește subsistemul QUSRWRK pentru lucrul utilizatorului
- Eliberează cozile de joburi QS36MRT și QS36EVOKE dacă au fost reținute (acestea sunt utilizate de către mediul System/36).
- Pornește curățarea de Asistent operațional, dacă îi este permis.
- Pornește toate scriitorii de imprimantă doar dacă utilizatorul nu a specificat Nu în ecranul Opțiuni IPL.
- Dacă subsistemul de control este QCTL, el pornește subsistemele QINTER, QBATCH și QCMN.

Tip	Lungime	Valoare livrată CL
Caracter	20	'QSTRUP QSYS'

Vedeți valoarea de sistem Program de pornire pentru setare sistem (QSTRUPPGM) pentru informații mai detaliate.

Apelarea unui program de recuperare IPL special

Pentru a apela un program de recuperare special pentru situații când IPL-ul sesizează că oprirea anterioară a sistemului a fost anormală, puteți adăuga o intrare de job autostart la descrierea subsistemului pentru subsistemul de control.

Acest program verifică valoarea de sistem QABNORMSW (stare oprire anterioară sistem). Pentru o oprire normală a sistemului, valoarea lui QABNORMSW este '0' și pentru o oprire anormală a sistemului valoarea lui QABNORMSW este '1'. O alternativă este să se abandoneze mesajele și să se pornească alte subsisteme când funcția dumneavoastră de recuperare s-a terminat.

```

1.00 /* SPCRECOV - Program autostart pentru apelarea programului de recuperare special */
2.00      PGM
3.00      DCL      &QABNORMSW *CHAR LEN(1)
4.00      RTVSYSVAL SYSVAL(QABNORMSW) RTNVAR(&QABNORMSW)
5.00      IF      (&QABNORMSW *EQ '1') DO /* Recuperare */
6.00      SNDPGMMSG MSG('Program de recuperare în curs de operare-nu +
7.00      porniți subsisteme până când nu sunteți notificat') +
8.00      TOMSGQ(QSYSOPR)
9.00      CALL      RECOVERY
10.00     SNDPGMMSG MSG('Recuperare completă-pot fi pornite joburi') +
11.00     TOMSGQ(QSYSOPR)
12.00     ENDDO /* Recuperare */
13.00     ENDPGM

```

Monitorizare activitate sistem

Monitorizarea activității sistemului este una din cele mai importante operații pe care le face un administrator. Monitorizarea fluxului de lucru prin sistem este doar o parte din informația care trebuie zilnic monitorizată. Puteți realiza aceasta într-o mulțime de moduri, cum ar fi utilizând Navigatorul iSeries și Administrarea centrală a Navigatorului iSeries.

Modelată după jumătatea de sus a ecranului Gestionare stare sistem (WRKSYSSTS) din interfața bazată pe caractere, fereastra Stare sistem oferă o cale rapidă și ușoară de verificare a stării unui sistem. Administrarea centrală vă permite să monitorizați mai în adâncime funcțiile prin utilizarea monitoarelor de sistem.

Puteți accesa fereastra Stare sistem din folderul **Sistem** sau folderul **Control funcționare**.

Pentru a ajunge la Starea sistemului din folderul **Sistem** :

1. În Navigatorul iSeries, expandați **Conexiunile mele**.
2. Faceți clic dreapta pe conexiunea pe care doriți să lucrați și faceți clic pe **Stare sistem**.

Pentru a ajunge la Starea sistemului din folderul Control funcționare:

1. În Navigatorul iSeries, expandați **Control funcționare**.
2. Faceți clic dreapta pe **Control funcționare** și faceți clic pe **Stare sistem**.

Pentru informații suplimentare despre diferitele operații pe care le puteți efectua utilizând starea sistemului, vedeți ajutorul Navigatorului iSeries.

Verificarea utilizării pool-ului de memorie

Verificarea periodică a cantității de memorie folosită de pool-ului de memorie este importantă. Prin monitorizarea acestor niveluri, se poate regla rularea la eficiență maximă a grupurilor, care la rândul lor, păstrează rularea lentă a ciclului de lucru. În Navigatorul iSeries, puteți cu ușurință monitoriza cantitatea de memorie pe care o utilizează pool-urile dumneavoastră.

Ca să verificați folosirea memoriei, faceți următoarele:

1. În Navigatorul iSeries , expandați **Conexiunile mele** → **server** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active** sau **Pool-uri partajate**.
2. Faceți clic dreapta pe pool-ul de memorie cu care doriți să lucrați (de exemplu, Interactiv) și faceți clic pe **Proprietăți**.
3. Faceți clic pe fișa Configurație. Câmpul **Curent**, care se află în grupul Dimensiune, arată cantitatea de memorie pe care o are momentan pool-ul.

Notă: Puteți de asemenea să vedeți mărimea curentă a unui pool de memorie când apăsați **Pool-uri de memorie** sau **Pool-uri participante**. Dimensiune curentă (în MB) este o coloană implicită pe care o vedeți când este afișată o listă cu pool-uri de memorie în panoul din dreapta a Navigatorului iSeries.

Controlul nivelurilor de activitate a sistemului

Puteți controla câtă activitate există pe sistem controlând câte joburi pot fi active în același timp într-un subsistem sau controlând utilizarea unității de procesare de către joburile care au fost deja pornite.

Tabela 6. Modalități de a controla nivelurile de activitate ale sistemului

Ce pot să controlez?	Ce pot să utilizez pentru a controla?	Metoda interfeței bazate pe caractere	Metoda interfeței Navigatorului iSeries
Număr de joburi active	Descriere subsistem	<p>Comandă: CHGSBSD MAXJOBS</p> <p>Utilizați acest parametru pentru a specifica câte joburi pot fi active în același timp într-un subsistem.</p> <p>Pentru un subsistem activ, suma tuturor joburilor care sunt active în același timp care sunt pornite prin intrări de lucru din subsistem nu poate depăși valoarea parametrului MAXJOBS.</p> <p>Aceasta exclude joburile autostart, care pot cauza temporar depășirea limitei când este pornit subsistemul.</p>	<p>Utilizați fereastra Rulare comandă.</p> <p>Faceți clic dreapta pe sistemul punct final → Rulare comandă</p> <p>Tastați comanda CHGSBSD și apoi faceți clic pe Prompt.</p>
	Intrare în coada de joburi	<p>Comandă: CHGJOBQE MAXACT</p> <p>Utilizați acest parametru pentru a specifica câte joburi batch dintr-o coadă de joburi pot fi active simultan în subsistem.</p> <p>Un MAXACT de 1 pentru o coadă de joburi forțează joburile să fie selectate serial, după prioritatea jobului. Parametrul MAXPTYn este utilizat pentru a specifica câte joburi pot fi active pentru o prioritate de job specificată.</p>	<p>Utilizați fereastra Rulare comandă.</p> <p>Faceți clic dreapta pe sistemul punct final → Rulare comandă</p> <p>Tastați comanda CHGJOBQE și apoi faceți clic pe Prompt.</p>
	Intrare stație de lucru	<p>Comandă: CHGWSE MAXACT</p> <p>Utilizați acest parametru dacă este specificat parametrul WRKSTNTYPE. Acest parametru specifică câte joburi interactive pot fi active în același timp în subsistem pentru acea intrare.</p>	<p>Utilizați fereastra Rulare comandă.</p> <p>Faceți clic dreapta pe sistemul punct final → Rulare comandă</p> <p>Tastați comanda CHGWSE și apoi faceți clic pe Prompt.</p>
	Intrare de comunicații	<p>Comandă: CHGCMNE MAXACT</p> <p>Utilizați acest parametru pentru a specifica câte joburi batch de comunicații pot fi active simultan pentru acea intrare.</p>	<p>Utilizați fereastra Rulare comandă.</p> <p>Faceți clic dreapta pe sistemul punct final → Rulare comandă</p> <p>tastați comanda CHGCMNE și apoi faceți clic pe Prompt.</p>
	Intrare de rutare	<p>Comandă: CHGRTGE MAXACT</p> <p>Utilizați această comandă pentru a specifica câte joburi pot fi active simultan utilizând o intrare de rutare dată.</p>	<p>Utilizați comanda Rulare comandă.</p> <p>Faceți clic dreapta pe sistemul punct final → Rulare comandă</p> <p>tastați comanda CHGRTGE și apoi faceți clic pe Prompt.</p>
	Intrare job prestart	<p>Comandă: CHGPJE MAXJOBS</p> <p>Utilizați această comandă pentru a specifica câte joburi prestart pot fi active simultan pentru acea intrare.</p>	<p>Utilizați fereastra Rulare comandă.</p> <p>Faceți clic dreapta pe sistemul punct final → Rulare comandă</p> <p>Tastați comanda CHGPJE și apoi faceți clic pe Prompt.</p>

Tabela 6. Modalități de a controla nivelurile de activitate ale sistemului (continuare)

Ce pot să controlez?	Ce pot să utilizez pentru a controla?	Metoda interfeței bazate pe caractere	Metoda interfeței Navigatorului iSeries
Număr de joburi active (continuat)	Sistemul	Valoarea de sistem QMAXACTLVL (Maximum eligible threads - Maxim de fire de execuție eligibile) este utilizată pentru a specifica câte fire de execuție pot partaja spațiul de stocare principal și resursele procesorului simultan. Toate joburile active (inclusiv joburile de sistem) din toate pool-urile de spațiu de stocare sunt controlate de QMAXACTLVL.	Conexiunile mele → server → Configurare și service → Valori de sistem → Categorie de performanță → fișa Pool-uri de memorie → Maxim de fire de execuție eligibile
Utilizarea unității de procesare și a spațiului de stocare principal	Pool-uri de spațiu de stocare de bază	Valoarea de sistem QBASACTLVL pentru numărul maxim de fire de execuție eligibile pentru pool-ul Bază este utilizată pentru a specifica câte fire de execuție poate partaja în același timp Pool-ul de memorie Bază și pentru a limita conflictele din memoria principală.	Conexiunile mele → server → Configurare și service → Valori de sistem → Categorie de performanță → fișa Pool-uri de memorie → Pool de memorie de bază: Maxim de fire de execuție eligibile
	Pool-uri partajate	Comandă: WRKSHRPOOL Utilizați această comandă pentru a specifica nivelul de activitate pentru pool-urile partajate	Conexiunile mele → server → Control funcționare → Pool-uri de memorie → Pool-uri partajate → Faceți clic dreapta pe un pool partajat → Proprietăți → fișa Configurare și modificați câmpul Maximul de fire de execuție eligibile
	Pool-uri de spațiu de stocare privat	Comandă: CHGSBSD POOLS Utilizați această comandă pentru a specifica nivelul de activitate pentru pool-urile de memorie principală definite de utilizator.	Utilizați fereastra Rulare comandă. Faceți clic dreapta pe sistemul punct final → Rulare comandă Tastați comanda CHGSBSD și apoi faceți clic pe Prompt .

Exemple: controlul activității:

Aceste exemple arată relația unora din controalele de activitate. Presupuneți că nivelul de activitate al sistemului este 100 și joburile sunt cu un singur fir de execuție.

Exemplu de pool de memorie de bază

Două subsisteme, SBSA și SBSB, utilizează pool-ul de memorie de bază pentru a rula joburi. SBSA are momentan două joburi care rulează în acest pool de memorie și SBSB are unul. O intrare în coada de joburi din descrierea subsistemului SBSB specifică faptul că pot fi pornite oricâte joburi. Nivelul de activitate a Pool-ului de memorie de bază este 3. De aceea, doar trei joburi din Pool-ul de memorie de bază pot concura pentru unitatea de procesare la un moment dat. Totuși, toate joburile sunt pornite.

Exemplu de patru joburi într-un subsistem

Un job autostart, două joburi de stație de lucru, un job batch (patru joburi în total) sunt în subsistemul SBSC. MAXACT pentru SBSC este specificat a fi 4. Indiferent de ce este specificat pentru MAXACT al intrărilor de lucru, nici un alt job nu poate fi pornit până când cel puțin un job nu termină de rulat.

Exemplu de subsistem batch MAXACT(1)

Subsistemul SBSE este un subsistem batch pentru care MAXACT este specificat a fi 1. Deși intrarea în coada de joburi nu specifică MAXACT, limita este de un singur job deoarece este specificat 1 pentru valoarea MAXACT a subsistemului. De aceea, joburile sunt procesate, după prioritatea jobului, câte unul din coadă.

Determinarea stării jobului

Monitorizarea jobului vă va ajuta să înțelegeți ce face jobul dumneavoastră. Starea jobului este o importantă piesă a informației pe care o puteți folosi să depistați ce joburi sunt făcute.

Pentru a verifica starea unui job activ sau job sever, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active** sau **Joburi server**.

Notă: Puteți vedea starea unui job din orice loc din folderul Control funcționare de unde accesați joburile.

2. Priviți în coloana Stare detaliată pentru a determina starea unui job (de exemplu, În așteptarea unui eveniment, În așteptarea intervalului de timp sau În așteptarea scoaterii din coadă).

Indiciu: Dacă nu vedeți coloana Stare detaliată, o puteți adăuga în ecran făcând clic dreapta pe **Joburi active**(sau **Joburi server**) și selectând **Personalizarea acestei vizualizări** → **Coloane**.

Monitorizarea unui subsistem

Pentru că subsistemele sunt importante pentru activitatea zilnică făcută pe sistemul dumneavoastră, este important să monitorizați activitatea din subsisteme.

Într-o descriere de job puteți specifica numărul de joburi care pot rula la un moment dat în sistem setând valoarea maxim de joburi active. Pe măsură ce cantitatea de lucru în sistemul dumneavoastră crește ați putea dori să modificați valoarea maximului de joburi active din subsistemul dumneavoastră. Numărul pe care îl furnizați aici ar trebui setat astfel încât resursele disponibile să fie utilizate corect. Creșterea numărului de joburi active fără a crește resursele disponibile poate afecta negativ performanța sistemului dumneavoastră.

Pentru a verifica valoarea maximului de joburi active a subsistemului dumneavoastră, puteți utiliza fie Navigatorul iSeries sau interfața bazată pe caractere.

Navigatorul iSeries:

1. În In iSeries Navigator, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Subsisteme** → **Subsisteme active**.
2. Faceți clic dreapta pe subsistemul pe care doriți să îl monitorizați.
3. Selectați **Proprietăți**.

Notă: Asigurați-vă că setați această opțiune cu mare atenție. Dacă setați o valoare maximă de joburi active prea mare, este posibil să faceți sistemul să ruleze foarte greu. Totuși, dacă setați numărul maxim de joburi active prea mic, este posibil ca lucrul dumneavoastră să înceapă să lucreze cu slabe performanțe și să aibă loc o gâtuire.

Interfața bazată pe caractere:

Comandă: Afișare descriere subsistem (DSPSBSD)

Selectați opțiunea 1: Atribute operaționale, pentru a vedea valoarea pentru numărul maxim de joburi din subsistem.

Determinare a numărului de subsisteme folosind un pool de memorie

Subsistemelor le este alocat un anumit procentaj din memorie pentru rularea de joburi. Este important de știut cum multe subsisteme diferite sunt trase din același pool de memorie. După ce aflați câte subsisteme sunt joburi lansate la un pool și câte joburi rulează într-un pool, veți dori să reduceți conflictul pentru resurse prin ajustarea dimensiunii și nivelul de activitate a pool-ului.

Navigators iSeries:

Pentru a folosi Navigator iSeries pentru a monitoriza numărul de subsisteme care folosesc un pool de memorie, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active** sau **Pool-uri partajate**.
2. Faceți clic dreapta pe pool-ul de memorie cu care doriți să lucrați și faceți clic pe **Subsisteme**.
De la această fereastră puteți determina numărul de subsisteme care folosesc o memorie individuală pentru a-și rula joburile.

Interfața bazată pe caractere:

Comanda: Lucrul cu subsisteme (WRKSBS)

Această comandă afișează o listă a tuturor subsistemelor și pool-urile lor corespunzătoare.

Vizualizarea statisticilor de performanță ale jobului

O performanță de job este importantă pentru oricine folosește un server iSeries deoarece un singur job care rulează mai prost poate afecta alte joburi din sistem. Pentru a vizualiza joburile potențial problematice vi se oferă posibilitatea de a preveni problemele de performanță înainte ca ele să apară.

Fereastra Statistici performanțe trecute vă permite să monitorizați folosirea unui CPU de job, disc I/E (unitate de disc intrare/ieșire), rata paginilor lipsă, timp de răspuns medii, și numărul de tranzacții interactive. Puteți selecta o opțiune în această fereastră pentru a reîmprospăta aceste statistici manual sau într-o planificare.

Pentru a afișa statisticile de performanță trecute, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Joburi active**.
Notă: Puteți vizualiza performanța unui job din orice locație din gestiunea de lucru unde puteți vedea joburi.
Fereastra Statistici de performanță trecute poate fi afișată de la fișa Performanță a unei ferestre Proprietate job.
2. Faceți clic dreapta pe job pentru care doriți să afișați statisticile de performanță, și faceți clic pe **Detalii** → **Statistici de performanță trecute**.
Puteți reîmprospăta, reseta și planifica statisticile de performanță pentru a se reîmprospăta automat.
Notă: Puteți căuta în Statistici de performanță trecute pentru mai multe joburi în același timp prin deschiderea mai multor ferestre. Aceasta vă permite să vizualizați mai multe joburi problematice în același timp. Fiecare fereastră deține informații pentru un singur job.

Statisticile de performanță trecute sunt o variantă de a vizualiza performanța unui job în timpul traseului său în sistem. O altă cale pentru a vizualiza joburile pe sistem este prin folderul Administrare centrală. Puteți monitoriza joburi în Administrare centrală precum și performanța sistemului și mesajele.

Vizualizarea stării generale a sistemului

Navigators iSeries pune toate informațiile referitoare la starea sistemului într-un singur loc. Aceasta vă ușurează monitorizarea funcționării sistemului, identificarea zonelor cu potențiale defecte și determinarea rapidă a acțiunii de care aveți nevoie pentru a îmbunătăți performanțele.

Fereastra Stare sistem împarte starea generală a sistemului în șase zone specifice:

General

Aceasta este procentul de utilizare trecută a CPU, numărul de joburi active, procentul de utilizare a adreselor, procentul de utilizare a pool-ului de discuri, totalul de joburi din sistem, procentele de adrese permanente sau temporare utilizate, spațiul total de disc și capacitatea pool-ului de discuri a sistemului.

Joburi Numărul total de joburi, numărul de joburi active, numărul maxim de joburi și numărul de fire de execuție active.

Procesoare

Procentul de utilizare trecută a CPU. (În funcție de configurația dumneavoastră hardware, este posibil să mai vedeți și informații suplimentare referitoare la tipul de procesor(oare), numărul de procesoare, puterea de procesare, procesoare virtuale, performanță interactivă, utilizarea trecută a pool-ului de procesor partajat și utilizarea trecută a capacității izolate a CPU.)

Memorie

Memoria totală (memoria principală) din sistemul dumneavoastră și un buton care vă permite accesul la o listă de pool-uri de memorie active din sistem.

Spațiu pe disc

Spațiul total de pe disc, capacitatea și utilizarea pool-ului de discuri al sistemului, informațiile despre spațiul de stocare temporar utilizat și butoanele care vă permit accesul la informații mai multe despre starea discului, lista de pool-uri de disc din sistem și valorile de spațiu de stocare al sistemului.

Adrese Informațiile despre adresele permanente și temporare utilizate, adresele permanente și temporare mari (256 MB) utilizate și adresele permanente și temporare foarte mari (4 GB) utilizate.

Pentru a vedea starea generală a sistemului, faceți următoarele:

1. Din Navigatorul iSeries, expandați **Conexiunile mele**.
2. Faceți clic dreapta pe server și faceți clic pe **Stare sistem**.

Apare fereastra Stare sistem. Pentru mai multe informații despre această fereastră, vedeți ajutorul online al Navigatorului iSeries.

Verificare stare disc:

Câteodată este posibil să doriți să verificați performanța unităților de disc de pe sistemul dumneavoastră sau să vizualizați informații despre starea lor.

Pentru a vedea fereastra Stare disc, faceți următoarele:

1. În Navigator iSeries expandați **Conexiunile mele** → **Faceți clic dreapta pe server** → **Spațiu de disc** → **fișa Stare sistem**.
2. În fereastra Spațiu de disc, faceți clic dreapta pe **Stare disc**. Se deschide fereastra Stare disc.

Puteți utiliza opțiunea **Personalizează această vizualizare** → **Coloane** a ferestrei Stare disc pentru a vizualiza informațiile următoare:

- Cantitatea citită (KB)
- Cantitatea scrisă (KB)
- Procentul ocupat
- Comprimare
- Pool de discuri
- Cereri I/O
- Procent utilizat
- Stare protecție
- Tip protecție
- Cereri de citire

- Dimensiune cerere (KB)
- Dimensiune (MB)
- Tip
- Cereri de scriere

Gestionarea joburilor

Așa cum orice administrator de gestionare de lucru știe, gestionarea joburilor înseamnă mai mult decât plasarea joburilor în așteptare și mutarea joburilor de la coada de joburi la coada de joburi. Acest subiect vorbește despre cele mai comune operații de gestionare de job la fel bine ca unele dintre operațiile cele mai implicate care vă pot ajuta la îmbunătățirea performanțelor sistemului dumneavoastră.

Operații de job obișnuite

Aceste informații discută despre cele mai comune operații pe care le puteți realiza cu joburi. Vă furnizează instrucțiuni pentru ambele, atât pentru Navigator iSeries (când este disponibil), cât și pentru interfața bazată pe caractere.

Pornirea unui job:

Joburile interactive sunt pornite când utilizatorul se înscrie la o stație de lucru. Porniți joburile prestart și joburile batch prin folosirea Navigator iSeries sau interfața bazată pe caractere, depinzând de circumstanțe.

Pornirea unui job batch care așteaptă în coada de joburi:

Ocazional, este posibil să fiți nevoit să forțați un job să pornească imediat. În timp ce mutarea unui job într-o coadă de joburi care nu este ocupată este cea mai eficientă metodă de a realiza acest lucru, există și alte metode pe care le puteți utiliza.

Pentru a porni un job batch, întâi verificați starea cozii de joburi în care se află jobul și determinați dacă are sens mutarea jobului în altă coadă, în situația dumneavoastră. (**Conexiunile mele** → **server** → **Control funcționare** → **Cozi de joburi** → **Cozi de joburi active sau Toate cozile de joburi**)

Dacă mutarea jobului în altă coadă nu este posibilă, puteți reține joburile care rulează și apoi muta jobul pe care aveți nevoie să îl porniți cu prioritate. Totuși, fiți prudenți când utilizați această metodă deoarece joburile reținute sunt încă incluse în numărul maxim de joburi care pot fi active.

Pentru a modifica prioritatea jobului și pentru a indica momentul când acesta ar trebui să ruleze, faceți următoarele:

1. Faceți clic dreapta pe job și faceți clic pe **Proprietăți**.
2. În fereastra Proprietăți job, faceți clic pe fișa **Coadă de joburi**.
3. Modificați **Prioritatea în coada de joburi** cu o prioritate mai mare (0 este cea mai mare).
4. Setati **Când să fie pus jobul la dispoziție pentru rulare** fie pe Acum sau specificați data și ora.
5. Faceți clic pe **OK**.

Pornirea unui job prestart:

Joburile prestart pornesc, de obicei, în același moment în care este pornit subsistemul. Porniți manual un job prestart când toate joburile prestart au fost oprite de către sistem din cauza unei erori sau nu au fost niciodată pornite în timpul pornirii subsistemului din cauza STRJOBS (*NO) din intrarea de job prestart. Pentru a porni un job prestart, utilizați interfața bazată pe caractere.

Comandă: Pornire joburi prestart (STRPJ)

Comanda STRPJ nu ar trebui să fie utilizată până când nu s-a terminat pornirea subsistemului înrudit. Pentru a vă asigura că jobul prestart necesar pornește cu succes, codati o buclă de întârziere cu o reîncercare dacă comanda STRPJ eșuează.

Numărul de joburi prestart care pot fi active în același timp este limitat de atributul MAXJOBS din intrarea de job prestart și de atributul MAXJOBS pentru subsistem. Atributul MAXACT din intrarea de comunicații controlează numărul de cereri de pornire program care pot fi servite prin intrarea de comunicații în același timp.

Notă: Dacă ați specificat *NO în atributul STRJOBS, nu pornește nici un job prestart pentru intrarea jobului prestart când subsistemul pornește. Rularea comenzii STRPJ nu cauzează modificarea valorii parametrului **STRJOBS**.

Exemplu: Acest exemplu pornește joburi prestart pentru intrarea de job prestart PJPGM din subsistemul SBS1. Subsistemul SBS1 trebuie să fie activ când este lansată această comandă. Numărul de joburi pornite este numărul specificat în valoarea INLJOBS a intrării jobului prestart PJPGM. Subsistemul pornește programul PJPGM din biblioteca PJLIB.

```
STRPJ  SBS(SBS1)  PGM(PJLIB/PJPGM)
```

Terminarea unui job:

Puteți utiliza Navigatorul iSeries sau interfața bazată pe caractere pentru a opri un job. Jobul poate fi activ sau poate fi într-o coadă de joburi. Puteți opri jobul imediat sau specificând un interval de timp astfel încât să poată avea loc procesarea de oprire-job.

Navigator iSeries:

Pentru a utiliza Navigatorul iSeries, faceți următoarele:

1. În Navigatorul iSeries, expandați **Control funcționare** → **Joburi active** .
2. Localizați jobul pe care doriți să îl opriți.
3. Faceți clic dreapta pe job și faceți clic pe **Ștergere/Oprire**.
4. Terminați în fereastra Confirmare Ștergere/Oprire și faceți clic pe **Ștergere**.

Interfața bazată pe caractere:

Comandă: Oprire job (ENDJOB)

Dacă nu știți numele jobului pe care doriți să îl opriți, puteți utiliza una din următoarele comenzi pentru a găsi numele jobului:

- Gestionare joburi active (WRKACTJOB)
- Gestionare joburi de utilizator (WRKUSRJOB)
- Gestionare joburi lansate în execuție (WRKSBMJOB)
- Gestionare joburi de subsistem (WRKSBSJOB)
- Oprire subsistem (ENDSBS) Această comandă oprește toate joburile din subsistem.
- Oprire sistem (ENDSYS) Această comandă oprește majoritatea activității din sistem și părăsește sistemul într-o condiție în care doar consola este activă în subsistemul de control.
- Oprire alimentare sistem (PWRDWN SYS) Această comandă pregătește sistemul pentru oprire și apoi pornește secvența de oprire alimentare.

Este posibil ca un job să fie oprit fie imediat sau într-o manieră controlată. Este recomandat să încercați întotdeauna să opriți jobul într-o manieră controlată.

Oprirea unui job: controlată:

Oprirea unui job într-o manieră controlată permite programelor care rulează în job să își realizeze curățarea de sfârșit de job. Poate fi specificat un timp de întârziere pentru a permite jobului să se oprească într-o manieră controlată. Dacă timpul de întârziere se termină înainte ca jobul să se termine, jobul este oprit imediat.

Orice aplicație care necesită realizarea unei curățări de sfârșit de job se oprește într-o manieră controlată. Există trei metode prin care o aplicație poate detecta aceasta:

Extragerea sincronă a Stării de terminare

La anumite momente, o aplicație poate verifica sincron Starea de terminare a jobului în care rulează. Puteți extrage starea de terminare a jobului lansând comanda CLRTVJOBA (Retrieve Job Attributes - Extragere atribute job). În plus, puteți utiliza unul din mai multe API-uri care extrag starea de terminare a jobului. Puteți afla mai multe informații despre aceste API-uri în raportul de experiență, *Attribute control funcționare job*

Verificați sincron codurile retur majore și minore după o operație de I/E

Pentru afișarea atât I/E, cât și I/E de comunicații ICF, un cod retur major de 02 sau un cod retur major de 03 cu un cod retur minor de 09 indică faptul că jobul se termină într-o manieră controlată.

Tratarea semnalului asincron SIGTERM

Unele aplicații utilizează un program de tratare a semnalului pentru a îmbunătăți curățarea aplicației atunci când jobul este oprit. Sistemul generează semnalul asincron SIGTERM pentru jobul care este oprit, când jobul se oprește controlat și toate condițiile următoare sunt împlinite:

- Jobul este activat pentru semnale
- Jobul este un program de tratare a semnalelor care este stabilit pentru semnalul SIGTERM
- Jobul rulează momentan în faza problemă

Dacă oricare din condițiile de mai sus nu este îndeplinită, semnalul SIGTERM nu este generat pentru jobul care este oprit.

Când un job care este oprit într-o manieră controlată are o procedură de tratare a semnalelor pentru semnalul asincron SIGTERM, atunci semnalul SIGTERM este generat pentru acel job. Când procedura de tratare a semnalelor pentru semnalul SIGTERM preia controlul, aceasta poate întreprinde acțiunile corespunzătoare pentru a permite aplicației să fie oprită într-o manieră controlată.

Oprire job: imediat:

Utilizați opțiunea de oprire a unui job imediat doar dacă o oprire controlată nu a avut succes. Când un job se oprește imediat, puteți obține rezultate nedorite precum date de aplicație parțial actualizate.

Înainte de a încheia un job, ar trebui să verificați că nici o unitate de lucru logică nu este într-o stare nerezolvată din cauza unei operații de comitere în două faze care este în curs de desfășurare. Dacă este, atunci valoarea opțiunii de comitere Acțiune la ENDJOB poate avea un impact puternic asupra procesării ENDJOB. Această opțiune face parte din API-ul QTNCHGCO (Change Commitment Options - Modificare opțiuni de comitere). De exemplu, dacă opțiunea de comitere Acțiune la ENDJOB este valoarea implicită WAIT, acest job va fi reținut și nu se va efectua procesarea jobului până când nu se termină operația de control comitere. Aceasta asigură integritatea bazei de date pentru toate sistemele înrudite.

Când utilizați o opțiune de oprire imediată, sistemul realizează o procesare minimă de oprire a jobului, care poate include:

- Închiderea fișierelor de bază de date
- Punerea istoricului de job într-o coadă de ieșire spool
- Curățarea obiectelor interne din sistemul de operare
- Afișarea ecranului oprire job (pentru joburi interactive)
- Efectuarea procesării de control de comitere

Găsirea joburilor:

Este important să înțelegeți cum să procedați pentru a găsi joburi pe serverul dumneavoastră iSeries. Oricare ar fi motivul, la un moment dat este posibil să aveți nevoie de anumite informații pentru un anume job.

În Navigatorul iSeries, puteți face o operație de Găsire asupra tuturor joburilor dumneavoastră sau puteți să vă restrângeți căutarea utilizând funcția Includere urmată de Găsire. Funcția Includere vă permite să puneți limitări asupra a ceea ce este afișat în Navigatorul iSeries. De exemplu, în loc să faceți o operație de Găsire pe sute de joburi, puteți rula o funcție de Includere pentru a afișa doar anumite tipuri de joburi. Sau, puteți afișa doar de la joburile cu anumit ID de utilizator.

Din punct de vedere al performanței, dacă aveți un număr mare de joburi în sistem, se recomandă utilizarea funcției Includere pentru a restrânge numărul de joburi căutate. Dacă aveți o mulțime de joburi pe sistem, căutând prin toate poate să rețină performanța sistemului.

Notă: Puteți utiliza funcțiile Găsire și Includere peste tot în Control funcționare, oriunde găsiți joburi. Puteți de asemenea să folosiți uneltele pentru a găsi cozi de joburi, subsisteme și pool-uri de memorie în aceeași manieră. Țineți minte că trebuie înainte să faceți clic în zona unde doriți să căutați, pentru a putea folosi aceste unelte.

Navigatorul iSeries:

Pentru a găsi un job folosind opțiunea **Găsire (Ctrl+F)**, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active**.
2. În meniul Editare, faceți clic pe **Găsire (Ctrl+F)**.
3. În câmpul **Căutare**, tastați ID-ul de job pe care doriți să-l găsiți (de exemplu, Qqqtemp1). Toate coloanele jobului caută jobul dumneavoastră.
4. Faceți clic pe **Găsire**. Navigatorul iSeries va evidenția jobul dumneavoastră o dată ce va fi găsit.

De reținut: Numele de joburi sunt sensibile la majuscule doar atunci când sunt incluse între ghilimele (de exemplu, "Jobul meu"). Dacă numele jobului nu este între ghilimele, atunci nu este sensibil la majuscule.

Limitarea informațiilor care sunt afișate:

Pentru a limita informațiile care sunt afișate, utilizați funcția Includere.

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active** sau **Joburi de server**.
2. Din meniul Vizualizare, faceți clic pe **Personalizarea acestei vizualizări**, apoi **Includere**. Apare fereastra Includere.
3. În fereastra Includere, selectați opțiunile cu care doriți să căutați jobul dumneavoastră.
4. Faceți clic pe **OK**.

Interfața bazată pe caractere:

Pentru a găsi un job în sistem, utilizați fie comanda WRKACTJOB (Work with Active Job - Gestionare joburi active), WRKUSRJOB (Work with User Job - Gestionare joburi utilizator) sau comanda WRKSBMJOB (Work with Submitted Job.Gestionare joburi lansate în execuție).

Vizualizarea joburilor din coada de joburi:

Cozile de joburi filtrează puțin din lucrul care este procesat în controlul funcționării (de exemplu, unele joburi batch). A avea posibilitatea de a vedea joburilor din coada de joburi vă înseamnă a vedea ce joburi așteaptă să fie trimise unui subsistem.

Navigatorul iSeries:

Pentru a vedea joburile din coada de joburi, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Cozi de job** → **Cozi de joburi active** sau **Toate cozile de job**.

2. Faceți clic pe coada de joburi cu care doriți să afișați joburile (de exemplu, Jobqueue1). Apar joburile din coada de joburi.

Interfața bazată pe caractere:

Comandă: Gestionare coadă de joburi (WRKJOBQ)

Această comandă afișează o listă cu toate cozile de joburi disponibile în sistem. După ce ați localizat coada de joburi care conține jobul dumneavoastră puteți selecta opțiunea **5=Gestionare** și afișa toate joburile din coada de joburi.

Puteți utiliza, de asemenea, comanda Work with Subsystems Job - Gestionare joburi de subsistem pentru a afișa o listă a cozilor de joburi și joburile lor respective.

Comandă: Gestionare job de subsistem (WRKSBSJOB) SBS(*JOBQ)

Vizualizarea joburilor din subsistem:

Subsistemele coordonează fluxul de lucru și resursele folosite de un job în rulare. Navigatorul iSeries vă permite să vedeți ce joburi sunt momentan active (dar nu neapărat în curs de rulare) în subsistem.

Navigatorul iSeries:

Pentru a vizualiza joburile din subsistem, urmați acești pași:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Subsisteme** → **Subsisteme active**.
2. Faceți clic pe subsistemul care are joburile pe care doriți să le afișați.

Interfața bazată pe caractere:

Comandă: Gestionare joburi active (WRKACTJOB SBS(ume subsistem))

Comandă: Gestionare descrieri de subsistem (WRKSBSD)

Utilizați comanda Work with Subsystem Descriptions - Gestionare descrieri de subsistem pentru a afișa o listă de subsisteme. După ce găsiți subsistemul care conține jobul dumneavoastră, utilizați opțiunea **8=Gestionare joburi de subsistem** pentru a afișa informațiile jobului.

Notă: Subsistemul trebuie să fie activ pentru a afișa informațiile jobului.

Vizualizare atribute job:

Atributele jobului conțin informații despre cum sunt procesate joburile. Ele sunt inițial specificate când jobul este creat. Unele dintre atribute provin din descrierea de job. După ce jobul este creat, atributele jobului pot fi vizualizate și gestionate prin Control funcționare din Navigatorul iSeries. Paginile cu proprietățile jobului din Navigatorul iSeries ușurează jobul unui operator de sistem furnizând funcții eficiente și ușor de utilizat pentru gestionarea joburilor.

Navigatorul iSeries:

Pentru a vizualiza atributele de job, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active** sau **Joburi de server**, în funcție de tipul de job cu care doriți să lucrați.
2. Găsiți jobul pe a cărui proprietăți doriți să le vedeți sau modificați.
3. Faceți clic dreapta pe **Nume job** și faceți clic pe **Proprietăți**.

Atributele jobului pot fi văzute de către orice utilizator, dar pot fi modificate doar de către un utilizator cu autorizarea corespunzătoare. În mod asemănător, un utilizator autorizat poate gestiona joburile prin acțiunile jobului. Atributele pentru joburi de sistem nu pot fi modificate din Navigatorul iSeries. Totuși, prioritatea de rulare a unor joburi de sistem poate fi modificată din interfața bazată pe caractere utilizând comanda CHGSYSJOB (Change System Job - Modificare job de sistem).

Interfața bazată pe caractere:

Comandă: Gestionare job (WRKJOB) Când jobul este activ puteți vizualiza următoarele informații: atributele de rulare a jobului, informații despre stiva de apeluri, informații de blocare job, informații despre lista de biblioteci, informații despre istoricul jobului, informații despre fișier deschis, informații despre înlocuire fișier, starea de control comitere, starea comunicațiilor, informații grup de activare, informații mutex și informații despre firul de execuție

Comandă: Afișare job (DSPJOB)

Această comandă afișează următoarele informații despre job: atributele stării jobului, atributele definiției jobului, atributele rulării jobului, informații fișier pus în spool, informații despre istoricul jobului, informații despre stiva de apeluri, informații despre blocarea jobului, informații despre lista de biblioteci, informații despre fișierul deschis, informații despre înlocuirea fișierului, informații despre starea de comitere, informații despre starea comunicațiilor, informații despre grupul de activare, informații mutex, informații despre firul de execuție, bibliotecă media și informații despre atribute.

Vizualizare stive de apeluri:

Puteți vizualiza informații despre stiva de apeluri a unui fir de execuție sau a unui job utilizând fie Navigatorul iSeries™ sau interfața bazată pe caractere.

Navigatorul iSeries:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active sau joburi de server**, în funcție de tipul de job cu care doriți să lucrați.
2. Faceți clic dreapta pe numele jobului și apoi faceți clic pe **Detalii** → **Stivă de apeluri**.

Dacă doriți să vedeți o stivă de apeluri pentru un fir de execuție, urmați acești pași:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active sau joburi de server**, în funcție de tipul de job cu care doriți să lucrați.
2. Faceți clic dreapta pe numele jobului și apoi faceți clic pe **Detalii** → **Fire de execuție**.
3. Dintr-o listă de fire de execuție, faceți clic dreapta un anumit fir de execuție și apoi faceți clic pe **Detalii** → **Stivă de apeluri**.

Dacă rulați sub un profil utilizator cu autorizarea specială *SERVICE și vreți să vedeți intrări suplimentare pentru LIC și i5/OS PASE Kernel, din fereastra Stivă de apeluri, utilizați opțiunea Include din fereastra Personalizarea acestei vizualizări. (**Vizualizare meniu** → **Personalizarea acestei vizualizări** → **Include**)

Interfața bazată pe caractere:

Comandă: Gestionare joburi (WRKJOB) sau Afișare joburi (DSPJOB)

Selectați opțiunea 11: Afișare stivă de apeluri, dacă este activă.

Dacă doriți să vizualizați o stivă de apeluri pentru un fir de execuție, după lansarea comenzii WRKJOB sau DSPJOB, selectați opțiunea 20: Gestionare fire de execuție, dacă sunt active. Apoi, selectați opțiunea 10: Opțiunea Afișare stivă de apeluri pentru firul de execuție selectat.

Punerea unui job în coada de joburi:

Joburile sunt puse în coada de joburi fie prin mutarea unui job existent dintr-o coadă în alta, sau prin lansarea unui nou job. Utilizați Navigatorul iSeries pentru a face mutarea între cozi. Utilizați interfața bazată pe caractere pentru a lansa un nou job.

Navigatorul iSeries:

Pentru a utiliza interfața Navigatorului iSeries, jobul trebuie să existe deja în altă coadă de joburi. Apoi puteți muta jobul dintr-o coadă în alta. (Pentru a pune un job nou într-o coadă de joburi, utilizați interfața de linie de comandă.)

1. În Navigatorul iSeries, expandați **Control funcționare** → **Cozi de joburi** → **Toate cozile de joburi**.
2. Faceți clic dreapta pe jobul pe care doriți să îl mutați. Se deschide fereastra Mutare unde puteți specifica coada de destinație.

Interfața bazată pe caractere:

Următoarea este o listă de metode ale interfeței bazată pe caractere pentru punerea unui job nou într-o coadă de joburi nouă.

- Lansare job (SBMJOB): Permite unui job care rulează să pună un alt job într-o altă coadă care să fie rulat mai târziu ca job batch. Doar un element de date de cerere poate fi pus în coada de mesaje a noului job. Datele de cerere pot fi o comandă CL dacă intrarea de rutare utilizată pentru job specifică un program de procesare a comenzilor CL (cum ar fi programul QCMD livrat de IBM).
- Adăugare intrare de planificare job (ADDJOBSCDE): Sistemul lansează automat un job în coada de joburi la ora și data specificate în intrarea de planificare job.
- Lansare joburi de bază de date (SBMDBJOB): Lansează joburi în cozi de joburi astfel încât ele să ruleze ca joburi batch. Fluxul de intrare este citit fie dintr-un fișier fizic de bază de date sau dintr-un fișier bază de date logic care are un format cu o singură înregistrare. Această comandă vă permite să specificați numele fișierului bază de date și a membrului său, numele cozii de joburi care va fi utilizată și să decideți dacă joburile care sunt lansate pot fi afișate de către comanda WRKSBMJOB (Work with Submitted Jobs - Gestionare joburi lansate).
- Pornire cititor bază de date (STRDBRDR): Citește un flux de intrare batch dintr-o bază de date și pune unul sau mai multe joburi în cozi de joburi.
- Transfer job (TFRJOB): Mută jobul curent în altă coadă de joburi dintr-un subsistem activ.
- Transfer job batch (TFRBCHJOB): Mută jobul curent în altă coadă de joburi.

Mutarea unui job într-o coadă de joburi diferită:

Există multe motive pentru care ați putea dori să mutați un job în altă coadă. De exemplu, unele joburi devin restante în coadă din cauza unui job care are o durată mare de rulare. Probabil timpul de rulare planificat al jobului este în conflict cu un nou job care are o prioritate mai mare. O metodă de a gestiona această situație este să fie mutate joburile aflate în așteptare în altă coadă care nu este atât de ocupată.

Puteți utiliza fie interfața Navigatorului iSeries sau interfața bazată pe caractere pentru a muta jobul dintr-o coadă în alta.

Navigatorul iSeries:

Pentru a utiliza Navigatorul iSeries, faceți următoarele:

1. În Navigatorul iSeries, expandați **Control funcționare** → **Cozi de joburi** → **Toate cozile de joburi**.
2. Localizați și deschideți coada care conține jobul momentan.
3. Faceți clic dreapta pe jobul pe care doriți să îl mutați. Se deschide fereastra Mutare unde puteți specifica coada destinație.

Notă: Dacă doriți să mutați mai mult de un job din această coadă, țineți apăsată tasta CTRL în timp ce faceți clic pe fiecare job. Apoi faceți clic dreapta și apoi faceți clic pe **Mutare**.

- Joburile care așteaptă să ruleze sunt mutați în aceeași poziție relativă în coada destinație (de exemplu, joburile cu o prioritate de coadă de joburi de 3 sunt mutate după toate celelalte joburi cu prioritatea 3 care așteaptă să ruleze în coada destinație).
- Joburile care sunt reținute rămân reținute și sunt puse în aceeași poziție relativă în coada destinație (de exemplu, joburile reținute cu prioritatea de coadă de joburi 3 sunt mutate după toate celelalte joburi reținute cu prioritatea 3 din coada destinație).
- Joburile care sunt planificate să ruleze sunt mutate în coada destinație și orele lor planificate rămân nemodificate.

Interfața bazată pe caractere:

Comandă: Modificare job (CHGJOB)

Exemplu: Următorul exemplu mută jobul JOBA în coada de joburi JOBQB

CHGJOB JOB(JOBA) JOBQ(LIBA/JOBQB)

Mutarea unui job la o prioritate mai mare în coada de joburi:

Toate joburile din coada de joburi așteaptă la rând pentru procesare. Pe măsură ce fiecare job din coada de joburi se efectuează, următorul job din rând începe. Ordinea de procesare a joburilor din coadă depinde de prioritatea jobului și de numărul maxim de joburi care pot rula simultan în subsistem.

Câteodată importanța unui job se schimbă când începe ciclul său de viață. El poate să crească sau să descrească în prioritate, în relație cu alte joburi. Deoarece apar aceste schimbări, aveți nevoie să știți cum puteți schimba prioritatea unui job în coada de joburi.

Prioritatea unui job într-o coadă de joburi ajută să determinați când un job începe să ruleze în subsistem. Un interval de la zero la nouă (zero fiind cel mai important) determină prioritatea unui job în coada de joburi.

Navigatorul iSeries:

Puteți utiliza Navigatorul iSeries pentru a modifica prioritatea unui job din coada de joburi.

1. Expandați **Conexiunile mele** → **server** → **Control funcționare** → **Cozi de joburi** → **Cozi de joburi active sau Toate cozile de joburi** → **Coada de joburi în care se află jobul dumneavoastră**.
2. Faceți clic dreapta pe job și faceți clic pe **Proprietăți**.
3. În fereastra Job - Proprietăți, faceți clic pe fișa **Coadă de joburi**.
4. Din lista **Priorități în coada de joburi**, selectați un număr cu prioritate mai mare (sau mai mică). Prioritatea cozii de joburi are intervalul între 0-9, cu 0 fiind cel cu prioritatea cea mai mare.
5. Faceți clic pe **OK**. Prioritatea cozii de joburi a fost schimbată pentru jobul dumneavoastră. De exemplu, modificarea unui job cu prioritate 4 la o prioritate 3 mută jobul în partea de jos a listei de joburi care are o prioritate 3.
6. Apăsați F5 pentru a reîmprospăta fereastra Coadă de joburi.

Interfața bazată pe caractere:

Comandă: Modificare job (CHGJOB)

Parametru: JOBPTY

Exemplu: Această comandă modifică prioritatea de planificare pentru jobul PAYROLL cu 4. Deoarece este specificat doar un nume simplu al jobului, poate exista doar un job numit PAYROLL în sistem. Dacă există mai mult de unul, valoarea implicită DUPJOBPT(*SELECT) duce la afișarea unui panou de selecție într-un job interactiv.

CHGJOB JOB(PAYROLL) JOBPTY(4)

Sugestii pentru setarea priorităților pentru joburi:

Prioritățile pentru joburile care rulează în medii batch ar trebui, în mod normal, să fie mai mici decât prioritățile pentru joburile din medii interactive. De asemenea, felia de timp ar trebui să fie destul de mică astfel încât un program care rulează în bucle (luping) să nu domine timpul procesorului și un nivel de activitate.

Este posibil să doriți ca prioritatea pentru joburile operatorului de sistem să fie mai mare decât prioritățile altor joburi astfel încât operatorul de sistem să poată răspunde eficient la necesitățile sistemului.

Dacă utilizați QCTL ca subsistem de control, operatorul rulează automat la o prioritate mai mare după semnarea la consolă. Aceasta deoarece QCTL face rutarea jobului de consolă utilizând clasa QCTL, care specifică o prioritate mai mare.

O altă cale prin care puteți seta sistemul dumneavoastră astfel încât operatorul să poată rula la o prioritate mai mare ar fi să faceți următoarele:

1. Adăugați o intrare de rutare la subsistem cu date unice de rutare și specificați clasa QSYS/QCTL.
2. Creați o descriere nouă de job pentru operator, specificând aceleași date de rutare unice pe care le-ați utilizat în intrarea de rutare.
3. Modificați profilul utilizator al operatorului pentru a specifica noua descriere de job.
4. Acum, când operatorul se semnează pe acel subsistem, jobul va ruta utilizând clasa QCTL, care specifică o prioritate mai mare decât clasa utilizată de joburile interactive normale.

Prioritatea de rulare a jobului este cea mai mare prioritate la care orice fir de execuție din job poate rula. Fiecare fir de execuție poate avea propria prioritate de fir de execuție mai mică decât prioritatea jobului. Comanda CHGJOB (Change Job - Modificare job) va modifica numai prioritatea jobului. API-ul QWTCHGJB (Change Job - Modificare job) poate fi utilizat fie pentru modificarea priorității jobului sau a unui fir de execuție.

Lansați un job o dată:

Când trebuie să lansați un job o dată, imediat sau la o dată și o oră planificată, folosiți comanda Lansare job (SBMJOB). Această metodă pune jobul imediat în coada de joburi.

Pentru a lansa o dată un job batch, folosiți interfața bazată pe caractere.

Comanda: Lansare job (SBMJOB)

Comanda SBMJOB lansează un job la o coadă de joburi batch prin specificarea unei descrieri de job și prin specificarea unei comenzi CL sau date de cerere, sau prin specificarea datelor de rutare pentru a porni un program. Dacă doriți să rulați o singură comandă CL din jobul batch, folosiți parametrul **CMD** pe SBMJOB, care face verificare de sintaxă și permite promptare.

Exemplu: În următorul exemplu, comanda SBMJOB lansează un job numit WSYS, folosind descrierea de job QBATCH, la coada de joburi QBATCH. Parametrul **CMD** dă comanda CL care va rula în job.

```
SBMJOB JOB(QBATCH) JOB(WSYS) JOBQ(QBATCH) CMD(WRKSYSSTS)
```

Vizualizarea informațiilor despre afinitatea joburilor:

Fiecare job de pe iSeries conține informații despre afinitatea memoriei și procesorului.

Informațiile de afinitate descriu dacă firele de execuție vor avea afinitate la același grup de procesoare și memorie ca și firul de execuție inițial, atunci când vor fi pornite. Specifică și gradul la care sistemul încearcă să mențină afinitatea între șirele de execuție și subsetul de resurse de sistem la care sunt alocate. În plus, informațiile de afinitate specifică dacă un job este grupat cu alte joburi astfel încât ele să aibă afinitate la același subset de resurse ale sistemului.

Grupând împreună firele de execuție care împart un set comun de date în spațiul de stocare principal, ratele de accesare a memorie și a memoriei cache ale sistemului dumneavoastră se pot îmbunătăți.

Navigatorul iSeries:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active**.

Notă: Puteți vizualiza informațiile de afinitate ale unui job din orice locație din Control funcționare unde puteți vizualiza joburi.

2. Faceți clic dreapta pe jobul pe care doriți să îl vizualizați și faceți clic pe **Proprietăți**.
3. În pagina Resurse, puteți vizualiza informații despre **Afinitatea procesorului și a memoriei**.

Interfața bazată pe caractere:

Comandă: Gestionare job (WRKJOB)

Selectați opțiunea 3: Afișare atribute rulare job, dacă este activă

Gestionarea descrierii de job

Din moment ce o descriere de job colectează un set specific de atribute înrudire-job, aceeași descriere de job poate fi folosită de către joburi multiple. Cu toate acestea, dacă folosiți o descriere de job, nu trebuie să specificați aceeași parametri în mod repetat pentru fiecare job. Puteți crea descrieri de job pentru a descrie joburi batch sau joburi interactive. Puteți de asemenea crea descrieri unice pentru fiecare utilizator al sistemului. Descrierile de job sunt create și gestionate prin folosirea interfeței bazată pe caractere.

Crearea unei descrieri de job:

Utilizați interfața bazată pe caractere și comanda WRKJOB (Work With Job Description - Gestionare descrieri de job) sau comanda CRTJOB (Create Job Description - Creare descriere de job) pentru a crea descrieri de job.

Comandă: Creare descriere de job (CRTJOB)

Exemplu: În acest exemplu, este creată o descriere de job numită INT4 în biblioteca curentă a utilizatorului. Această descriere de job este pentru joburi interactive și este utilizată de către Departamentul 127. Când vă semnați, trebuie să vă tastați parola. Caracterele QCMDI sunt utilizate ca date de rutare care sunt comparate cu tabela de rutare a subsistemului unde este rulat jobul. Toate mesajele de interogare sunt comparate cu intrările din lista de răspunsuri a sistemului pentru a determina dacă este emis automat un răspuns.

```
CRTJOB JOB(INT4) USER(*RQD) RTGDTA(QCMDI)
      INQMSGRPY(*SYSRPLY)
      TEXT('Interactiv #4 JOB pentru Departamentul 127')
```

Această comandă creează o descriere de job numită BATCH3 în biblioteca curentă a utilizatorului. Joburile care utilizează această descriere sunt puse în coada de joburi NIGHTQ. Prioritatea pentru joburile care utilizează această descriere și ieșirea lor spool este 4. QCMDI reprezintă datele de rutare care sunt comparate cu intrările din tabela de rutare a subsistemului unde rulează jobul. Codul de contabilizare NIGHTQ012345 este utilizat la înregistrarea statisticilor de contabilitate pentru joburile care utilizează această descriere de job.

```
CRTJOB JOB(BATCH3) USER(*RQD) JOBQ(NIGHTQ) JOBPTY(4)
      OUTPTY(4) ACGCDE(NIGHTQ012345) RTGDTA(QCMDI)
      TEXT('Batch #3 JOB pentru munca pe timp de noapte cu prioritate mare')
```

Notă: Valorile din descrierea de job sunt de obicei utilizate ca valori implicite pentru parametrii corespunzători din comenzile BCHJOB (Batch Job - Job batch) și SBMJOB (Submit Job - Lansare job), atunci când parametrii lor nu sunt specificați. Valorile din descrierea de job pot fi suprascrise de valorile specificate în comenzile BCHJOB și SBMJOB.

Modificarea unei descrieri de job:

Utilizați interfața bazată pe caractere și comanda WRKJOB (Work With Job Description - Gestionare descrieri de job) sau comanda CHGJOB (Change Job Description - Modificare descriere job) pentru a modifica descrieri de job.

Comandă: Modificare descriere de job (CHGJOB)

Toate joburile care folosesc acea descriere de job, pornite după modificarea descrierii jobului, sunt afectate. Dacă ați modificat un parametru de job cu ceva diferit de ceea ce este specificat în descrierea de job, acel parametru nu va fi afectat.

Utilizați o descriere de job:

Modul cel mai comun de a utiliza o descriere de job este specificându-l în comanda SBMJOB (Submit Job - Lansare job). Specificați în Parametrul Descriere job (**JOB**) descrierea de job pe care doriți ca acest job să o utilizeze. Când definiți un job batch, puteți utiliza descrierea de job în unul din două moduri:

- Utilizați o descriere de job specificată fără a înlocui nici unul din atributele sale. De exemplu:
`SBMJOB JOB(OEDAILY) JOB(QBATCH)`
- Utilizați o descriere de job specificată înlocuind unele din atribute (utilizând comanda BCHJOB sau SBMJOB). De exemplu, pentru a înlocui înregistrarea în istoric a mesajului din descrierea de job QBATCH, specificați:
`SBMJOB JOB(OEDAILY) JOB(QBATCH) LOG(2 20 *SECLVL)`

Următoarele reprezintă comenzi suplimentare care suportă parametrul de descriere job:

- Job batch (BCHJOB): Această comandă indică începutul unui job batch într-un flux de intrare batch. Poate specifica, de asemenea, valori diferite pentru atributele unui job în locul celor specificate în descrierea de job sau profilul utilizator pentru acest job. Valorile conținute în descrierea de job sau în profilul utilizator menționat în acea descriere de job sunt utilizate pentru majoritatea parametrilor necodați în comanda BCHJOB.
- Adăugare intrare de job prestart (ADDPJE): Comanda ADDPJE (Add Prestart Job Entry - Adăugare intrare de job prestart) adaugă o intrare de job prestart la descrierea de subsistem specificată. Intrarea identifică joburile prestart care pot fi pornite când este pornit subsistemul sau când este introdusă comanda STRPJ (Start Prestart Jobs - Pornire joburi prestart).
- Adăugare intrare de job autostart (ADDAJE): Comanda ADDAJE (Add Autostart Job Entry - Adăugare intrare de job autostart) adaugă o intrare de job autostart la o descriere de subsistem specificată. Intrarea identifică numele de job și descrierea de job care vor fi utilizate pentru a porni automat un job.
- Adăugare intrare de stație de lucru (ADDWSE): Comanda ADDWSE (Add Work Station Entry - Adăugare intrare stație de lucru) adaugă o intrare de stație de lucru la descrierea de subsistem specificată. Fiecare intrare descrie una sau mai multe stații de lucru care sunt controlate de către subsistem. Stațiile de lucru identificate în intrările de stație de lucru au permisiunea de a se semna sau de a intra în subsistem și de a rula joburi.

Notă: Nu puteți înlocui nici un atribut al unei descrieri de job pentru joburi autostart, joburi de stație de lucru sau joburi de comunicații.

Controlul sursei de atribute de job:

Atributele pe care subsistemul le alocă joburilor provin din cinci surse; din descrierea jobului, din profilul de utilizator al utilizatorului, o valoare de sistem, jobul care a lansat comanda SBMJOB (Submit Job - Lansare job) și stația de lucru (doar pentru joburi interactive). Dumneavoastră controlați de unde extrage subsistemul atributele jobului respectiv specificând sursa în descrierea jobului. Pentru a modifica o descriere de job, utilizați interfața bazată pe caractere.

Comandă: Modificare descriere job (CHGJOB)

Pentru a controla atributele unui job și pentru a spune subsistemului de unde și când să obțină atribute de job din obiecte de sistem diferite, utilizați una din următoarele:

- *JOB: Spune jobului să își obțină atributele din descrierea jobului.
- *USRPRF: Spune jobului să își ia atributele din profilul de utilizator al utilizatorului.
- *SYSVAL: Spune jobului să își ia atributele din valoarea de sistem.
- *CURRENT: Spune jobului să își ia atributele din jobul care lansează comanda SBMJOB (Submit Job - Lansare job).

- *WRKSTN: Spune jobului să își obțină atributele din stația de lucru cu jobul (doar pentru joburi interactive).

Ștergerea unei descrieri de job:

Utilizați interfața bazată pe caractere și comanda WRKJOB (Work With Job Description - Gestionare descrieri de job) sau comanda DLTJOB (Delete Job Description - Ștergere descrieri de job) pentru a șterge descrieri de job.

Comandă: Ștergere descrieri de job DLTJOB

Notă: Joburile care sunt deja în lucru nu sunt afectate de această comandă.

Gestionarea joburilor batch

Joburile care nu necesită interacțiune cu utilizatorul pentru a rula pot fi procesate ca joburi batch. Un job batch, de obicei, este un job cu o prioritate scăzută și poate necesita un mediu special de sistem în care să ruleze.

Lansarea în execuție a unui job batch:

Din moment ce joburile batch sunt de obicei joburi cu prioritate redusă care necesită un mediu de sistem special în care să ruleze (cum ar fi rularea pe timp de noapte) ele sunt puse în cozi de joburi batch. În coada de joburi jobul batch primește o planificare de runtime și o prioritate. Pentru a pune un job în coada de joburi batch utilizați interfața bazată pe caractere și o comandă sau două.

Comandă: Lansare job (SBMJOB)

Comandă: Lansare job de bază de date (SBMDBJOB)

Diferența din aceste comenzi este sursa jobului:

- Comanda SBJOB pune un job într-o coadă de joburi specificând o descriere de job și specificând o comandă CL sau date de cerere sau specificând date de rutare pentru a rula un program. Dacă doriți să rulați o singură comandă CL într-un job batch, utilizați parametrul CMD din SBJOB, care face verificarea sintaxei și permite promptarea.
- Comanda SBMDBJOB poate fi utilizată pentru a pune un job într-o coadă de joburi batch dintr-un fișier bază de date. Pentru aceste joburi, descrierea de job provine din instrucțiunea BCHJOB din fluxul de intrare.

Exemplu: În următorul exemplu, comanda SBJOB pune un job numit WSYS, utilizând descrierea de job QBATCH, în coada de joburi QBATCH. Parametrul CMD dă comanda CL care va rula în job.

```
SBJOB JOB(QBATCH) JOB(WSYS) JOB(QBATCH) CMD(WRKSYSSTS)
```

Notă: Dacă primiți un mesaj că jobul nu a fost pus, puteți afișa fișierul pus în spool al istoricului de job pentru a găsi erorile. Utilizați comanda WRKJOB. Specificați jobul care nu a fost planificat, selectați opțiunea 4 pentru fișiere puse în spool. Afișați fișierul pus în spool a istoricului de job pentru a găsi erorile.

Utilizarea unui fișier de date inline:

Un fișier de date inline este un fișier de date care este inclus ca parte a unui job batch atunci când jobul este citit de un cititor sau de către o comandă de lansare joburi. Utilizați SBMDBJOB sau STRDBRDR pentru a pune în coada un flux batch CL (flux de comenzi CL pentru a fi executate sau rulate). Acel flux batch CL poate include date care să fie puse în fișiere "temporare" (fișiere inline). Când un job se termină, fișierele inline sunt șterse.

Un fișier de date inline este delimitat în job de către o comandă //DATA la începutul fișierului și de un delimitator sfârșit-de-date la sfârșitul fișierului.

Delimitatorul sfârșit-de-date poate fi un șir de caractere definit de utilizator sau implicit // // trebuie să apară în pozițiile 1 și 2. Dacă datele dumneavoastră conțin // în pozițiile 1 și 2, ar trebui să utilizați un set unic de caractere precum: // *** END OF DATA Pentru a specifica acesta ca fiind un delimitator de sfârșit-de-date unic, parametrul **ENDCHAR** din comanda //DATA ar trebui codat astfel:

ENDCHAR('/// *** END OF DATA')

Notă: Fișierele de date inline pot fi accesate doar în timpul primului pas de rutare a jobului batch. Dacă un job batch conține o comandă TFRJOB (Transfer Job - Transfer job), RRTJOB (Reroute Job - Rerutare job) sau TFRBCHJOB (Transfer Batch Job - Transfer job batch), fișierele de date inline nu pot fi accesate în noul pas de rutare.

Un fișier de date inline poate fi fie denumit fie nedenumit. Pentru un fișier de date inline nedenumit, fie este specificat QINLINE ca numele fișierului din comanda //DATA sau se specifică fără nume. Pentru un fișier de date inline denumit, se specifică un nume de fișier.

Un fișier de date inline denumit are următoarele caracteristici:

- Are un nume unic într-un job. Nici un alt fișier de date inline nu poate avea același nume.
- Poate fi utilizat de mai multe ori într-un job.
- De fiecare dată când este deschis, el este poziționat în prima înregistrare.

Pentru a utiliza un fișier de date inline denumit, trebuie fie să specificați numele fișierului în program sau să utilizați o comandă de suprascriere pentru a modifica numele fișierului specificat în program cu numele fișierului de date inline. Fișierul trebuie să fie deschis doar pentru intrare.

Un fișier de date inline nedenumit are următoarele caracteristici:

- Numele său este QINLINE. (Într-un job batch, toate fișierele de date inline nedenumite primesc același nume.)
- El poate fi utilizat doar o dată într-un job.
- Când sunt incluse mai multe fișiere de date inline nedenumite într-un job, fișierele trebuie să fie în fluxul de intrare în aceeași ordine ca atunci când sunt deschise.

Pentru a utiliza un fișier de date inline nedenumit, faceți una din următoarele:

- Specificați QINLINE în program.
- Utilizați o comandă de suprascriere fișier pentru a modifica numele de fișier specificat în program cu QINLINE.

Dacă limbajul dumneavoastră de nivel înalt necesită nume de fișier unice într-un program, puteți utiliza QINLINE ca nume de fișier doar o singură dată. Dacă aveți nevoie să utilizați mai mult de un fișier de date inline nedenumit, puteți utiliza o comandă de suprascriere fișier în program pentru a specifica QINLINE pentru filiere de date inline nedenumite suplimentare.

Notă: Dacă rulați comenzi condițional și procesați mai mult de un fișier de date inline nedenumit, rezultatele nu pot fi precise dacă este utilizat fișierul de date inline nedenumit greșit.

Considerente deschise pentru fișiere de date inline:

Următoarele considerente se aplică la deschiderea fișierelor de date inline:

- Lungimea înregistrării specifică lungimea înregistrărilor de intrare. (Lungimea înregistrării este opțională.) Când lungimea înregistrării depășește lungimea datelor, este trimis un mesaj programului dumneavoastră. Datele sunt completate cu blancuri. Când lungimea înregistrării este mai mică decât lungimea datelor, înregistrările sunt trunchiate.
- Când este specificat un fișier într-un program, serverul caută fișierul ca pe un fișier de date inline denumit, înainte să caute fișierul într-o bibliotecă. De aceea, dacă un fișier de date inline denumit are același nume ca și fișierul care nu este un fișier de date inline, fișierul de date inline este întotdeauna utilizat, chiar dacă numele fișierului este calificat de un nume de bibliotecă.
- Fișierele de date inline denumite pot fi partajate între programele din același job specificând SHARE(*YES) într-o comandă de creare fișier sau suprascriere fișier. De exemplu, dacă o comandă de suprascriere fișier specificând un fișier numit INPUT și SHARE(*YES) este într-un job batch cu un fișier de date inline numit INPUT, toate

programele care rulează în job, care specifică numele de fișier INPUT vor partaja același fișier de date inline denumit. Fișierele de date inline nedenumite nu pot fi partajate între programele din același job.

- Când utilizați fișiere de date inline, ar trebui să vă asigurați că este specificat tipul corect de fișier în comanda //DATA. De exemplu, dacă fișierul va fi utilizat ca un fișier sursă, tipul fișierului din comanda //DATA trebuie să fie sursă.
- Fișierele de date inline trebuie să fie deschise doar pentru intrare.

Pornirea unui job batch care așteaptă în coada de joburi:

Ocazional, este posibil să fiți nevoit să forțați un job să pornească imediat. În timp ce mutarea unui job într-o coadă de joburi care nu este ocupată este cea mai eficientă metodă de a realiza acest lucru, există și alte metode pe care le puteți utiliza.

Pentru a porni un job batch, întâi verificați starea cozii de joburi în care se află jobul și determinați dacă are sens mutarea jobului în altă coadă, în situația dumneavoastră. (**Conexiunile mele** → **server** → **Control funcționare** → **Cozi de joburi** → **Cozi de joburi active sau Toate cozile de joburi**)

Dacă mutarea jobului în altă coadă nu este posibilă, puteți reține joburile care rulează și apoi muta jobul pe care aveți nevoie să îl porniți cu prioritate. Totuși, fiți prudenți când utilizați această metodă deoarece joburile reținute sunt încă incluse în numărul maxim de joburi care pot fi active.

Pentru a modifica prioritatea jobului și pentru a indica momentul când acesta ar trebui să ruleze, faceți următoarele:

1. Faceți clic dreapta pe job și faceți clic pe **Proprietăți**.
2. În fereastra Proprietăți job, faceți clic pe fișa **Coadă de joburi**.
3. Modificați **Prioritatea în coada de joburi** cu o prioritate mai mare (0 este cea mai mare).
4. Setati **Când să fie pus jobul la dispoziție pentru rulare** fie pe Acum sau specificați data și ora.
5. Faceți clic pe **OK**.

Gestionarea joburilor interactive

Un job interactiv pornește când vă semnați pe sistem sau vă transferați la un job secundar sau de grup. Jobul interactiv se termină când renunțați. Lucrând cu o stație de afișare, interacționați cu sistemul lansând comenzi, utilizând taste funcționale și rulând programe și aplicații. Următoarele informații discută diversele metode pentru gestionarea și controlul joburilor interactive.

Controlul joburilor și a stațiilor de lucru inactive:

Puteți controla durata cât o stație de lucru poate să rămână inactivă, înainte ca subsistemul să trimită un mesaj (numit time-out), specificând un interval de timp în valoarea de sistem QINACTITV, interval Time-out pentru joburi inactive. Controlul joburilor inactive furnizează securitate astfel încât utilizatorii să nu părăsească ecranele inactive în care s-au înregistrat.

Cum determină un subsistem că stația de lucru este inactivă

Subsistemul determină faptul că stația de lucru este inactivă dacă sunt adevărate următoarele:

- Jobul nu a procesat nici o tranzacție suplimentară în timpul intervalului de cronometru.

Notă: O tranzacție se definește ca orice interacțiune cu operatorul, precum defilarea, apăsarea tastei Enter, apăsarea tastelor funcționale și așa mai departe. Tastarea la stația de lucru fără a apăsa tasta Enter nu este considerată o tranzacție. Dacă un job de la stația de lucru nu întrunește criteriile de inactivitate, jobul este considerat activ.

- Starea jobului este așteptare afișare.
- Jobul nu este deconectat.
- Starea jobului nu s-a modificat.
- Subsistemul în care rulează jobul nu este într-o stare restricționată.

Tratarea joburilor inactive

Pentru a trata un job inactiv aflat pe sistem, utilizați valoarea de sistem QINACTMSGQ (Când un job ajunge la time-out). Pentru a determina opțiunile de procesare alegeți din următoarele:

- Setări valoarea de sistem QINACTMSGQ pe un nume de coadă de mesaje.

Dacă specificați un nume de coadă de mesaje pentru valoarea de sistem QINACTMSGQ, un utilizator sau un program poate monitoriza coada de mesaje și poate face ce acțiune este necesară, precum terminarea unui job.

Dacă o stație de lucru cu o pereche secundară de joburi este inactivă, sistemul trimite două mesaje (unul de la fiecare pereche secundară de joburi) către coada de mesaje. Utilizatorul sau programul poate apoi utiliza fie comanda ENDJOB împotriva unuia sau ambelor joburi secundare, sau comanda DSCJOB împotriva jobului activ pe ecran.

- Setări valoarea de sistem QINACTMSGQ pe *DSCJOB.

Dacă specificați *DSCJOB pentru valoarea de sistem QINACTMSGQ, sistemul deconectează toate joburile de la stația de lucru. Sistemul trimite un mesaj care indică faptul că toate joburile de la stația de lucru au fost deconectate de la QSYSOPR sau de la coada de mesaje configurată. (O coadă de mesaje configurată este coada de mesaje specificată în parametrul **MSGQ** al descrierii dispozitivului de afișare. În mod implicit el este QSYS sau QSYSOPR.) Dacă jobul interactiv nu suportă deconectarea jobului (de exemplu, sesiunile TELNET care utilizează descrieri de dispozitive QPADEVxxxx), atunci jobul este oprit.

Continuă să fie trimis un mesaj pentru fiecare interval în care jobul este inactiv.

- Setări valoarea de sistem QINACTMSGQ pe *ENDJOB.

Dacă specificați *ENDJOB pentru valoarea de sistem QINACTMSGQ, sistemul oprește toate joburile de la stația de lucru. Sistemul trimite un mesaj, care indică faptul că toate joburile de la stația de lucru s-au oprit, la QSYSOPR sau la coada de mesaje configurată.

Notă: Joburile pass-through sursă, joburile client VTM (manager de terminal) și joburile de emulare a dispozitivului 3270 sunt excluse din time-out deoarece ele apar mereu inactive. Joburile MRT de mediuSistem/36 sunt de asemenea excluse din moment ce apar ca joburi batch.

Oprirea joburilor interactive:

Există mai multe metode pe care le puteți utiliza pentru a opri un job interactiv.

1. Puteți utiliza Navigatorul iSeries pentru a opri jobul. Din fereastra Confirmare Ștergere/Oprire puteți specifica dacă doriți ca acest job interactiv să fie oprit într-o manieră controlată sau imediat.
2. Puteți utiliza comanda ENDJOB (End Job - Oprire job) a interfeței bazate pe caractere.
3. Pentru a opri un job interactiv imediat utilizând interfața bazată pe caractere, utilizați comanda SIGNOFF (Sign Off - Anulare semnare) de la stația de lucru. Pentru a opri o conexiune prin rețea, utilizați parametrul de oprire conexiune ENDCNN din comanda SIGNOFF.
4. Pentru a deconecta toate joburile de la un dispozitiv, utilizați comanda DSCJOB (Disconnect Job - Deconectare job).

Pentru a utiliza Navigatorul iSeries și fereastra Confirmare Ștergere/Oprire, faceți următoarele:

1. Expandați **Conexiunile mele** → **Sistem punct final** → **Control funcționare** → **Joburi active** .
2. Faceți clic dreapta pe jobul pe care doriți să îl opriți și apoi faceți clic pe **Ștergere/Oprire**. Apare fereastra Confirmare Ștergere/Oprire unde puteți specifica cum și când doriți ca jobul interactiv să se termine.

Notă: Pentru a opri toate joburile interactive asociate cu stația de lucru, sau toate joburile asociate cu grupul (dacă jobul este un job de grup), setați valoarea câmpului **Acțiune pentru joburi interactive înrudite** fie pe Oprire pentru joburile de grup sau Oprire toate (aceasta este echivalentă cu parametrul **ADLINTJOBS** din comanda ENDJOB).

Puteți, de asemenea, să cereți subsistemului să trimită un mesaj către coada de mesaje când un job interactiv a fost inactiv pentru o perioadă de timp specificată. Dumneavoastră, sau un program care monitorizează acea coadă de mesaje, puteți atunci opri sau deconecta jobul.

Deconectarea tuturor joburilor de la un dispozitiv:

Comanda DSCJOB (Disconnect Job - Deconectare job) permite utilizatorului interactiv să deconecteze toate joburile interactive de la stația de lucru și să se întoarcă la ecranul de semnare. Linia comutată este abandonată doar dacă acest lucru este specificat în descrierea dispozitivului de stație de lucru al acestei stații de lucru și dacă nici o altă stație de lucru de pe această linie nu este activă. Dacă jobul este deconectat când se ajunge la intervalul de deconectare din valoarea de sistem QDSCJOBTV, interval de timeout pentru joburi deconectate, jobul este oprit și istoricul jobului nu este inclus cu ieșirea spool a jobului.

Restricții:

1. Un job care este deconectat trebuie să fie un job interactiv.
2. Un job care este reținut nu poate fi deconectat.
3. Un job pass-through nu poate fi deconectat decât dacă utilizatorul a utilizat funcția Sysreq (cerere sistem) pentru a se întoarce la sistemul sursă din sistemul destinație pass-through.
4. Comanda trebuie să fie lansată din jobul care este deconectat, sau emitentul comenzii trebuie să ruleze sub un profil utilizator care este același ca și identitatea de utilizator job a jobului care este deconectat, sau emitentul comenzii trebuie să ruleze sub un profil utilizator care autorizare specială de control job (*JOBCTL).
5. Identitatea de utilizator job este numele profilului utilizator după care un job este cunoscut față de celelalte joburi.
6. Un job nu poate fi deconectat dacă PC Organizer este activ.

Comandă: Deconectare job (DSCJOB)

Considerente pentru deconectarea unui job:

Există mai mulți factori pe care trebuie să îi luați în considerare când deconectați un job.

- O opțiune din meniul Cerere sistem vă permite să deconectați un job interactiv, făcând astfel să apară ecranul de semnare. Opțiunea apelează comanda DSCJOB (Disconnect Job - Deconectare job).
- Când vă conectați din nou la un job, valorile specificate în ecranul de semnare pentru program, meniu și bibliotecă curentă, sunt ignorate.
- Un job care are activă o funcție de organizare PC sau de asistare text PC nu poate fi deconectat.
- Un job TCP/IP TELNET poate fi deconectat dacă sesiunea utilizează o descriere de dispozitiv **denumită specificată de utilizator**. Puteți crea o descriere de dispozitiv denumită specificată de utilizator utilizând una din următoarele căi:
 - Utilizând Stații rețea cu parametrul **DISPLAY NAME**
 - Utilizând suportul Acces client iSeries cu funcția ID a stației de lucru
 - Utilizând punctul de ieșire TCP/IP TELNET Inițializare dispozitiv pentru a specifica un nume de stație de lucru.
- Dacă jobul nu poate fi deconectat din oricare motiv, jobul va fi în schimb oprit.
- Toate joburile deconectate din subsistem se opresc atunci când se oprește subsistemul. Dacă un subsistem se oprește, comanda DSCJOB nu poate fi lansată în oricare din joburile din subsistem.
- Valoarea de sistem QDSCJOBTV (Disconnect Job Interval - Interval deconectare job) poate fi utilizată pentru a indica un interval de timp pentru care un job poate fi deconectat. Dacă se termină intervalul de timp, jobul deconectat se oprește.
- Joburile deconectate care nu au depășit valoarea de sistem QDSCJOBTV se vor opri atunci când subsistemul este oprit sau survine un IPL.

Evitarea unei funcții cu durată mare de rulare, de la o stație de lucru:

Pentru a evita o funcție cu durată mare de rulare (precum salvare/restaurare) de la o stație de lucru, fără a o încerca, operatorul de sistem poate lansa jobul într-o coadă de joburi.

Descrierea subsistemului QSYS/QBATCH sau QSYS/QBASE, care este furnizată de IBM, are o coadă de joburi QSYS/QBATCH care poate fi utilizată în acest scop. Dacă ați creat propriul dumneavoastră subsistem, ar trebui să faceți referire la coada de joburi pentru acel subsistem. Operatorul de sistem poate lansa comenzile din meniu operatorului de sistem.

Următorul este un exemplu de lansare a unei comenzi cu durată mare de rulare:

```
SBMJOB JOB(SAVELIBX) JOB(QBATCH) JOB(QSYS/QBATCH)
      CMD(SAVLIB LIBX DEV(DKT01))
```

Gestionarea joburilor prestart

Folosiți joburi prestart pentru a reduce durata de timp necesară pentru a manevra o cerere de pornire de program. Acest subiect vorbește despre cele mai obișnuite operații asociate cu joburile prestart.

Pornirea unui job prestart:

Joburile prestart pornesc, de obicei, în același moment în care este pornit subsistemul. Porniți manual un job prestart când toate joburile prestart au fost oprite de către sistem din cauza unei erori sau nu au fost niciodată pornite în timpul pornirii subsistemului din cauza STRJOBS (*NO) din intrarea de job prestart. Pentru a porni un job prestart, utilizați interfața bazată pe caractere.

Comandă: Pornire joburi prestart (STRPJ)

Comanda STRPJ nu ar trebui să fie utilizată până când nu s-a terminat pornirea subsistemului înrudit. Pentru a vă asigura că jobul prestart necesar pornește cu succes, codați o buclă de întârziere cu o reîncercare dacă comanda STRPJ eșuează.

Numărul de joburi prestart care pot fi active în același timp este limitat de atributul MAXJOBS din intrarea de job prestart și de atributul MAXJOBS pentru subsistem. Atributul MAXACT din intrarea de comunicații controlează numărul de cereri de pornire program care pot fi servite prin intrarea de comunicații în același timp.

Notă: Dacă ați specificat *NO în atributul STRJOBS, nu pornește nici un job prestart pentru intrarea jobului prestart când subsistemul pornește. Rularea comenzii STRPJ nu cauzează modificarea valorii parametrului **STRJOBS**.

Exemplu: Acest exemplu pornește joburi prestart pentru intrarea de job prestart PJPGM din subsistemul SBS1. Subsistemul SBS1 trebuie să fie activ când este lansată această comandă. Numărul de joburi pornite este numărul specificat în valoarea INLJOBS a intrării jobului prestart PJPGM. Subsistemul pornește programul PJPGM din biblioteca PJLIB.

```
STRPJ  SBS(SBS1)  PGM(PJLIB/PJPGM)
```

Punerea în coadă sau refuzarea cererilor de pornire program:

Dacă o cerere de pornire program sosește când numărul curent de joburi prestart este mai mic decât numărul specificat în atributul MAXJOBS din intrarea jobului prestart și nici unul din joburile prestart nu este disponibil pentru a trata cererea de pornire program, aveți opțiunea de a pune această nouă cerere în coadă sau de a o refuza.

Pentru a refuza sau pune în coadă cererea de pornire program, utilizați atributul WAIT din intrarea de job prestart.

WAIT(*NO) înseamnă că dacă nici un job prestart nu este disponibil imediat, cererea de pornire program este refuzată.

WAIT (*YES) înseamnă că dacă nici un job prestart nu este disponibil imediat și nici un job prestart nu poate fi pornit din cauza MAXJOBS pentru a servi cererea de pornire program, cererea de pornire program este refuzată. Dacă nu este disponibil nici un job prestart imediat, dar pot fi sau au fost pornite joburi prestart suplimentare, cererea de pornire program este pusă în buclă.

Această comandă adaugă o intrare de job prestart pentru programul PGM1 din biblioteca QGPL la descrierea de subsistem PJSBS conținută în biblioteca QGPL. Intrarea specifică faptul că 15 joburi prestart (programul PGM1 din

biblioteca QGPL) sunt pornite când este pornit subsistemul PJSBS din biblioteca QGPL. Când pool-ul de joburi prestart disponibile este redus la patru (deoarece joburile prestart servesc cererile specificate pentru programul PGM1 din biblioteca QGPL), sunt pornite zece joburi suplimentare. Dacă nu există joburi prestart disponibile pentru această intrare când se primește o cerere, cererea este refuzată.

```
ADDPJE SBS(DQGPL/PJSBS) PGM(QGPL/PGM1) INLJOBS(15)
      THRESHOLD(5) ADLJOBS(10) WAIT(*NO)
```

Oprirea unui job prestart:

Puteți utiliza Navigatorul iSeries sau interfața bazată pe caractere pentru a opri un job prestart și orice fișiere de date inline asociate pentru o intrare de job prestart într-un subsistem activ. Joburile pot să aștepte o cerere sau pot fi deja asociate cu o cerere. Fișierele de ieșire spool asociate cu joburile care sunt oprite pot fi de asemenea oprite sau lăsate să rămână în coada de ieșire. Limita numărului de mesaje scrise în istoricul fiecărui job poate fi de asemenea modificată.

Navigator iSeries:

Pentru a utiliza Navigatorul iSeries, faceți următoarele:

1. În Navigator iSeries, expandați **Control funcționare** → **Joburi active**.
2. Localizați jobul prestart pe care doriți să îl opriți.
3. Faceți clic dreapta pe jobul prestart și apoi faceți clic pe **Ștergere/Oprire**.
4. Terminați în fereastra Confirmare Ștergere/Oprire și faceți clic pe **Ștergere**.

Interfața bazată pe caractere:

Comandă : Oprire job prestart (ENDPJ)

Exemplu: Această comandă oprește toate joburile asociate cu intrarea jobului prestart PJPGM din subsistemul SBS1 imediat. Ieșirea spool produsă de aceste joburi prestart este ștearsă și istoricele de job sunt salvate.

```
ENDPJ SBS(SBS1) PGM(PJLIB/PJPGM) OPTION(*IMMED)
      SPLFILE(*YES)
```

Exemplu: Această comandă oprește toate joburile asociate cu intrarea de job prestart PJPGM2 din subsistemul SBS2. Ieșirea spool pentru aceste joburi prestart este salvată pentru procesare normală de către scriitorul de punere în spool. Joburile au 50 de secunde pentru realizarea oricărei rutine de curățare, după care sunt oprite imediat.

```
ENDPJ SBS(SBS2) PGM(PJPGM2) OPTION(*CNTRLD)
      DELAY(50) SPLFILE(NO)
```

Gestionarea obiectelor de clasă de joburi

Un obiect de clasă conține atributele de rulare care controlează mediul de runtime al unui job. Obiectele de clasă livrate de IBM, sau clasele, întrunesc atât necesitățile aplicațiilor interactive tipice cât și pe cele batch. Clasa utilizată de către un job este specificată în intrarea de rutare a descrierii de subsistem utilizată pentru a porni jobul. Dacă un job este alcătuit din mulți pași de rutare, clasa folosită de fiecare pas de rutare următor este specificată în intrarea de rutare folosită pentru a porni pasul de rutare.

Crearea unui obiect de clasă:

Puteți crea un obiect de clasă utilizând interfața bazată pe caractere. Clasa definește atributele de procesare pentru joburile care utilizează clasa. Clasa utilizată de către un job este specificată în intrarea de rutare a descrierii subsistemului utilizată pentru a porni jobul. Dacă un job conține pași de rutare multipli, clasa utilizată de fiecare pas de rutare următor este specificată în intrarea de rutare utilizată pentru a porni pasul de rutare.

Comandă: Creare clasă (CRTCLS)

Exemplu: Acest exemplu creează o clasă numită CLASS1. Clasa este memorată într-o bibliotecă curentă specificată pentru job. Textul utilizatorului 'Această clasă pentru toate joburile batch de la Dept 4836' descrie clasa. Atributele

acestei clase furnizează o prioritate de rulare de 60 și o felie de timp de 900 milisecunde. Dacă jobul nu a terminat de rulat la sfârșitul unei felii de timp, este eligibil de a fi scos din spațiul de stocare principal până când îi este alocată altă felie de timp. Ceilalți parametri se presupune că au valorile implicite.

```
CRTCLS CLS(CLASS1) RUNPTY(60) TIMESLICE(900)
TEXT('Această clasă pentru toate joburile batch de la Dept 4836')
```

Modificarea unui obiect de clasă:

Puteți modifica atributele unui obiect de clasă utilizând interfața bazată pe caractere. Orice atribut poate fi modificat cu excepția atributelor cu autoritate publică. Faceți referire la comanda RVKOBJAUT (Revoke Object Authority - Revocare autoritate obiect) și la comanda GRTOBJAUT (Grant Object Authority - Acordare autoritate unui obiect), pentru informații suplimentare despre modificarea autorizărilor obiectelor.

Comandă: Modificare clasă (CHGCLS)

Exemplu: Această comandă modifică o clasă numită CLASS1 din biblioteca din lista de biblioteci a jobului. Prioritatea de rulare pentru clasă este modificată cu 60 și felia de timp de 900 milisecunde.

```
CHGCLS CLS(CLASS1) RUNPTY(60) TIMESLICE(900)
```

Gestionarea firelor de execuție

Există multe operații pe care le puteți face când gestionați firele de execuție.

Vizualizarea firelor de execuție care rulează sub un anumit job:

Fiecare job activ care rulează pe un sistem iSeries are cel puțin un fir de execuție care rulează sub el. Un fir de execuție este o unitate independentă de lucru care rulează într-un job care folosește aceleași resurse ca și jobul. Pentru că un job depinde de lucrul făcut de un fir de execuție, este important să știe cum să găsească firele de execuție care rulează într-un anumit job.

Navigator iSeries:

Pentru a vizualiza firele de execuție care rulează sub un anumit job, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Joburi active**.
2. Faceți clic dreapta pe jobul cu care doriți să lucrați, și faceți clic pe **Detalii** → **Fire de execuție**.

Interfața bazată pe caractere:

Comanda: Gestionare job (WRKJOB)

Exemplu: Următorul exemplu afișează ecranul Lucrul cu fire de execuție pentru Crtpfrdta al jobului.

```
WRKJOB JOB(Crtpfrdta) OPTIUNE(*FIR DE EXECUȚIE)
```

Ce puteți face cu firele de execuție:

Din moment ce firele de execuție ajută joburile să proceseze mai mult de operație o dată în timp ce rulează, este posibil să fie necesară monitorizarea firelor de execuție care rulează într-un job. Aceasta vă ajută să păstrați eficiență rularea jobului. Puteți utiliza Navigatorul iSeries pentru a găsi firul de execuție pe care doriți să îl gestionați.

După ce ați localizat firul de execuție, puteți face clic dreapta pe firul de execuție și selecta una din următoarele acțiuni:

Resetare statistici

Vă permite să resetați lista de informații pe care o vizualizați, și setarea timpului trecut la 00:00:00.

Detalii Pentru că funcțiile unui fir de execuție sunt similare cu cele ale unui job, ele își partajează unele din funcțiile la fel. Detaliile conțin informații detaliate despre următoarele acțiuni fir de execuție:

- Apel stivă

- Listă de biblioteci
- Obiecte blocate
- Tranzacții
- Statistici ale performanței din trecut

Reținere

Vă permite să rețineți firul de execuție. Firele de execuție pot fi reținute de mai multe ori. Sistemul de operare urmărește numărul reținerilor unui fir de execuție.

Eliberare

Eliberează firul de execuție care a fost reținut. Firul de execuție trebuie să fie eliberat de fiecare dată când este reținut pentru a putea rula.

Ștergere/Oprire

Vă permite să opriți firul sau firele de execuție selectate.

Proprietăți fir de execuție

Afișează diversele atribute ale unui fir de execuție.

Pentru informații mai detaliate despre acțiunile pe care le puteți realiza asupra firelor de execuție vedeți ajutorul online al Navigatorului iSeries.

Vizualizarea proprietăților de fir de execuție:

Firele de execuție permit joburilor să facă mai multe lucruri în același timp. Dacă un fir de execuție își oprește procesarea, el poate opri jobul din rulare. Fereastra Proprietăți fir de execuție din Navigatorul iSeries vă permite să vedeți diverse atribute de fir de execuție și de performanță a firului de execuție care pot ajuta la înțelegerea motivului pentru care un fir de execuție nu rulează.

Navigator iSeries:

Pentru a vedea atributele unui fir de execuție, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active** sau **Joburi de server**.
2. Faceți clic dreapta pe jobul cu care doriți să lucrați și faceți clic pe **Detalii** → **Fire de execuție**.
3. Faceți clic dreapta pe firul de execuție cu care doriți să lucrați și faceți clic pe **Proprietăți**.

Informațiile de subfișa General vă permit să vedeți atributele unui fir de execuție. Aceste atribute includ identificatorul de fir de execuție, starea detaliată a unui fir de execuție, utilizatorul curent, tipul de fir de execuție care rulează, jobul sub care rulează firul de execuție și grupul de pool-uri de discuri în care rulează firul de execuție.

Informațiile de sub fișa Performanță vă permit să vizualizați elemente de performanță de bază și vă permit să modificați prioritatea de rulare a firului de execuție. **Prioritate rulare** indică importanța firului de execuție în relație cu alte fire de execuție care rulează în sistem. Valorile posibile variază între prioritatea job și 99 (ceea ce înseamnă că prioritatea cea mai mare posibilă va varia). Prioritatea de rulare a firului de execuție nu poate fi niciodată mai înaltă decât prioritatea de rulare pentru jobul în care rulează firul de execuție.

Puteți vedea valorile de performanță calculate de când a pornit firul de execuție, care includ CPU și I/O disc total. Puteți de asemenea vedea, reîmprospăta, seta o reîmprospătare automată sau puteți reseta **Statisticile de performanță trecute** care au fost calculate pentru un fir de execuție.

Interfața bazată pe caractere:

Comandă: Gestionare job (WRKJOB)

Exemplu: Următorul exemplu afișează ecranul Gestionare fire de execuție pentru jobul Crtpfrdta.

```
WRKJOB JOB(Crtpfrdta) OPTION(*THREAD)
```

Oprirea sau ștergerea firelor de execuție:

Un fir de execuție inițial, care este creat când începe un job, poate să nu fie niciodată șters sau terminat. Totuși, câteodată este necesar să termini un fir de execuție secundar astfel încât un job să poată să continue să ruleze. Fiți conștienți de firul de execuție pe care doriți să îl ștergeți deoarece jobul în care acesta rulează este posibil să nu fie capabil să se efectueze fără acest fir de execuție.

Important: Oprirea firelor de execuție nu trebuie să facă parte din rutina zilnică legată de controlul de funcționare. Oprirea unui fir de execuție este mai serioasă decât oprirea unui job deoarece lucrul din celelalte fire de execuție este posibil să se oprească sau nu. Când opriți un job, toate activitățile se opresc. Totuși, când opriți un fir de execuție, doar o parte a activităților se opresc. Celelalte fire de execuție este posibil să continue să ruleze sau nu. Dacă ele continuă să ruleze fără firul de execuție pe care l-ați oprit, este posibil să producă rezultate nedorite.

Pentru a șterge sau opri un al doilea fir de execuție, trebuie să aveți autorizare specială de service (*SERVICE) sau autorizare Control fire de execuție.

Navigatorul iSeries:

Pentru a șterge sau opri un fir de execuție, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active** sau **Joburi de server**.
2. Faceți clic dreapta pe jobul cu care doriți să lucrați și faceți clic pe **Detalii** și apoi **Fire de execuție**.
3. Faceți clic dreapta pe firul de execuție pe care doriți să îl opriți și apoi faceți clic pe **Ștergere/Oprire**.

Interfața bazată pe caractere:

Comandă: Gestionare job (WRKJOB) Opțiunea 20: **Gestionare fire de execuție, dacă sunt active**

Exemplu: Următorul exemplu afișează ecranul Gestionare fire de execuție pentru jobul Crtpfrdta.

```
WRKJOB JOB(Crtpfrdta) OPTION(*THREAD)
```

În ecranul Gestionare fire de execuție, selectați Opțiunea: 4=Oprire.

Gestionarea planificării de job

Puteți planifica un job pentru a rula utilizând Advanced Job Scheduler, utilizând fereastra Proprietăți job din Navigatorul iSeries sau modificând intrarea de planificare job prin intermediul interfeței bazată pe caractere.

Planificarea unui job batch folosind Navigator iSeries

Proprietățile jobului - fereastra Coadă de joburi furnizează o cale pentru dumneavoastră să planificați un job batch să ruleze acum, să ruleze o dată la o dată și o oră anume sau să ruleze la intervale regulate (cum ar fi în prima zi a fiecărei luni).

Pentru a planifica un job folosind Navigatorul iSeries , faceți următoarele:

1. Expandăți **serverul** → **Conexiunile mele** → **Control funcționare** → **Cozi de joburi** → **Cozi de joburi active** sau **Toate cozile de joburi** → **Coadă de job care conține jobul dumneavoastră** .
2. Faceți clic dreapta pe job și faceți clic pe **Proprietăți**.
3. În fereastra Proprietăți de job, faceți clic pe fișa Cozi de joburi.
4. Pentru a planifica jobul, folosiți opțiunile care sunt localizate sub **Când să faceți jobul disponibil să ruleze**.

Pentru informații despre cum să folosiți această fereastră, vedeți ajutorul Navigatorului iSeries.

Planificarea unui job utilizând Planificatorul din Administrare centrală

Dacă nu aveți instalat add-in-ul Advanced Job Scheduler, puteți utiliza Planificatorul din Administrare centrală pentru a planifica joburi.

Puteți invoca Planificatorul din Administrare centrală făcând clic pe butonul **Planificare** care apare în multe din ferestrele Navigatorului iSeries. De exemplu se presupune că dumneavoastră utilizați fereastra Rulare comandă a Navigatorului iSeries pentru a lansa în execuție un job de curățare dar nu doriți ca acel job să ruleze decât după ce trec orele de vârf.

1. În Navigatorul iSeries faceți clic dreapta pe serverul pe care doriți să rulați jobul de curățare și faceți clic pe **Rulare comandă**.
2. Din fereastra Rulare comandă, tastați sintaxa bazată pe caractere pentru rularea jobului dumneavoastră. Dacă aveți nevoie de ajutor tastați prima comandă și apoi faceți clic pe **Prompt**.
3. După ce ați efectuat comanda, faceți clic pe **Planificare**. Apare fereastra planificatorului din Administrare centrală unde puteți planifica acest job să ruleze o dată sau să ruleze ca un job recurent.

Planificarea joburilor cu ajutorul Advanced Job Scheduler

Următoarele informații vă vor ajuta să gestionați Advanced Job Scheduler. Mai întâi trebuie să instalați programul licențiat și apoi să citiți despre task-urile care vă permit să personalizați Advanced Job Scheduler. În sfârșit, restul task-urilor vă permit să lucrați cu acest planificator.

Ce este nou în Advanced Job Scheduler pentru V5R4:

Au fost aduse mai multe îmbunătățiri la Advanced Job Scheduler.

Adăugare comenzi multiple la un task planificat

- O listă de comandă este un set memorat de instrucțiuni pe care Advanced Job Scheduler le utilizează pentru a procesa joburi de Administrare centrală. Acum puteți adăuga o serie de comenzi la task-ul planificat de Administrare centrală și puteți controla secvența de rulare a acestor comenzi. În trecut puteați planifica doar un task de Administrare centrală (puteați planifica un job pentru a colecta inventarul, apoi alt job pentru a instala corecții și apoi altul pentru a rula comenzi). Acum puteți crea un job Advanced Job Scheduler care face toate aceste activități.

Când planificați un task, aveți opțiunea de a crea un nou job planificat, de a crea un job planificat pe baza unui job planificat existent, sau să adăugați un task la un job existent. Pot fi, de asemenea, adăugate comenzi CL la task-urile planificate de Administrare centrală. De exemplu, pentru a întârzia jobul între task-uri puteți utiliza comanda (DLYJOB)(Delay Job - Întârziere job).

Task-urile sunt procesate pe sistemul de punct final care a fost selectat la crearea task-ului. Totuși, toate comenzile CL sunt procesate pe sistemul Central. Fiecare task trebuie să se termine înainte ca noul task sau noua comandă CL din listă să proceseze.

După ce faceți clic pe butonul **Planificare** pentru un task de Administrare centrală, următoarea fereastră vă îndeamnă să specificați dacă doriți să creați un job nou, să creați un job nou pe baza altui job, sau să faceți adăugări la un job existent.

Câmpul de **Comandă** este localizat în fereastra **Proprietăți job planificat - General**. (Conexiunile mele → server → Control funcționare → Advanced Job Scheduler → Joburi planificate → Faceți clic dreapta pe un job → Proprietăți)

Controlul Banner-ului de notificare

- La distribuția fișierelor spool utilizând Distribuție raport, puteți acum alege dintr-o listă de elemente și le puteți tipări cu un font mare pe pagina de banner a noului fișier spool. Elementele implicite sunt Nume job și Nume fișier spool. Puteți selecta până la 2 elemente de banner pentru a le tipări mari.

Câmpul **Elemente de banner disponibile** este localizat în fereastra **Proprietăți ale notificării**. (Conexiunile mele → server → Control funcționare → Advanced Job Scheduler → Faceți clic dreapta pe Notificare → Proprietăți)

Adăugare în sistem a opțiunii din meniu de trimitere e-mail

- Această nouă opțiune din meniu vă permite să utilizați Advanced Job Scheduler pentru a trimite un e-mail. Când selectați acest element din meniu, apare fereastra Mesaj de e-mail nou. Aceasta este aceeași fereastră care apare când faceți clic pe **Conexiunile mele** → **server** → **Control funcționare** → **Advanced Job Scheduler** → **Notificări** → **Faceți clic dreapta pe e-mail** → **E-mail nou**.

Conexiunile mele → **Faceți clic dreapta pe un server** → **Trimitere e-mail via AJS**

Distribuire rapoarte prin containerul Operații de bază

- Fereastra **Distribuire rapoarte** vă furnizează un loc pentru a distribui manual fișierele spool care sunt generate de un job de utilizator care utilizează o listă de distribuție raport. Jobul poate fi un job pornit de Advanced Job Scheduler sau pornit manual de către un utilizator. Veți fi promptat pentru o listă de Distribuție raport. Lista de Distribuție raport este o listă de fișiere spool și destinatarilor cărora le vor fi livrate fișierele spool.

Conexiunile mele → **server** → **Operații de bază** → **Joburi** → **Faceți clic dreapta pe un job** → **Distribuire rapoarte**

Planificare disponibilitate pentru destinatarii de e-mail

- Planificarea disponibilității este acea planificare pentru care destinatarul este disponibil să primească mesaje de notificare. Puteți selecta **Întotdeauna disponibil**, **blanc** (niciodată disponibil) sau o opțiune de planificare care a fost definită anterior în fereastra de proprietăți **Advanced Job Scheduler - Planificări**.

Câmpul **Planificare disponibilitate** este localizat în fereastra **Proprietăți destinatar - Email**. (**Conexiunile mele** → **server** → **Control funcționare** → **Advanced Job Scheduler** → **Notificări** → **Destinatari** → **Faceți clic dreapta pe un nume de destinatar** → **Proprietăți**)

Manager flux de lucru

- Managerul de flux de lucru este o unealtă nouă care vă dă posibilitatea să definiți unități de lucru care pot conține o combinație de pași automatizați și manuali. Unitățile de lucru pot apoi fi planificate sau pornite manual. Cu diverse puncte de notificare, utilizatorii pot fi anunțați când pașii au pornit, s-au terminat, nu au rulat după un timp anume și au depășit limita de rulare. Fiecare pas poate avea joburi predecesoare și succesoare. Înainte ca un pas să poată fi terminat automat sau manual joburile sale predecesoare trebuie să se termine. După ce un pas s-a terminat, sunt setate să ruleze joburi succesoare lui. Se obișnuiește să se specifice joburile predecesoare care sunt aceleași ca și joburile succesoare ale pasului anterior. Aceasta face ca pasul să aștepte terminarea joburilor înainte de a face notificarea despre terminarea pasului.

Un candidat bun pentru utilizarea Managerului de flux de lucru Advanced Job Scheduler este procesarea statelor de plată. Procesarea statelor de plată conține pași manuali precum introducerea cardurilor de pontaj, validarea rapoartelor și tipărirea și plata cecurilor. Pașii automatizați pot curăța fișierele de lucru batch, pot procesa intrări de carduri de pontaj, pot rula actualizările statelor de plată și pot crea rapoarte și cecuri.

Conexiunile mele → **server** → **Control funcționare** → **Advanced Job Scheduler** → **Manager flux de lucru**

Dependența resurselor obiectelor într-un sistem de fișiere integrate

- Fereastra **Dependențe resurse** afișează informații despre dependențele resurselor unui anumit job, incluzând o listă de dependențe, cerințele necesare înainte de a continua rulare a unui job și timpul de așteptare înainte de resetarea unui job și vă permite să adăugați, înlăturați sau vizualizați proprietățile dependenței unei anumite resurse. Mai nou, în versiunea V5R4, puteți indica dacă acest obiect de dependență este un sistem de fișiere integrate și puteți să specificați calea.

Conexiunile mele → **server** → **Control funcționare** → **Advanced Job Scheduler** → **Joburi planificate** → **Faceți clic dreapta pe un job** → **Dependențe resurse** → **Creați un obiect cu un tip nou de dependență**

Selecție pagină pentru atașamente de fișiere spool de notificare

- Funcția Selecție pagină vă permite să specificați informații de selecție pe baza textului și a locației sale în fiecare pagină a unui fișier spool. Puteți specifica faptul că trebuie să existe text la o anumită locație pe fiecare pagină sau oriunde pe pagină. Puteți, de asemenea, împărți în subseturi fișierul spool selectând un interval de pagini.

Funcția Selecție pagină poate fi găsită la **Conexiunile mele** → **server** → **Control funcționare** → **Advanced Job Scheduler** → **Notificare** → **Listă distribuție rapoarte** → **Faceți clic dreapta pe o listă** → **Proprietăți** → **Faceți clic pe un fișier spool** → **Faceți clic pe Proprietăți**

Adăugarea opțiunii de a nu se reseta joburile reținute

- Momentan există un posibil impact asupra performanței atunci când sunt reținute joburile care sunt planificate să ruleze periodic. De fiecare dată când se ajunge la o dată și oră planificate pentru un job reținut, jobul de server al Advanced Job Scheduler determină dacă jobul mai este reținut și dacă da, calculează următoarea dată și oră la care ar trebui să ruleze jobul reținut. Nou în versiunea V5R4, puteți suprima acest calcul asigurându-vă că este debifat câmpul **Resetați joburi**. Când nu este bifat câmpul **Resetați joburi reținute**, când se ajunge la următoarea oră și dată pentru un job reținut, câmpurile pentru data și ora planificate sunt golite și nu mai este declanșată nici o activitate de procesare în continuare, pentru jobul reținut. Când eliberați jobul, serverul calculează următoarea dată și oră când jobul ar trebui să ruleze. Utilizarea câmpului **Resetați joburi reținute** se aplică pentru toate joburile definite utilizând Advanced Job Scheduler.

Câmpul **Resetați joburi reținute** se află în fereastra **Proprietăți Advanced Job Scheduler - General**.
(**Conexiunile mele** → **server** → **Control funcționare** → **Faceți clic dreapta pe Advanced Job Scheduler** → **Proprietăți**)

Instalarea Advanced Job Scheduler:

Prima dată când v-ați conectat la serverul dumneavoastră de Administrare centrală, Navigatorul iSeries v-a întrebat dacă doriți să instalați Advanced Job Scheduler. Dacă ați ales nu și doriți să îl instalați acum, o puteți face utilizând caracteristica Instalare plug-in-uri a Navigatorului iSeries.

1. Din fereastra dumneavoastră **Navigator iSeries**, faceți clic pe **Fișier** în bara de meniuri.
2. Faceți clic pe **Opțiuni de instalare** → **Instalare plug-in-uri**.
3. Faceți clic pe sistemul sursă unde este instalat Advanced Job Scheduler și faceți clic pe **OK**. Consultați administratorul de sistem dacă nu sunteți sigur ce sistem sursă să folosiți.
4. Introduceți-vă **ID-ul utilizator** și **Parola** pentru iSeries și faceți clic pe **OK**.
5. Faceți clic pe **Advanced Job Scheduler** din lista de selecție a plug-in-urilor.
6. Faceți clic pe **Următorul** și apoi faceți clic din nou pe **Următorul**.
7. Faceți clic pe **Terminare** pentru a termina și a ieși din setare.

Acum ați instalat Advanced Job Scheduler

Localizarea planificatorului:

Pentru localizarea planificatorului, parcurgeți pașii următori:

1. Expandați **Administrare centrală**.
2. Faceți clic pe **Scanare acum** ca răspuns la mesajul că Navigatorul iSeries a detectat o componentă nouă. Este posibil să vedeți acest mesaj din nou când accesați sisteme din containerul **Conexiunile mele**.
3. Expandați **Conexiunile mele** → de pe serverul iSeries care are instalat programul cu licență Advanced Job Scheduler → **Control funcționare** → **Advanced Job Scheduler**.

După ce ați terminat lucrul preliminar cu Advanced Job Scheduler, sunteți pregătit să setați Advanced Job Scheduler.

Setarea Advanced Job Scheduler:

După ce ați instalat Advanced Job Scheduler, trebuie să îl configurați. După ce ați terminat această muncă preliminară sunteți gata pentru a începe să planificați joburi.

Alocarea proprietăților generale:

Alocarea proprietăților generale utilizate de Advanced Job Scheduler. Puteți specifica cât timp să fie reținute intrările de activitate și istoric pentru Advanced Job Scheduler, precum și perioada cât joburile nu vor avea permisiunea de a rula.

Puteți specifica zilele lucrătoare în care joburile vor procesa și dacă este necesară o aplicație pentru fiecare job planificat. Dacă aveți instalat un produs de notificare puteți, de asemenea, seta comanda care va fi utilizată pentru trimiterea unei notificări când un job se termină sau eșuează sau puteți utiliza comanda SNDDSTJS (Send Distribution using Job Scheduler) pentru a notifica un destinatar.

Puteți specifica cât timp să fie reținute înregistrări de activitate pentru joburi, precum și perioada cât joburile nu au permisiunea de a rula. Puteți specifica zilele lucrătoare în care joburile au permisiunea de a rula și dacă este necesară o aplicație pentru fiecare job lansat în execuție.

Puteți instala un produs de notificare care vă permite să recepționați un mesaj de notificare când se termină un job. Puteți defini comanda de notificare, care va trimite o notificări când un job se termină sau eșuează sau puteți utiliza comanda SNDDSTJS pentru a notifica un destinatar.

Pentru a seta proprietățile generale pentru Advanced Job Scheduler, urmați acești pași:

1. Expandați **Control funcționare** din fereastra Navigatorului dumneavoastră iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Specificați **Reținere activitate**. Reținerea activității este intervalul de timp cât doriți să rețineți înregistrările de activitate pentru joburi. Valorile posibile sunt 1 până la 999 zile sau apariții. Faceți clic pe **Zile** pentru a specifica dacă doriți să păstrați activitatea pentru un anumit număr de zile, sau faceți clic pe **Apariții per job** dacă doriți să păstrați activitatea pentru un anumit număr de apariții per job.
4. Specificați **Reținere istoric**. Reținerea istoricului specifică, în zile, cât timp doriți să fie păstrate intrările de istoric Advanced Job Scheduler.
5. Puteți specifica o **Perioadă rezervată**. Joburile nu vor rula în această perioadă.
6. Specificați zilele lucrătoare din listă. Dacă este selectată o zi, ea este desemnată ca zi lucrătoare și se poate face referire la ea atunci când sunt planificate joburi.
7. Faceți clic pe **Aplicație necesară pentru job planificat** pentru a desemna dacă o aplicație este necesară pentru fiecare job planificat. **Aplicațiile** sunt joburi care au fost grupate împreună pentru procesare. Aceasta nu poate fi selectată dacă joburile existente nu conțin o aplicație. Dacă alegeți ca o aplicație să fie necesară pentru anumite joburi, deplasați-vă la lucrul cu aplicații.
8. Faceți clic pe **Calendar** pentru a seta planificarea, calendarele de vacanță și cel fiscal care vor fi utilizate, setați calendarul de vacanță și calendarul fiscal.
9. Faceți clic pe **Frecvența periodică se va baza pe ora de început** pentru ca următorul runtime să se bazeze pe ora de început pentru joburile care sunt planificate să ruleze periodic. De exemplu, un job este rulat la fiecare 30 de minute, începând cu ora 8:00. (Pentru ca un job să ruleze tot timpul fără întrerupere, specificați 7:59 ca oră de oprire.) Jobul rulează 20 de minute. Cu acest câmp bifat, jobul rulează la 8:00, 8:30, 9:00 și așa mai departe. Dacă acest câmp nu este bifat, jobul rulează la 8:00, 8:50, 9:40, 10:30 și așa mai departe.
10. Faceți clic pe **Resetare joburi reținute** pentru a continua recalcularea și afișarea următoarei date și ore la care rulează un job reținut.
11. Specificați o **Oră de început din zi**. Aceasta este ora din zi la care considerați începutul unei zile noi. Tuturor joburilor specificate că utilizează această oră din zi li se va modifica data cu ziua anterioară, dacă ora la care jobul pornește este înainte de câmpul **Ora de început din zi**.
12. Specificați un **Utilizator monitor de joburi**. Acest câmp specifică numele profilului utilizator de folosit ca proprietar al monitorului jobului. Toate joburile care au specificat un **Utilizator curent** folosesc profilul utilizator al monitorului jobului. Profilul utilizator implicit al monitorului jobului este QIJS.

13. Puteți specifica o comandă în câmpul **Comandă de notificare**. Utilizați notificarea din comanda SNDDSTJS (Send Distribution using Job Scheduler) livrată cu sistemul sau o comandă specificată de software-ul dumneavoastră de notificare. Comanda SNDDSTJS folosește funcția de notificare Advanced Job Scheduler. Destinatarii desemnați pot primi mesaje pentru terminări normale sau anormale ale intrărilor joburilor planificate.

Specificarea nivelurilor de permisiune:

Specificarea nivelurilor de permisiune pentru joburi, funcții ale unui produs și furnizarea de permisiuni implicite pentru un job nou.

Puteți specifica niveluri de permisiune pentru joburi, funcții ale produsului și furniza permisiuni implicite pentru un job nou care să fie asociate cu fiecare Control/Aplicație de job. Permisunile pentru un job vă permit să acordați sau retrageți accesul la următoarele acțiuni: lansare, gestionare, permisiune, afișare, copiere, actualizare sau ștergere. Puteți acorda sau refuza accesul și pentru funcții individuale ale produsului precum Gestionare calendare de planificare, Trimitere rapoarte și Adăugare job.

Nivelurile de permisiune implicite sunt transferate la joburi noi când sunt adăugate. În acest caz, sistemul va transfera permisunile noului job pe baza aplicației specificate în definiția jobului. Dacă nu este utilizată nici o aplicație, el va transfera permisunile noului job din *SYSTEM.

Specificarea nivelurilor de permisiune pentru funcțiile produsului:

Pentru specificarea nivelurilor de permisiune pentru funcțiile produsului, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Faceți clic pe **Permisuni**.
4. Selectați o funcție și faceți clic pe **Proprietăți**.
5. În fereastra Proprietăți ale permisiunilor funcțiilor, editați nivelul de permisiune după cum este necesar. Puteți acorda sau refuza accesul la public sau la utilizatori specifici.

Specificarea nivelurilor de permisiune pentru joburi:

Pentru specificarea nivelurilor de permisiune la joburi, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Joburi planificate** pentru a lista joburile.
3. Faceți clic dreapta pe jobul planificat și faceți clic pe **Permisuni**.
4. În fereastra de Proprietăți permisiuni, editați nivelul de permisiune după cum este necesar. Puteți acorda sau refuza accesul la public sau la utilizatori specifici. În plus, puteți specifica lansarea, gestionarea, permisiunea, afișarea, copierea, actualizarea sau ștergerea permisiunilor.

Specificarea nivelurilor de permisiune implicite:

Pentru specificarea nivelurilor de permisiune implicite pentru noul job asociat cu un/o Control/Aplicație de job, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Faceți clic pe **Controale/Aplicații job**.
4. Selectați un control sau o aplicație de job din listă și faceți clic pe **Permisuni job nou**.
5. În fereastra Proprietăți ale permisiunilor funcțiilor, editați nivelul de permisiune după cum este necesar. Puteți acorda sau refuza accesul la public sau la utilizatori specifici. În plus, puteți specifica lansarea, gestionarea, permisiunea, afișarea, copierea, actualizarea sau ștergerea permisiunilor.

Setarea unui calendar de planificare:

Setați un calendar cu zilele selectate pentru planificarea unui job sau a unui grup de joburi. Acest calendar poate specifica datele de utilizat pentru planificarea unui job, sau poate fi folosit în combinație cu alte planificări.

Un **calendar de planificare** este un calendar de zile selectate pe care îl puteți utiliza pentru a planifica un job sau un grup de joburi. Puteți afișa calendare de planificare, adăuga un nou calendar de planificare, adăuga un nou calendar de planificare pe baza unuia existent, sau înlătura un calendar de planificare, cu condiția ca el să nu fie în curs de utilizare de către un job planificat curent.

Puteți selecta un calendar și îi puteți afișa proprietățile pentru a face modificări. Când selectați un calendar, detaliile calendarului sunt afișate sub Detalii.

Pentru setarea unui calendar de planificare, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Pe pagina General, faceți clic pe **Calendare**.
4. În pagina Calendare de planificare, faceți clic pe **Nou**.
5. Specificați un **Nume**.
6. În câmpul **Descriere**, specificați un text care descrie calendarul.
7. Alegeți un **Calendar de referință** dacă este aplicabil. Acesta este un calendar care a fost setat anterior și proprietățile sale vor fi aplicate la noul calendar ca și cum ați combina două calendare. Nu veți avea calendare de referință dacă aceasta este prima dată când folosiți Advanced Job Scheduler.
8. Selectați datele pe care doriți să le includă calendarul dumneavoastră. Trebuie să specificați dacă fiecare dată pe care ați selectat-o este pentru anul curent sau pentru fiecare an din câmpul **Data selectată**, înainte să puteți adăuga altă dată la calendar. Altfel, orice dată pe care ați selectat-o va fi deselectată când veți face clic pe o dată diferită.
9. Specificați dacă doriți ca anumite zile ale săptămânii să fie incluse în calendar.

Setarea unui calendar de vacanță:

Setați un calendar pentru zilele în care nu doriți să permiteți procesarea unui job planificat. Pot fi specificate zile alternative pentru fiecare zi de excepție sau pentru acea zi procesarea poate fi ocolită complet.

Un **calendar de vacanță** este un calendar de excepție, pentru zile în care nu doriți să procesați un job al Advanced Job Scheduler. Pot fi specificate zile alternative pentru fiecare zi de excepție pe care o specificați într-un calendar de vacanță. Puteți afișa calendare de vacanță, adăuga un nou calendar de vacanță, adăuga un nou calendar de vacanță pe baza unuia existent sau înlătura un calendar existent, cu condiția ca el să nu fie utilizat de un job planificat curent.

Planificările predefinite pot fi folosite în calendarele de vacanță. Puteți crea o planificare THIRDFRI care are o frecvență de a treia vineri a fiecărei luni. Când folosiți THIRDFRI într-un calendar de vacanță, veți face ca toate joburile folosite de acest calendar de vacanță să nu ruleze într-a treia vineri a fiecărei luni. Într-un calendar de vacanță pot fi folosite una sau mai multe planificări. Datele care sunt generate de planificare vor fi arătate pe calendar cu margine neagră.

Puteți selecta un calendar și îi puteți afișa proprietățile pentru a face modificări. Când selectați un calendar, detaliile calendarului sunt afișate sub Detalii.

Setarea unui calendar de vacanță:

Pentru setarea unui calendar de vacanță, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și selectați **Proprietăți**.
3. Pe pagina General, faceți clic pe **Calendare**.
4. Faceți clic pe fișa **Calendare de vacanță**.
5. Faceți clic pe **Nou** și introduceți un nume pentru calendar.

6. În câmpul **Descriere**, specificați textul de descriere a calendarului.
7. Alegeți un **Calendar de referință** dacă este aplicabil. Acesta este un calendar care a fost setat anterior și proprietățile sale vor fi aplicate la noul calendar ca și cum ați combina două calendare. Nu veți avea calendare de referință dacă aceasta este prima dată când folosiți Advanced Job Scheduler.
8. Selectați datele pe care doriți să le includă calendarul dumneavoastră. Trebuie să specificați dacă fiecare dată pe care ați selectat-o este pentru anul curent sau pentru fiecare an din câmpul **Dată selectată**, înainte să puteți adăuga altă dată la calendar. Altfel, orice dată pe care ați selectat-o va fi deselectedă când veți face clic pe o dată diferită.
9. Selectați o zi alternativă pentru rularea jobului. Puteți alege ziua lucrătoare anterioară, următoarea zi lucrătoare, o dată anumită sau nimic. Pentru selectarea unei date specifice, faceți clic pe **Dată alternativă specifică** și introduceți data.
10. Selectați zilele specifice ale săptămânii pentru a fi incluse în calendar.

Adăugarea unei planificări la un calendar de vacanță:

Pentru adăugarea unui calendar de vacanță la un job planificat, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Pe pagina General, faceți clic pe **Calendare**.
4. În pagina Calendar de vacanță, selectați calendarul de vacanță și faceți clic pe **Proprietăți**.
5. În colțul din stânga-jos al fișei, faceți clic pe **Planificări**.
6. Selectați planificarea corespunzătoare și faceți clic pe **Adăugare**.
7. Pentru a modifica **Ziua alternativă**, faceți clic dreapta pe planificarea din lista **Planificări selectate** și faceți clic pe **Ziua alternativă** corectă.

Setarea unui calendar fiscal:

Setați un calendar fiscal cu zilele selectate pentru planificarea unui job sau a unui grup de joburi. Utilizați acest tip de calendar dacă doriți să împărțiți anul fiscal în perioade nu în luni.

Un *calendar fiscal* este un calendar de zile selectate pe care îl puteți utiliza pentru planificarea unui job sau a unui grup de joburi. Folosiți un calendar fiscal pentru definirea unui calendar fiscal care este unic pentru afacerile dumneavoastră. Puteți specifica datele de pornire și terminare pentru fiecare perioadă din anul fiscal.

Pentru setarea unui calendar fiscal, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului dumneavoastră iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. În fereastra General, faceți clic pe **Calendare**.
4. În pagina Calendare fiscale, faceți clic pe **Nou**.
5. Specificați un **Nume**.
6. În câmpul **Descriere**, tastați un text pentru a descrie calendarul.
7. Faceți clic pe **Nou** în fereastra Proprietăți calendar fiscal pentru a crea o nouă intrare.
8. Selectați o perioadă și specificați datele de pornire și terminare. Puteți specifica până la 13 perioade.
9. Faceți clic pe **OK** pentru salvarea intrărilor de calendar fiscal.
10. Repetați pașii 7 până la 9 de câte ori este necesar.

Specificarea unui server de poștă pentru a-l folosi la notificare:

Setați un server de poștă pentru a-l folosi la notificarea mesajelor de e-mail. Este necesar un server de poștă electronică dacă doriți să trimiteți notificări prin e-mail.

Pentru setarea proprietăților de notificare, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Expandați **Advanced Job Scheduler**.
3. Faceți clic dreapta pe **Notificare** și faceți clic pe **Proprietăți**.
4. Specificați câte zile să fie memorate mesajele. Specificați un număr în câmpul **Păstrare mesaj**.
5. Specificați un **Server de poștă ieșire (SMTP)**. De exemplu, SMTP.serveruldumneavoastră.com.
6. Specificați un **Port**. Numărul de port implicit este 25.
7. Specificați o adresă e-mail în câmpul **Adresă de răspuns**. Toate mesajele de răspuns sunt trimise la această adresă.
8. Selectați **Da** sau **Nu** în câmpul **Istoric activitate de trimitere**. Activitatea de trimitere este folosită pentru determinarea problemelor.
9. Specificați **Numărul de pagini banner** permise. Acesta este folosit în Raportul de distribuție.
10. Faceți clic pe **OK** pentru salvarea proprietăților de notificare.

Setare mai multor medii de planificare:

Puteți seta medii de planificare pe același sistem. Făcând aceasta, biblioteca de date originală poate acționa ca bibliotecă de date active și biblioteca de date copiate poate fi folosită pentru testare. Astfel aveți două medii de planificare, unul pentru testare și unul care este cel real. În plus, biblioteca de date pentru testare poate servi drept copie de rezervă dacă se produce o defecțiune a sistemului original. Această opțiune vă oferă protecție suplimentară dacă apare o eroare în biblioteca de date originală, pentru că aveți o copie de rezervă a bibliotecii de date.

Există mai multe motive de ce ați putea dori să setați mai multe medii de planificare. Este posibil să doriți să ruleze în același timp o versiune de producție și o versiune de testare a produsului. Acest tip de mediu vă permite să testați diferite planificări de joburi înainte de a le folosi cu adevărat în bibliotecii de date în sistemul de producție. Sau este posibil să aveți un sistem care este un înlocuitor al unuia sau mai multor sisteme, în care puteți utiliza un produs de oglindire a datelor care să copieze biblioteca de date Advanced Job Scheduler (QUSRIJS) din sistemul sursă într-o bibliotecă numită diferit. În acest caz, biblioteca de date este activă până când nu apare o problemă cu sistemul sursă.

Un mediu de planificare este o copie a bibliotecii QUSRIJS cu excepția datelor diferite. De exemplu puteți avea altă bibliotecă de date numită QUSRIJSTST cu toate obiectele ca și QUSRIJS. Fiecare este considerat bibliotecă de date.

Pentru setarea unui mediu de planificare multiplu, parcurgeți pașii următori:

1. Obținerea unei bibliotecii de date dintr-un sistem

Pentru a crea o bibliotecă de date, trebuie să obțineți o bibliotecă de date dintr-un sistem. Următoarele sunt trei moduri în care puteți obține o bibliotecă de date dintr-un sistem:

- Salvați biblioteca de date dintr-un sistem și restaurați-o în sistemul de producție.
- Copiați biblioteca de date în sistemul curent folosind comanda Copiere bibliotecă (CPYLIB).
- Oglindiți o bibliotecă de date pe sistemul de testare. Aceste sisteme ar trebui să ruleze același nivel de ediție versiune.

Notă: Biblioteca de date copiată, restaurată sau oglindită utilizează un nume diferit față de sistemul original.

2. Alocarea bibliotecilor de date la utilizatori

După ce obțineți o bibliotecă de date de test, adăugați biblioteca de date la proprietățile Advanced Job Scheduler și alocați utilizatori bibliotecii de date. De aceea, când un utilizator folosește Advanced Job Scheduler, modificările făcute de utilizator sunt memorate în biblioteca de date alocată utilizatorului.

3. Copiați joburi din biblioteca de date din sistemul de testare în biblioteca de date reală (opțional)

Dacă utilizați o bibliotecă de date în scopuri de testare, este posibil să doriți să copiați joburile din biblioteca de date de testare în biblioteca de date reală care este în curs de utilizare. Trebuie să faceți aceasta doar dacă ați restaurat sau copiat o bibliotecă de date în pasul 1 și aveți joburi pe care doriți să le mutați în biblioteca de date reală în utilizare. Nu trebuie să faceți aceasta dacă ați oglindit o bibliotecă de date din sistemul real într-un sistem de test.

Pentru a copia joburi din biblioteca de date a unui sistem în altul, utilizați comanda CPYJOBJS (Copy Job using Job Scheduler). Pentru informații suplimentare despre parametrii specifici pentru această comandă, vedeți ajutorul online.

Alocarea bibliotecilor de date la utilizatori:

Specificați care biblioteci de date sunt asociate cu fiecare utilizator. O bibliotecă de date va conține toate obiectele găsite în biblioteca QUSRIJS. Puteți avea oricâte biblioteci de date.

Biblioteca de date memorează orice modificare pe care utilizatorul o face utilizând Advanced Job Scheduler. O bibliotecă de date conține toate obiectele găsite în biblioteca QUSRIJS. Puteți avea un număr nelimitat de biblioteci de date.

Pentru alocarea bibliotecilor de date la utilizatori, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului dumneavoastră iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. În fereastra Biblioteci de date, faceți clic pe **Adăugare** pentru a specifica o bibliotecă de date. Bibliotecile de date care sunt menționate sunt disponibile tuturor utilizatorilor din sistem.
4. În fereastra Utilizatori, faceți clic pe **Adăugare** pentru a adăuga utilizatori noi.
5. Specificați un nume.
6. Selectați o bibliotecă de date.
7. Faceți clic pe **OK** pentru adăugarea utilizatorului.
8. Faceți clic pe **Proprietăți** pentru a modifica biblioteca de date alocată unui utilizator.

Cu bibliotecile de date puteți seta medii de planificare multiple.

Gestionarea Advanced Job Scheduler:

Următoarele informații vă vor ajuta să gestionați Advanced Job Scheduler. Întâi trebuie să planificați joburi folosind Advanced Job Scheduler. Apoi, folosiți restul operațiilor pentru a gestiona jobul.

Crearea și planificarea unui job:

Planificați un job și specificați comenzile care sunt asociate cu jobul. Puteți specifica și comenzi de pornire și oprire pentru a rula o versiune specială a jobului planificat.

Pentru a crea și planifica un nou planificator de job, urmați acești pași:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler**.
3. Faceți clic dreapta pe **Joburi planificate** și faceți clic pe **Job nou planificat**.

Crearea și planificarea unui grup de joburi:

Setați și planificați o serie de joburi care rulează consecutiv într-o ordine specificată. Joburile dintr-un grup de joburi necesită completarea înainte ca următorul job să fie lansat pentru procesare.

Grupurile de job sunt joburi care sunt grupate împreună pentru a rula consecutiv în ordinea specificată. O finalizare normală este necesară pentru fiecare job din grup înainte ca următorul job din grup să fie lansat în execuție. Dacă orice job din grup nu se finalizează normal, se oprește procesarea pentru acel grup.

Pentru crearea și planificarea unui grup de joburi nou, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic pe **Advanced Job Scheduler**.

3. Faceți clic dreapta pe **Grupuri de joburi** și faceți clic pe **Grup de joburi nou** .

Pe măsură ce completați detalii pentru noul grup de joburi, faceți referire la ajutorul online pentru informații suplimentare.

Planificări predefinite:

Creați planificări care conțin informații necesare planificării unui job sau calculați date de excepție în calendarul de sărbători.

Puteți crea planificări care conțin informații necesare planificării unui job sau calculați date de excepție într-un calendar de sărbători.

De exemplu, puteți crea o planificare ENDOFWEEK care conține ziua din săptămână de rulat, împreună cu orice alt calendar suplimentar. Planificarea ENDOFWEEK poate fi apoi folosită de toate joburile care se potrivesc cu acea frecvență de planificare. Puteți accesa această caracteristică doar prin Navigatorul iSeries.

Puteți folosi aceleași planificări predefinite care sunt folosite într-un job cu calendarele dumneavoastră de sărbători. Puteți crea o planificare THIRDFRI care are o frecvență de a treia vineri a fiecărei luni. Când folosiți THIRDFRI într-un calendar de vacanță, veți face ca toate joburile folosite de acest calendar de vacanță să nu ruleze într-a treia vineri a fiecărei luni. Puteți folosi una sau mai multe planificări într-un calendar de sărbători. Datele care sunt generate de planificare vor fi arătate pe calendar cu margine neagră.

Setați o planificare predefinită:

Pentru a seta o planificare predefinită, urmați acești pași.

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Faceți clic pe fișa Planificare.
4. Faceți clic pe **Nou** și introduceți un nume pentru planificare.
5. Tastați o descriere pentru planificare.
6. Selectați frecvența și datele pe care doriți să le includeți în planificarea dumneavoastră, la fel ca la orice calendar suplimentar.

Consultați ajutorul online pentru informații suplimentare despre completarea detaliilor pentru noua planificare.

Adăugarea unei planificări la un job planificat:

Pentru adăugarea unei planificări la un job planificat, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler**.
3. Faceți clic pe **Joburi planificate** pentru a lista joburile.
4. Faceți clic dreapta pe jobul planificat și faceți clic pe **Proprietăți**.
5. Faceți clic pe fișa de Planificare.
6. Din colțul din dreapta-sus al fișei, selectați opțiunea de planificare corespunzătoare.

Adăugarea unei planificări la un calendar de vacanță:

Un calendar de vacanță este un calendar de excepții pentru zilele în care nu doriți să procesați un job Advanced Job Scheduler. Pot fi specificate zile alternative pentru fiecare excepție pe care o specificați în calendarul de vacanță.

Pentru adăugarea unei planificări la un calendar de vacanță, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.

2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. În pagina **General**, faceți clic pe **Calendare de vacanță**.
4. În pagina de Calendare de vacanță, selectați calendarul de vacanță și faceți clic pe **Proprietăți**.
5. În colțul din stânga-jos al fișei, faceți clic pe **Planificări**.
6. Selectați planificarea corespunzătoare și faceți clic pe **Adăugare**.
7. Pentru a modifica **Ziua alternativă**, faceți clic dreapta pe planificarea din lista de **Planificări selectate** și faceți clic pe **Ziua alternativă** corectă.

Consultați ajutorul online pentru informații suplimentare.

Crearea un job planificat temporar:

Câteodată este posibil să fie necesar să se ruleze un job planificat acum sau în viitor, în plus față de planificarea sa normală. Utilizați comanda SBMJOBJS (Submit Job using Job Scheduler), opțiunea 7 din ecranul Gestionare joburi sau opțiunea **Rulare** din Navigatorul iSeries. Este posibil, de asemenea, să fie necesar să procesați numai o porțiune din comenzile din lista de comenzi, la setarea acestei rulări speciale.

Comanda SBMJOBJS vă permite să specificați ordinea comenzilor Pornire și Oprire. De exemplu, JOBA are 5 comenzi, secvențe de la 10 la 50. Puteți specifica în comanda SBMJOBJS să se pornească cu numărul de ordine 20 și să se termine cu numărul de ordine 40. Astfel se ocolesc numerele de ordine 10 și 50.

Navigatorul iSeries vă permite să selectați o comandă de pornire din lista de comenzi și o comandă de oprire.

Pentru a rula o versiune specială a unui job planificat cu Navigatorul iSeries, urmați acești pași:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler**.
3. Faceți clic pe **Joburi planificate** pentru a lista joburile.
4. Faceți clic dreapta pe jobul planificat și faceți clic pe **Rulare**.
5. Specificați dacă să ruleze jobul acum sau în viitor.
6. Selectați comenzile de început și sfârșit.

Apelați la ajutorul online pentru mai multe informații pe măsură ce completați detaliile pentru noul job.

Dependențele planificării jobului:

Setați joburi sau grupuri de joburi care sunt dependente unele de altele. Puteți selecta tipul de dependență care reflectă cum sunt procesate joburile în mediul dumneavoastră.

Advanced Job Scheduler vă permite să setați dependențe care reflectă cum sunt procesate joburile în mediu.

Dependențele determină când un job sau un grup de joburi poate rula. Puteți selecta să îndepliniți toate dependențele înainte de a putea rula un job, sau puteți îndeplini cel puțin o dependență înainte ca jobul să poată rula. Dependențele includ următoarele:

- **Dependențe job**

Dependențele jobului se referă la relațiile de predecesor și succesori dintre joburi. Joburile predecesoare sunt acelea care trebuie să ruleze înainte ca jobul succesori să ruleze. Un job succesori este un job care rulează după ce toate joburile predecesoare au fost procesate. Pot fi mai multe joburi succesori pentru un singur job predecesori precum și mai multe joburi predecesoare pentru un singur job succesori. În plus, puteți specifica să se ocolească un job dependent dacă predecesorii și succesori lui rulează într-o zi în care jobul dependent nu este planificat să ruleze.

- **Dependențe active**

Dependențele active sunt liste de joburi care nu pot fi active când jobul selectat va fi lansat. Dacă oricare din joburi este activ, Advanced Job Scheduler nu va permite jobului specificat să ruleze. Jobul selectat va fi întârziat până când toate joburile din listă sunt inactive.

- **Dependențe de resursă**

Dependențele de resursă sunt bazate pe mai multe lucruri. Fiecare tip care urmează descrie zonele care sunt verificate. Următoarele sunt tipurile de dependențe de resurse:

Fișier Jobul este dependent de existența sau non-existența unui fișier și dacă îndeplinește nivelul de alocare specificat pentru a fi procesat. Poate verifica și dacă înregistrările sunt prezente înainte ca jobul să fie procesat. De exemplu, JOBA poate fi setat astfel ca să ruleze numai când fișierul ABC există și fișierul poate fi alocat exclusiv și dacă înregistrările sunt prezente în fișier.

Obiect Jobul este dependent de existența sau inexistența unui obiect de tip QSYS și dacă el îndeplinește nivelul de alocare specificat pentru a fi procesat. De exemplu, JOBA poate fi setat astfel ca să ruleze numai când aria de date XYZ există. Jobul poate, de asemenea, să fie dependent de existența sau inexistența unui obiect aflat într-un sistem de fișiere integrat. Dacă dependența este bazată pe orice obiect din cale, încheiați calea sistemului de fișiere integrat cu un slash înainte '/'.

Configurația hardware

Jobul este dependent de existența sau non-existența unei configurații hardware și de starea sa de procesat. De exemplu, JOBA poate fi setat astfel ca să ruleze numai când dispozitivul TAP01 există și are starea Disponibil.

Fișier de rețea

Jobul este dependent de starea unui fișier de sistem pentru a fi procesat.

Subsistemul

Jobul este dependent de starea subsistemului pentru a fi procesat.

Pentru a gestiona dependențe de job, urmați acești pași:

1. Deschideți **Control funcționare** din fereastra Navigatorului dumneavoastră iSeries.
2. Expandați **Advanced Job Scheduler**.
3. Faceți clic pe **Joburi planificate**.
4. Faceți clic dreapta pe **Nume job** cu ale cărui dependențe doriți să lucrați.
5. Selectați una din următoarele: **Dependențe job**, **Dependențe active** sau **Dependențe resurse**. Consultați ajutorul online pentru informații suplimentare.

Managerul de flux de lucru:

Ca și la versiunea V5R4, Managerul fluxului de lucru vă lasă să definiți unități de lucru care conțin pași manuali și automați. Aceste unități de lucru pot fi apoi planificate sau pot rula interactiv. Managerul fluxului de lucru se află în containerul Advanced Job Scheduler din interfața Navigatorului iSeries .

Fiecare pas aflat în fluxul de lucru poate avea unul sau mai mulți predecesori joburi de Advanced Job Scheduler și unul sau mai mulți succesori joburi de Advanced Job Scheduler. Când începe un flux de lucru, primul pas este marcat cu un steguleț pentru a rula. Când el se termină, următorul pas este marcat cu steguleț pentru a rula și așa mai departe .

Următoarele reprezintă considerații suplimentare atunci când se utilizează Managerul de flux de lucru:

- Puteți porni manual un flux de lucru la orice pas. Când faceți asta, ocoliți toți pașii anteriori din fluxul de lucru.
- Pașii automați se termină după ce s-au terminat toți pașii aflați înainte. Aceasta include toate joburile Advanced Job Scheduler predecesoare.
- După ce se încheie un pas, joburile Advanced Job Scheduler succesoare sunt marcate cu steguleț pentru a rula.
- Pașii manuali se pot termina în orice ordine atât timp cât joburile predecesoare pasului s-au terminat.
- Puteți marca pașii manuali terminați ca și când nu s-ar fi terminat și să îi rulați din nou, atât timp cât nu urmează pași automați neterminați.
- Puteți face un pas să aștepte să facă notificarea că s-a terminat până la terminarea jobului specificând joburile predecesoare care sunt la fel ca și joburile succesoare ale pasului anterior.

- Puteți notifica alți utilizatori când un anumit job începe, se oprește, nu a pornit la o anumită oră, sau durează prea mult. De exemplu puteți spune unui utilizator cine este responsabil pentru un anumit pas manual pe care pașii automați anterior l-au efectuat.

Când utilizați fluxuri de lucru, istoricul activității afișează când a fost pornit un flux de lucru, pașii care au fost rulați, starea pașilor automați (succes sau eșuare), când s-a terminat un flux de lucru și starea finală a fluxului de lucru.

Tabela 7. Exemplu de Flux de lucru

Flux de lucru	PAYROLL
Planificat	Fiecare vineri la 13:00.
Notificare	Clerk - Fluxul de lucru pentru Payroll (stat de plată) a pornit
Pas 1	Automat - Specifică un job succesori pentru a inițializa fișiere de stat de plată
Pas 2	Automat: <ul style="list-style-type: none"> • Specifică jobul succesori de la pasul 1 ca job predecesori pentru acest pas • Notifică Clerk (funcționarul) că pot fi introduse carduri de pontaj
Pas 3	Manual: <ul style="list-style-type: none"> • Clerk se termină după ce sunt introduse cardurile de pontaj • Specifică un job succesori care să proceseze fișiere de carduri de pontaj și să tipărească rapoartele de carduri de pontaj • Notifică Supervisor dacă pasul nu este terminat în 120 minute
Pas 4	Automat: <ul style="list-style-type: none"> • Specifică jobul succesori de la pasul anterior ca job predecesori • Nu există joburi succesoare • Notifică Clerk să verifice rapoartele de carduri de pontaj
Pas 5	Manual: <ul style="list-style-type: none"> • Clerk (funcționarul) se va opri după ce a verificat rapoartele • Specifică un job succesori pentru a procesa state de plată
Pas 6	Automat: <ul style="list-style-type: none"> • Specifică jobul succesori de la pasul anterior ca un job predecesori • Nu există joburi succesoare • Notifică Clerk (funcționarul) și Supervisor că statul de plată s-a terminat

În acest exemplu fluxul de muncă PAYROLL (stat de plată) pornește în fiecare vineri la 13:00. Este trimisă o notificare la Clerk (funcționar) că fluxul de lucru a început.

Deoarece Pasul 1 este automat și nu are nici un job predecesori, el marchează cu steguleț jobul succesori care inițializează fișierele de stat de plată pentru a rula și apoi a se termina. Pasul 2 are jobul succesori pentru Pasul 1 ca predecesori său. Pasul 2 așteaptă ca jobul care inițializează fișierele stat de plată să se termine. După ce acesta s-a terminat, Pasul 2 notifică Clerk că poate introduce cardurile de pontaj. Nu există joburi succesori care să fie marcate cu steguleț pentru a rula.

Clerk încheie manual Pasul 3 după ce toate cardurile de pontaj au fost introduse. Jobul succesori care procesează fișierul de carduri de pontaj și tipărește un raport de carduri de pontaj, este marcat cu steguleț pentru a rula. Ca o precauție, este notificat Supervisor dacă pasul nu s-a efectuat în 120 de minute. Deoarece jobul predecesori pentru Pasul 4 este jobul succesori pentru Pasul 3, Pasul 4 așteaptă până când s-a efectuat jobul care procesează fișierul de carduri de pontaj și tipărește un raport de carduri de pontaj.

După ce jobul se termină, este notificat Clerk că raportul de carduri de pontaj poate fi verificat. Nu există joburi succesoare care să fie marcate cu steguleț pentru a rula. După ce este verificat raportul de carduri de pontaj, Clerk (funcționar) efectuează manual Pasul 5. Jobul succesor care procesează statul de plată și produce verificările este marcat cu steguleț pentru a rula.

Deoarece jobul predecesor pentru Pasul 6 este jobul succesor pentru Pasul 5, Pasul 6 așteaptă până când se termină jobul care procesează statul de plată și produce cecurile. După ce jobul se termină, el notifică Clerk și Supervisor că Payroll (stat de plată) s-a terminat. Cecurile pot fi acum tipărite și distribuite.

Pentru informații suplimentare, mai detaliate, despre Managerul de flux de lucru vedeți ajutorul online.

Creați un flux de lucru nou:

Când creați un flux de lucru nou veți specifica modul cum este pornit fluxul de lucru, timpul său maxim de procesare, pașii operațiilor și ordinea lor de rulare, detaliile de planificare, notificare și documentație.

Pentru a crea un flux de lucru nou, faceți următoarele:

- În Navigator iSeries expandați **Conexiunile mele** → **server** → **Control funcționare** → **Advanced Job Scheduler** → **Faceți clic dreapta pe Manager flux de lucru** → **Flux de lucru nou**.

Apare fereastra de Flux de lucru nou.

Pentru informații suplimentare despre cum se completează fereastra pentru Flux de lucru nou vedeți ajutorul online.

O dată ce v-ați setat fluxul de lucru puteți gestiona fluxul de lucru făcând clic dreapta pe numele fluxului de lucru și făcând clic pe **Stare flux de lucru**.

Pornirea unui flux de lucru:

Când porniți un flux de lucru, puteți alege dacă doriți ca fluxul de lucru să înceapă în ordinea inițială sau într-o ordine anume.

Pentru a porni un flux de lucru, urmați pașii:

1. Din Navigatorul iSeries, expandați **Control funcționare** → **Advanced Job Scheduler** → **Manager flux de lucru** → **Faceți clic dreapta pe un flux de lucru** → **Pornire**. Apare fereastra Pornire flux de lucru.
2. Selectați dacă doriți ca fluxul de lucru să înceapă în ordinea inițială sau într-o ordine anumită. Dacă selectați să porniți în altă ordine decât prima, toți pașii anteriori vor fi marcați ca efectuați.

Pentru informații suplimentare despre fereastra Pornire flux de lucru, vedeți ajutorul online.

Lucrați cu fluxuri de lucru:

Puteți controla și monitoriza fluxul de lucru pe măsură ce rulează utilizând fereastra Stare flux de lucru.

Puteți accesa fereastra Stare flux de lucru expandând **Conexiunile mele** → **server** → **Control funcționare** → **Advanced Job Scheduler** → **Manager flux de lucru** → **Faceți clic dreapta pe un flux de lucru** → **Stare**.

- Fereastra General vă arată starea curentă a fluxului de lucru.
- Fereastra Pași vă furnizează o listă a tuturor pașilor definiți curent în fluxul de lucru.

Puteți vedea dacă un pas a fost definit a fi automat sau manual și când a început și s-a încheiat pasul.

- Pentru a marca un pas ca fiind terminat, selectați pasul corect și bifați caseta **Terminat**.
- Pașii manuali pot fi marcați ca efectuați în orice ordine dacă s-au efectuat toate joburile Advanced Job Scheduler predecesoare.
- Pașii manuali pot fi marcați ca neterminați dacă nu există pași Automați efectuați mai departe în listă.
- Un flux de lucru poate fi pornit automat la orice pas. Aceasta ocolește toți pașii anteriori.

Pentru a reîmprospăta lista, faceți clic pe **Reîmprospătare**.

- Fereastra Documentație vă arată textul de documentație pentru fluxul de lucru.

Monitorizarea activității jobului pentru Advanced Job Scheduler:

Utilizați Advanced Job Scheduler pentru a vedea un job sau istoricul sau starea unui grup de joburi. Puteți seta și reținerea activității, adică intervalul de timp cât doriți să rețineți înregistrările de activitate pentru un job.

Activitatea jobului planificat:

Activitatea jobului planificat vă permite să specificați cât timp vor fi reținute înregistrările de activitate Advanced Job Scheduler. Valorile posibile sunt 1 până la 999 zile sau apariții. Puteți specifica să păstrați activitatea pentru un anumit număr de zile sau pentru un anumit număr de apariții per job.

Sunt afișate următoarele detalii despre un job planificat:

- Nume - Numele jobului planificat.
- Grup - Numele grupului de joburi pentru job.
- Ordine - Numărul de ordine al jobului în grup, dacă jobul se află într-un grup de joburi.
- Starea de efectuare - Starea jobului.
- Pornit - Momentul când jobul a început să ruleze.
- Oprit - Momentul când jobul s-a terminat.
- Timp scurs - Durata în ore și minute de care a avut nevoie jobul pentru a procesa.

Specificați păstrarea activității:

Pentru specificarea reținerii activității, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Expandați **Advanced Job Scheduler**.
3. Faceți clic dreapta pe **Activitate job planificat** și faceți clic pe **Proprietăți**.

Vizualizarea detaliilor activității jobului planificat:

Pentru a vizualiza detaliile activității jobului planificat, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Expandați **Advanced Job Scheduler**.
3. Faceți clic dublu pe **Activitate job planificat**.

Vizualizarea activității de job planificat a unui anumit job:

Pentru vizualizarea activității de job planificate pentru un job specific, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Expandați **Advanced Job Scheduler**.
3. Faceți clic pe **Joburi planificate**.
4. Faceți clic dreapta pe Numele jobului a cărui activitate doriți să o afișați și faceți clic pe **Activitate**.

Vizualizarea detaliilor activității jobului:

Istoricul de activitate afișează activitatea din planificator precum un job adăugat, modificat sau lansat. Sunt afișate violările de securitate, secvențele procesate de un job planificat și orice erori primite. Sunt afișate și datele și orele activităților anterioare.

Pentru a vizualiza mesajele cu informații detaliate faceți clic dreapta pe o dată și oră. Pentru a vizualiza detaliile istoricului de activitate, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Expandați **Advanced Job Scheduler**.
3. Faceți clic pe **Istoric activitate**. Sunt afișate intrările zilei curente. Pentru a modifica criteriul de selectare, selectați **Include** din meniul Opțiuni.

Vizualizarea istoricului activității pentru un anumit job:

Pentru a vizualiza istoricul de activitate pentru un job specific, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Expandați **Advanced Job Scheduler**.
3. Faceți clic pe **Joburi planificate**.
4. Faceți clic dreapta pe **Numele jobului** al cărui istoric de activitate doriți să îl afișați și faceți clic pe **Istoric activitate**.

Puteți, de asemenea, utiliza pagina **Ultima rulare**, a proprietăților jobului, pentru a vizualiza progresul jobului. Specificați comanda SETSTPJS (Set Step using Job Scheduler) înainte sau după un pas din programul CL împreună cu o descriere care prezintă starea progresului unui job. Când jobul ajunge la comanda SETSTPJS din program, descrierea asociată este afișată în pagina Ultima rulare și pe dispozitivul dumneavoastră de comunicații la distanță.

Monitorizarea mesajelor cu Advanced Job Scheduler:

Adăugați identificatori de mesaj oricărei comenzi din lista de comenzi a unui job pentru mesaje de monitorizare.

Fiecare comandă din lista de comenzi a unui job poate avea identificatori de mesaj care vor fi folosiți pentru monitorizare. Când jobul rulează și este lansat un mesaj de eroare care se potrivește cu cele introduse pentru comanda selectată, jobul înregistrează eroarea dar continuă procesarea cu următoarea comandă din listă.

Dacă sunt specificate zerouri în două sau în toate patru din pozițiile din partea dreaptă, precum pppmm00, este specificat un identificator de mesaj generic. De exemplu, dacă este specificat CPF0000, sunt monitorizate toate mesajele CPF.

Pentru adăugarea identificatorilor de mesaj la o comandă, urmați acești pași:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler**.
3. Faceți clic pe **Joburi planificate** pentru a lista joburile.
4. Faceți clic dreapta pe jobul planificat și faceți clic pe **Proprietăți**.
5. Selectați comanda de la listă și faceți clic pe **Proprietăți**.
6. Faceți clic pe **Mesaje**.
7. Introduceți identificatorii de mesaj pentru monitorizare și faceți clic pe **Adăugare**.

Crearea și gestionarea zonei de date locale:

O zonă de date locală este o porțiune de spațiu care este alocată pentru un job. Nu toate joburile își folosesc zona de date locală dar unele o fac. Fiecare comandă dintr-un job are acces la zona de date locală a jobului. Este posibil să doriți să utilizați o zonă de date locale dacă planificați un job care anterior a necesitat să specificați manual parametri suplimentari. Utilizați zona de date locală pentru a specifica parametrii suplimentari astfel încât să nu aveți nevoie să îi specificați manual de fiecare dată când pornește jobul.

Pentru specificarea informațiilor zonei de date locală pentru un job planificat, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului dumneavoastră iSeries.
2. Expandați **Advanced Job Scheduler** → **Joburi planificate**.

3. Faceți clic dreapta pe un job și faceți clic pe **Proprietăți**.
4. Editați fereastra Zonă de date locală după cum este necesar.

Consultați ajutorul online pentru informații suplimentare despre completarea detaliilor pentru zona de date locală.

Crearea și gestionarea aplicațiilor/controalelor de job:

Aplicațiile sunt joburi care sunt grupate pentru procesare. Acestea sunt mai întinse decât grupurile de joburi și nu procesează în mod necesar secvențial. Joburile din aplicații pot procesa simultan și un job nu are nevoie să îl aștepte pe altul să proceseze. Toate joburile din aplicație pot fi gestionate și pot avea propriul lor set de valori implicite de job. Controalele de job sunt valori implicite alocate unui job când îl adăugați la planificatorul de joburi precum și valori implicite folosite când jobul este lansat.

Aplicațiile sunt joburi care au fost grupate împreună pentru procesare. De exemplu, puteți avea o serie de joburi pe care le folosiți pentru statul de plată care-l doriți să-l grupați împreună pentru un proces de contabilitate.

Controale de job sunt alocate implicit la un job la fel ca și cum l-ați adăugat la un planificator de joburi la fel ca implicitele folosite când jobul este lansat. Controalele de job implicite includ lucruri cum ar fi calendarul, calendarul de vacanță, coada de joburi, descrierea de job și altele.

Puteți afișa toate aplicațiile/controalele de job existente pe sistemul dumneavoastră. Puteți adăuga o nouă aplicație/un nou control de job, adăuga o nouă aplicație/un nou control de job bazat pe una/unul existent sau înlătura o aplicație/un control de job. Puteți și selecta o aplicație/un control de job și afișa proprietățile lor pentru a face modificări.

Pentru a crea o aplicație nouă/un control de job nou, parcurgeți pașii următori:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Faceți clic pe fișa **Aplicații/Controale de job**.
4. Faceți clic pe **Nou** și introduceți un nume pentru aplicație.
5. Tastați o descriere pentru aplicație.
6. Alegeți contactele pentru aplicație. Contactele sunt nume de utilizatori care sunt contactați dacă aveți o problemă cu un job într-o aplicație. Puteți specifica până la 5 contacte per aplicație. Puteți, de asemenea, alege să adăugați sau să înlăturați contacte din lista de contacte.
7. Puteți tasta informații suplimentare care să vă ajute să identificați aplicația. Informațiile sunt asociate cu noile aplicații. Aceste informații este posibil să fie utile dacă survine vreo problemă.

Gestionarea notificărilor:

Această informație descrie cum se utilizează funcția de notificare din Advanced Job Scheduler.

În notificare, puteți realiza o serie de task-uri. Notificarea vă permite să specificați proprietățile destinatarului și să raportați proprietățile listei de distribuție. În plus, puteți trimite mesaje prin e-mail și seta o listă de escaladare în cazul în care destinatarul nu răspunde în durata de timp specificată.

Înainte de a putea să trimiteți un mesaj e-mail, trebuie să specificați un server de poștă electronică care va fi utilizat pentru notificare.

În continuare sunt evidențieri ale funcției de notificare din Advanced Job Scheduler:

Destinatar

La planificarea unui job, puteți specifica dacă să se trimită sau nu mesaje de notificare la destinarii specificați. Puteți să trimiteți un mesaj de notificare dacă un job eșuează, se termină cu succes sau nu începe în limita de timp specificată. Pentru fiecare destinatar specificat, trebuie să definiți proprietățile destinatarului. Puteți accesa proprietățile destinatarului selectând **Advanced Job Scheduler** → **Notificare** → **Destinatari** și apoi selectați un destinatar din lista de destinatari.

Listă de distribuire raport

Folosiți o listă de distribuție raport pentru a specifica o listă a fișierelor spool care sunt eligibile pentru distribuție. Fiecare fișier spool produs de un job este verificat dacă se potrivește cu lista de fișiere spool. Dacă este așa, destinatarii asociați cu acel fișier spool primesc o copie a fișierului spool via e-mail, un duplicat al fișierului spool din coada lor de ieșire, sau ambele. Puteți accesa liste de distribuire rapoarte selectând **Advanced Job Scheduler** → **Notificare** → **Listă distribuire raport**.

E-mail Puteți trimite un e-mail la orice destinatar care este definit în lista de destinatari precum și adrese de e-mail specifice. Proprietățile destinatarului trebuie să specifice o adresă de e-mail ca să trimită mesajul. Când trimiteți un e-mail, puteți atașa un fișier de spool. Fișierul de spool poate fi transmis în format PDF. În plus, puteți specifica o listă să folosească dacă destinatarul intenționat nu răspunde în perioada de timp specificat.

Specificați un fișier spool care să fie atașat la un e-mail:

Pentru a specifica un fișier de spool pe care să îl atașați la un e-mail, completați următoarele:

1. Expandați **Operații de bază** din fereastra Navigatorului iSeries.
2. Faceți clic pe **Ieșire imprimantă**.
3. Faceți clic dreapta pe fișierul spool și faceți clic pe **Trimitere via AJS**.
4. Specificați un destinatar, subiect și mesajul.

Notă: Aceasta se poate face, de asemenea, din **Cozi de ieșire**.

Listă de escaladare

O listă de escaladare specifică o listă a destinatarilor în ordine descendentă. Destinatarii sunt anunțați în ordinea în care ei sunt listați. Dacă primul receptor nu răspunde la mesaj, mesajul este trimis la următorul receptor. Acest proces continuă până când un răspuns este realizat. Pentru a defini o listă de escaladare, deplasați-vă la **Advanced Job Scheduler** → **Notificare** → **Liste de escaladare**.

Oprirea unui mesaj de la escaladare:

Pentru a opri un mesaj de la escaladare, completați următoarele:

1. Expandați **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic pe **Advanced Job Scheduler** → **Notificare** → **E-mail** → **Trimis**.
3. Faceți clic dreapta pe mesajul cu escaladare și faceți clic pe **Stop**.

Notă: Pentru a vedea doar mesajele cu escaladare, selectați **Vizualizare** → **Personalizează această vizualizare** → **Includere** din fereastra Navigatorului iSeries. Apoi, în câmpul **Tip**, selectați **Escaladare**.

Gestionarea listelor de biblioteci:

Listele de biblioteci sunt liste de biblioteci definite de utilizatori care sunt folosite de Advanced Job Scheduler când este procesat un job.

O **listă de biblioteci** este o listă definită de utilizator de biblioteci care sunt folosite de Advanced Job Scheduler pentru căutarea informațiilor necesare în timpul procesării. Puteți afișa liste de biblioteci, adăuga o nouă listă de biblioteci, adăuga o nouă listă de biblioteci pe baza uneia existente sau înlătura o listă de biblioteci cu condiția ca ea să nu fie în curs de utilizare de către un job planificat curent.

Puteți selecta o listă și îi puteți afișa proprietățile pentru a face modificări. Puteți pune până la 250 de biblioteci în lista de biblioteci.

Pentru adăugarea unei liste de biblioteci noi, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Faceți clic pe fișa **Liste de biblioteci**.

4. Faceți clic pe **Nou** și introduceți un nume pentru lista de biblioteci.
5. Tastați o descriere pentru lista de biblioteci.
6. Faceți clic pe **Răsfoire** pentru a vizualiza o listă de biblioteci existente și faceți clic pe o bibliotecă.
7. Faceți clic pe **Adăugare** pentru adăugarea listei de biblioteci selectate.

Gestionarea variabilelor de comandă:

O variabilă de comandă (cunoscută anterior ca parametru) este o variabilă pe care o puteți memora și utiliza în joburile lansate prin Advanced Job Scheduler. Exemplele de variabile de comandă includ începutul fiecărei luni, un număr de divizie, un număr de companie și altele.

Variabilele de comandă (cunoscute anterior ca și parametri) sunt variabilele pe care le memorați în Advanced Job Scheduler și le folosiți în joburile lansate prin Advanced Job Scheduler. Variabilele de comandă conțin informații care vor fi înlocuite înăuntrul șirului de comandă a unui job planificat. Exemplele de variabile de comandă includ începutul fiecărei luni, a numărului de divizie companie, a numărului companiei și altele. Puteți afișa variabile de comandă, adăuga o nouă variabilă de comandă, adăuga o nouă variabilă de comandă pe baza uneia existente sau înlătura o variabilă de comandă, cu condiția ca ea să nu fie în curs de utilizare de către un job planificat.

Puteți selecta o variabilă de comandă existentă și îi puteți afișa proprietățile pentru a face modificări.

Pentru adăugarea unei variabile de comandă nouă, parcurgeți pașii următori:

1. Deschideți **Control funcționare** din fereastra Navigatorului iSeries.
2. Faceți clic dreapta pe **Advanced Job Scheduler** și faceți clic pe **Proprietăți**.
3. Faceți clic pe fișa **Variabile de comandă**.
4. Faceți clic pe **Nou** și introduceți un nume pentru variabila de comandă.
5. Tastați o descriere pentru variabila de comandă.
6. Introduceți lungimea variabilei de comandă. Lungimea poate fi între intervalul 1 până la 90.
7. Alegeți cum doriți să furnizați valoarea de înlocuire:
 - a. Specificați datele de folosire pentru variabila de comandă. Utilizați orice caracter din acest câmp. Numărul caracterelor din date nu poate fi mai mare decât lungimea specificată în câmpul Lungime.
 - b. Introduceți o formulă de calcul a datelor. (De exemplu, consultați Ajutorul online.)
 - c. Introduceți numele de program pe care îl folosiți ca să extrageți valoarea de înlocuire.
 - d. Introduceți biblioteca pe care o folosiți ca să extrageți valoarea de înlocuire.
 - e. Alegeți dacă doriți valoarea de înlocuire extrasă de la operatorul de sistem la timpul de rulare.

Lucrați cu Advanced Job Scheduler for Wireless:

Advanced Job Scheduler for Wireless lucrează pe două tipuri de dispozitive. Un dispozitiv WML (Wireless Markup Language) este un telefon celular pentru Internet. Un dispozitiv HTML (Hypertext Markup Language) este un browser de Web pe PDA sau pe PC. În acest subiect dispozitivele sunt numite WML și HTML.

Cerințele de hardware și software:

Stabiliți dacă aveți tot software-ul și hardware-ul necesar pentru a rula Advanced Job Scheduler pentru comunicații fără fir.

Sunt necesare următoarele elemente pentru a rula Advanced Job Scheduler for Wireless:

- Programul cu licență 5722-JS1 V5R3: Produsul Advanced Job Scheduler care include Advanced Job Scheduler for Wireless.
- Un dispozitiv pentru rularea funcției
 - Un telefon cu funcții Internet, cu un serviciu Internet pentru comunicații fără fir

- Un PDA cu un browser de Web, un modem pentru comunicații fără fir și un serviciu de internet comunicații fără fir
- Un browser de Web tradițional, pe o stație de lucru
- Un server care rulează i5/OS^(R) V5R3 sau mai nouă, dintr-o rețea TCP/IP.
- Un server de aplicații Web ce rulează pe sistemul dumneavoastră central, cum ar fi oricare dintre următoarele:
 - Serverul de aplicații ASF Jakarta Tomcat
 - Orice alt server de aplicații care rulează pe sistemul central, având capacitatea de a găzdui servleturi
- Serverul HTTP instalat pe serverul iSeries
- Identificați-vă serverul HTTP cu caracteristica Advanced Job Scheduler pentru Comunicații fără fir. Pentru a face asta, conectați-vă la sistemul dumneavoastră iSeries care are instalat Advanced Job Scheduler, utilizând interfața bazată pe caractere. Apoi, specificați următoarea comandă:

CALL QIJS/QIJCINT

Selectarea unui dispozitiv:

Alegeți dispozitivele compatibile cu Advanced Job Scheduler for Wireless.

Telefoanele cu funcții de Internet și PDA-urile pentru comunicație fără fir sunt o tehnologie care se modifică cu rapiditate. Ele diferă prin dimensiunea ecranului, prin aspect și prin multe alte caracteristici semnificative. Următoarele secțiuni vă ajută să alegeți dispozitive compatibile cu Advanced Job Scheduler for Wireless. Și alte dispozitive de comunicații fără fir pot fi compatibile dacă suportă răsfoirea în Internet prin comunicații fără fir, dar interacțiunea poate fi diferită.

Telefoane cu funcții de Internet Selectați un telefon cu funcții de Internet pentru a-l utiliza cu Advanced Job Scheduler for Wireless.

PDA-uri Selectați un PDA pentru a-l utiliza cu Advanced Job Scheduler for Wireless.

PC-uri Puteți utiliza, de asemenea și un browser Web tradițional cu Advanced Job Scheduler for Wireless.

Configurați mediul de comunicații fără fir:

Modificați serverul de aplicații Web și configurația firewall-ului astfel încât Advanced Job Scheduler să ruleze corect.

Înainte de a începe să utilizați Advanced Job Scheduler for Wireless asigurați-vă că ați configurat sau setat corect următoarele elemente:

1. Configurați serverul dumneavoastră de aplicații Web. Setați Advanced Job Scheduler for Wireless să ruleze utilizând un motor de servlet ASF Jakarta Tomcat. Aceste instrucțiuni specifică cum să creați și să porniți serverul de aplicații Web. În plus, specifică un program pe care trebuie să îl rulați înainte de a lucra cu funcția de comunicații fără fir din Advanced Job Scheduler.
2. Configurați firewall-ul. Acest subiect vă descrie cum se face configurarea firewall-ului dumneavoastră pentru Navigatorul iSeries pentru comunicații fără fir. Acești pași de configurare se aplică și pentru Advanced Job Scheduler for Wireless. Vedeți acest subiect pentru a determina dacă aveți nevoie să modificați firewall-ul pentru a obține acces la sisteme de la un dispozitiv de comunicații fără fir.
3. Selectați o limbă. Limba implicită este setată pe Engleză, dar puteți configura dispozitivul să afișeze limba aleasă de dumneavoastră.

După ce ați efectuat acești pași sunteți gata pentru a vă conecta la serverul dumneavoastră și pentru a începe să utilizați Advanced Job Scheduler for Wireless.

Configurarea serverului de aplicații Web:

Înainte de a lucra cu Advanced Job Scheduler for Wireless, trebuie să porniți și să configurați serverul aplicației Web. Următoarele proceduri setează un motor de servlet ASF Tomcat pentru Serverul HTTP (motorizat de Apache), pentru a rula Advanced Job Scheduler for Wireless.

Cerințe

Înainte de a începe, trebuie să aveți autorizația QSECOFR și să aveți instalat:

- Server HTTP IBM^(R) (5722-DG1)

Notă: Următoarele instrucțiuni vor crea o nouă instanță de server HTTP; nu puteți folosi următoarele instrucțiuni pentru a seta Advanced Job Scheduler pe un server HTTP existent.

Inițializarea Advanced Job Scheduler for Wireless pe HTTP Server

Următoarea comandă va adăuga servletul Advanced Job Scheduler for Wireless la motorul de servlet Apache Software Foundation Jakarta Tomcat. Se va seta, de asemenea, un Server HTTP IBM (motorizat de Apache) numit Advanced Job SchedulerP care ascultă cererile pe portul 8210.

Înainte de a lucra cu Advanced Job Scheduler for Wireless, trebuie să inițializați Advanced Job Scheduler for Wireless pentru instanța serverului HTTP de pe sistemul dumneavoastră iSeries. Pentru a face aceasta, specificați următoarea comandă din interfața bazată pe caractere. Această comandă rulează un program care este livrat cu sistemul iSeries.

```
CALL QIJS/QIJSINT
```

După ce vă configurați serverul de aplicații Web și inițializați instanța Advanced Job Scheduler de pe serverul de aplicații Web, puteți continua să configurați mediul de comunicații fără fir al Advanced Job Scheduler.

Configurarea firewall-ului dumneavoastră:

Determinați dacă trebuie să modificați firewall-ul dumneavoastră pentru a obține acces la sisteme de la un dispozitiv aflat la distanță.

Când utilizați Navigatorul iSeries pentru comunicații fără fir, veți accesa cel puțin unul din sistemele dumneavoastră de pe Internet. Dacă accesați oricare din sistemele dumneavoastră de pe Internet astăzi, probabil aveți setat un firewall pentru a împiedica accesul neautorizat. În funcție de configurația firewall-ului dumneavoastră, este posibil să trebuiască să modificați setarea firewall-ului pentru a rula Navigatorul iSeries pentru Comunicații fără fir.

Dacă nu ați accesat niciodată sistemele dumneavoastră de pe Internet și nu aveți setat un firewall, următoarea Carte roșie IBM furnizează strategii pentru a face acest lucru în capitolele despre arhitectura de gazdă protejată și arhitectura de subrețea protejată. Vedeți Scenarii de securitate în Internet AS/400 : O Abordare practică.

Selecția o limbă:

Când vă conectați la Advanced Job Scheduler for Wireless, puteți specifica ce limbă să folosiți. Dacă nu doriți să specificați o anumită limbă, puteți continua conectarea la iSeries.

Pentru specificarea unei limbi, folosiți următorul URL:

```
gazdă.domeniu:port/servlet/AJSPervasive?lng= limba
```

- *gazdă:* Numele gazdă al sistemului care conține produsul.
- *domeniu:* Domeniul unde este localizată gazda.
- *port:* Portul la care ascultă instanța serverului Web
- *limb:* Identificatorul de 2 caractere pentru limbă. Următoarea este o listă de limbi disponibile și identificatoarele lor de 2 caractere. (ar: Arabă de: Germană en: Engleză es: Spaniolă fr: Franceză it: Italiană ja: Japoneză)

Acum puteți începe să gestionați Advanced Job Scheduler for Wireless.

Conectați-vă la iSeries:

Conectați-vă la iSeries care conține produsul Advanced Job Scheduler utilizând dispozitivul dumneavoastră de comunicații fără fir.

Pentru a începe Advanced Job Scheduler for wireless, specificați URL-ul iSeries în dispozitivul dumneavoastră de comunicații fără fir. Când indicați dispozitivul dumneavoastră către URL-ul de pe iSeries, utilizați următorul format. Asigurați-vă că sfârșitul URL-ului (/servlet/Advanced Job SchedulerPervasive) este tastat exact cum se arată:

gazdă. domeniu: port/servlet/Advanced Job SchedulerPervasive

gazdă: Numele gazdă al iSeries. *domeniu:* Domeniul unde este localizat iSeries. *port:* Portul de care ascultă instanța serverului Web. Valoarea implicită este 8210.

Pentru a specifica să se utilizeze o anumită limbă, vedeți *Selectarea unei limbi*.

Disponerea pentru telefon cu funcții Internet și browser pe PDA

Dacă v-ați conectat cu succes la caracteristica Advanced Job Scheduler for Wireless de pe iSeries, ecranul inițial conține informații sumare despre telefonul dumneavoastră cu funcții de Internet sau despre PDA. Rezumatul arată cum sunt informațiile curente, câte joburi de planificare există, câte intrări de activitate există și opțiunile de verificare a stării monitorului de job sau trimiterea unui mesaj la un destinatar. În plus, rezumatul oferă o stare generală de tip OK sau Atenție în partea de sus a ecranului. Dacă este specificat Atenție, un job are un mesaj care necesită atenție deosebită. Jobul care necesită atenție conține un semn de exclamare.


Disponerea pentru browser tradițional

Disponerea pentru browser-ul tradițional este exact la fel ca și cea din ecranul pentru telefon cu funcții Internet și PDA. Însă conținutul nu acoperă tot ecranul. De aceea, puteți reduce dimensiunea browser-ului de Web, pentru a avea mai mult spațiu pentru gestionarea altor aplicații în timp ce este ținut deschis browser-ul Advanced Job Scheduler for Wireless. În plus, dacă utilizați un browser de Internet tradițional, puteți selecta **Show all** din meniul principal Advanced Job Scheduler. Apoi, puteți vizualiza mai mult conținut într-o singură pagină Web.

După ce v-ați conectat cu succes la sistemul dumneavoastră, este posibil să doriți să vă personalizați conexiunea.

Personalizarea conexiunii:

Folosind dispozitivul fără fir, puteți personaliza interfața pentru nevoile dumneavoastră specifice. De exemplu, este posibil să doriți să vizualizați doar anumite joburi și vreți să specificați să nu vedeți numele grupului jobului. Este posibil, de asemenea, să nu doriți să accesați lista activității planificate. Pagina de personalizare de pe dispozitivul fără fir vă permite să filtrați joburile și să modificați preferințele de afișare.

Există multe căi pentru a vă personaliza conexiunea dacă un PC, PDA, sau un telefon cu funcții de Internet. Pentru a profita de aceste caracteristici, vedeți site-ul Web al produsului Advanced Job Scheduler. 

Gestionarea Advanced Job Scheduler for Wireless.:

Folosiți dispozitivul dumneavoastră de comunicații fără fir pentru a lucra cu Advanced Job Scheduler. Următoarele opțiuni sunt disponibile folosind dispozitivul pentru comunicații fără fir:

Vizualizarea joburilor active, reținute și pe cele aflate în așteptare

Puteți vizualiza o listă a joburilor obișnuite (joburi Advanced Job Scheduler) sau joburi de Administrare

centrală care au starea activ, reținut sau în așteptare. În continuare puteți personaliza joburile afișate sortând după tipul jobului, după nume sau timp. În plus, puteți specifica biblioteca de date care conține datele pentru joburi și activități.

Vizualizarea dependențele joburilor

Puteți vizualiza joburile predecesoare și succesoare ale unui anumit job. Un succesori este un job care este dependent de unul sau mai multe joburi (predecesoare) pentru a rula. În schimb, un job succesori poate fi un job predecesori pentru un alte joburi succesoare.

Afișare mesaje

Dacă un job are un mesaj care așteaptă, puteți vizualiza mesajul și să răspundeți la el folosind dispozitivul pentru comunicații fără fir.

Pornire joburi

Puteți folosi dispozitivul de comunicații fără fir pentru a lansa joburi. Opțiunile pe care puteți să le specificați când lansați un job depind de dispozitivul fără fir folosit.

Lucrați cu activitatea Advanced Job Scheduler

Puteți interacționa cu activitatea Advanced Job Scheduler de la un dispozitiv aflat la distanță. Fiecare activitate are opțiuni diferite, bazate pe starea intrării de activitate.


Internaționalizare

Advanced Job Scheduler for Wireless utilizează codurile de țară și limbă asociate cu Mașina virtuală Java^(TM) de pe iSeries^(TM) pentru a determina ce limbă și ce format de dată/oră să utilizeze pe dispozitivele dumneavoastră de comunicație fără fir. Dacă codurile implicite ale Mașinii dumneavoastră virtuale Java nu sunt cele pe care doriți să le utilizați, le puteți modifica cu ușurință. Pentru detalii suplimentare vedeți ajutorul online.

Vedeți ajutorul online pentru detalii suplimentare despre realizarea task-urilor specifice.

Depanarea Advanced Job Scheduler:

Când un job nu rulează la momentul planificat, aflați ce puteți face.

Pentru a depana Advanced Job Scheduler, mai întâi vedeți cele mai frecvente întrebări despre Planificatorul de job  Pagină Web. Citiți despre cele mai frecvente întrebări care identifică cum să faci anumite funcții cu Advanced Job Scheduler.

De asemenea, aici este o listă de elemente pe care le puteți vedea când un job nu rulează la timpul planificat:

Nivelul de corecții curent

Primul lucru pe care ar trebui să îl verificați este dacă corecțiile sunt actuale. Când cereți corecții, aveți grijă să cereți o listă cu toate corecțiile. Nu toate corecțiile sunt incluse în pachete cumulative.

Verificare monitor de joburi

- Jobul QIJSSCD ar trebui să fie activ în subsistemul QSYSWRK. Dacă nu este, procesați comanda STRJS (Start Job Scheduler-Pornire planificare job).
- Monitorizarea jobului poate fi într-o buclă dacă starea jobului este Rulează de peste zece minute. Dacă este într-o buclă, opriți jobul cu *IMMED și porniți monitorul jobului din nou (STRJS).
- Dacă este un mesaj de răspuns, răspundeți cu C (Cancel). Monitorizarea jobului va intra într-o întârziere de 90 secunde și apoi începe monitorizarea din nou. Listați înregistrarea jobului pentru monitorizarea jobului. Aceasta va conține mesajele de eroare.

Verificați istoricul Advanced Job Scheduler

Procesați comanda DSPLOGJS (Display Log for Job Scheduler) pentru job. Apăsăți F18 pentru a merge la sfârșitul listei. Intrări există pentru a explica de ce jobul nu a rulat. Exemple de intrări includ defectarea resursei, situația activă sau dependența jobului, sau eroare de supunere.

Dependența de alt job

Dacă jobul este dependent de un alt job, folosiți opțiunea 10 din ecranul Gestionare joburi pentru a afișa

dependențele jobului. Apăsăți F8 pentru a lista toate joburile anterioare. Un job dependent nu poate rula decât dacă joburile predecesoare arată *YES în coloana **Terminat** .

Urmăriți progresul unui job

Dacă un job nu funcționează cum trebuie, puteți utiliza comanda SETSTPJS (Set Step using Job Scheduler) înainte sau după un pas din programul dumneavoastră CL pentru a vă ajuta să determinați care este problema. Specificați comanda împreună cu descriere text în programul CL. Folosiți acest program de câte ori este necesar. Descrierea text care este asociată cu comanda curentă este afișată în câmpul pas Comandă din pagina Ultima rulare a proprietăților jobului planificat. În plus, puteți vizualiza câmpul pas Comandă din fereastra Stare a unui job activ. Câmpul pas Comandă este actualizat automat de fiecare dată când jobul întâlnește comanda SETSTPJS. Utilizați această comandă pentru a vă ajuta să determinați progresul unui job.

Colectarea acestor date de eșantionare vă va ajuta la analizarea problemelor:

Condiții de mesaj de eroare

Tipăriți istoricul jobului pentru sesiunea interactivă, jobul monitorizat sau planificat, în funcție de unde apare eroarea.

Data de planificare a jobului nu este corectă

Procesați comanda DSPJOBS pentru jobul cu OUTPUT(*PRINT). Tipăriți un raport calendar dacă un calendar este folosit în job. Tipăriți un raportul agendei de vacanță dacă o vacanță este folosită în job. Apăsăți tasta Print pentru a tipări afișul fiecărui calendar fiscal intrat pentru calendarul fiscal folosit în job.

Istoricul Advanced Job Scheduler

Întotdeauna tipăriți istoricul Advanced Job Scheduler pentru perioada în cauză.

Fișierele QAIJSMST și QAIJSHST

Fișierele QAIJSMST și QAIJSHST din biblioteca QUSRIJS este posibil să necesite să fie jurnalizate înainte de a încerca să se reproducă problema. De asemenea, biblioteca QUSRIJS este posibil să fie necesită de către suportul IBM.

Gestionarea intrărilor de planificare job

În plus față de fereastra Proprietăți job - Coadă de joburi a Navigatorului iSeries, mai puteți modifica intrarea de planificare job direct utilizând interfața bazată pe caractere. Următoarea este o listă de operații comune ale interfeței bazate pe caractere pe care le puteți utiliza când lucrați cu intrări de planificare job.

Important: Nu utilizați WRKJOBSCDE (Work with Job Schedule Entries - Gestionare intrări de planificare job) pentru a transforma sau șterge un job planificat care a fost planificat utilizând Planificatorul din Administrare centrală sau Advanced Job Scheduler. Dacă jobul este transformat sau șters utilizând WRKJOBSCDE, Administrarea centrală nu este notificată de modificări. Operația este posibil să nu ruleze după cum este așteptat și pot apărea mesaje de eroare în istoricele de joburi ale serverului de Administrare centrală.

Adăugarea unei intrări de planificare job:

Comanda (ADDJOBSCDE) (Add Job Schedule Entry - Adăugare intrare planificare job) vă permite să planificați joburi batch adăugând o intrare la planificarea jobului. Puteți utiliza această comandă pentru a planifica un job batch să fie lansat o dată, sau pentru a planifica un job batch să fie lansat la intervale regulate.

Comandă: Adăugare intrare planificare job (ADDJOBSCDE)

Exemplu: Această comandă lansează un job numit CLEANUP în fiecare vineri la ora 23:00. Jobul utilizează descrierea de job CLNUPJOB din biblioteca CLNUPLIB. Dacă alimentarea sistemului este oprită sau sistemul este într-o stare restricționată vineri, la ora 23:00, jobul nu mai este lansat la IPL sau când sistemul iese din starea restricționată.

```
ADDJOBSCDE JOB(CLEANUP) SCDDATE(*NONE)
            CMD(CALL PGM(CLNUPLIB/CLNUPPGM))
            SCDDAY(*FRI) SCDTIME('23:00:00')
            FRQ(*WEEKLY) RCYACN(*NOSBM)
            JOB(CLNUPLIB/CLNUPJOB)
```

Modificarea unei intrări de planificare job:

Această comandă modifică intrarea în planificarea jobului, dar nu afectează nici un job care a fost deja lansat în execuție utilizând această intrare. Pentru a modifica o intrare de job, utilizați interfața bazată pe caractere.

Pentru a modifica o intrare de planificare job, trebuie să aveți unele autorizări care sunt necesare pentru a adăuga o intrare. Totuși, autorizările asupra obiectelor individuale sunt verificate doar dacă modificați acel parametru pentru intrare. În plus, dacă nu aveți autorizarea specială *JOBCTL, puteți modifica doar intrările pe care profilul dumneavoastră utilizator le-a adăugat obiectului planificare job.

Comandă: Modificare intrare planificare job (CHGJOBSCDE)

Exemplu: Această comandă modifică intrarea BACKUP de planificare job, numărul 1001584, astfel încât joburile sale să fie puse în coada de joburi QBATCH din biblioteca QGPL.

```
CHGJOBSCDE JOB(BACKUP) ENRYNBR(001584) JOBQ(QGPL/QBATCH)
```

Exemplu: Această comandă modifică planificarea unui job batch pentru a rula programul A la 11:00 pe 15.12.03 și în fiecare săptămână în aceeași zi.

```
CHGJOBSCDE JOB(EXAMPLE) ENRYNBR(*ONLY) CMD(CALL PGM(A))  
FRQ(*WEEKLY) SCDDATE(121503) SCDTIME(110000)
```

Reținerea unei intrări de planificare job:

Comanda HLDJOBSCDE (Hold Job Schedule Entry - Reținere intrare de planificare job) vă permite să rețineți o intrare, toate intrările sau un set de intrări din planificarea jobului. Dacă o intrare este reținută, nici un job nu este lansat în execuție la momentul planificat. Pentru a reține o intrare de planificare job, utilizați interfața bazată pe caractere.

Pentru a reține intrări, trebuie să aveți autorizare specială de control job (*JOBCTL), altfel puteți reține doar acele intrări pe care le-ați adăugat. Dacă rețineți o intrare de planificare job:

- Intrarea este reținută până când este eliberată utilizând comanda RLSJOBSCDE (Release Job Schedule Entry - Eliberare intrare de planificare job) sau comanda WRKJOBSCDE (Work with Job Schedule Entries - Gestionare intrări de planificare job).
- Jobul nu este lansat în execuție atunci când este eliberat, chiar dacă ora și data la care a fost planificat să fie lansat în execuție au trecut în timp ce intrarea era reținută. Mai degrabă jobul este lansat în execuție la oricare din datele viitoare la care a fost planificat să fie lansat.

Comandă: Reținere intrare de planificare job (HLDJOBSCDE)

Exemplu: Următorul exemplu reține intrarea de planificare job CLEANUP.

```
HLDJOBSCDE JOB(CLEANUP)
```

Tipărirea unei liste de intrări de planificare job:

Pentru a tipări o listă de intrări de planificare utilizați interfața bazată pe caractere.

Comandă: Gestionare intrări de planificare job (WRKJOBSCDE)

Exemplu: Următorul exemplu tipărește o listă de intrări de planificare job.

```
WRKJOBSCDE OUTPUT(*PRINT)
```

Exemplu: Următorul exemplu tipărește informații detaliate despre fiecare intrare de planificare job.

```
WRKJOBSCDE OUTPUT(*PRINT) PRFMT(*FULL)
```

Eliberarea unei intrări de planificare job:

Comanda RLSJOBSCDE (Release Job Schedule Entry - Eliberare intrare de planificare job) vă permite să eliberați o intrare, toate intrările sau un set de intrări din planificarea jobului. Dacă eliberați o intrare de planificare job, jobul nu este lansat imediat, chiar dacă data și ora la care a fost planificat să fie lansat a trecut în timp ce intrarea era reținută. Dacă ora planificată a trecut în timp ce intrarea era reținută a fost trimis un mesaj de avertizare pentru a indica faptul că un job sau mai multe joburi au fost ratate. Atunci jobul este lansat în execuție la orice dată viitoare la care a fost planificat să fie lansat. Pentru a elibera intrări de planificare job, utilizați interfața bazată pe caractere.

Pentru a elibera intrări, trebuie să aveți autorizare specială de control job (*JOBCTL); altfel puteți lansa doar acele intrări pe care le-ați adăugat dumneavoastră.

Comandă: Eliberare intrare de planificare job (RLSJOBSCDE)

Exemplu: Acest exemplu eliberează toate intrările de planificare job care au o stare de reținere.

```
RLSJOBSCDE JOB(*ALL) ENRYNBR(*ALL)
```

Înlăturarea unei intrări de planificare job:

Comanda HLDJOBSCDE (Remove Job Schedule Entry-Înlăturare intrare de planificare job) vă permite să înlăturați o intrare, intrări sau un set de intrări din planificarea jobului. Fiecare intrare de planificare job corespunde unui job batch și conține informațiile necesare pentru a rula automat jobul o dată sau la intervale planificate regulat. Când o intrare este înlăturată cu succes vă este trimis un mesaj și coada de mesaje specificată în intrarea de planificare job. Pentru a înlătura o intrare de planificare job, utilizați interfața bazată pe caractere.

Pentru a înlătura intrări, trebuie să rulați sub un profil utilizator care are autorizare specială de control job (*JOBCTL); altfel puteți înlătura doar acele intrări pe care le-ați adăugat.

Comandă: Înlăturare intrare planificare job (RMVJOBSCDE)

Exemplu: Următorul exemplu înlătură jobul PAYROLL din planificarea de joburi.

```
RMVJOBSCDE JOB(PAYROLL) ENRYNBR(*ONLY)
```

Când jobul de sistem înlătură o intrare cu o singură supunere sau când este înlăturată o intrare de către comanda RMVJOBSCDE (Remove Job Schedule Entry-Înlăturare intrare de planificare job), mesajul de sistem CPC1239 este trimis la coada de mesaje specificată în intrare. Dacă o intrare cu o singură supunere a fost reținută când s-a ajuns la timpul său planificat și intrarea a specificat *NO pentru atributele sale de salvare, intrarea este înlăturată când este eliberată cu comanda Release Job Schedule Entry - Eliberare intrare de planificare job. În acest caz, este trimis mesajul CPC1245 către coada de mesaje specificată în intrare.

Gestionarea subsistemelor

Deoarece un job rulează în subsisteme, poate ați dori să monitorizați activitatea subsistemului pentru eventuale probleme care pot afecta abilitatea unui job de a mai funcționa.

Subsistemul este locul de lucru pentru joburi aflate pe serverul iSeries. Toți utilizatorii lucrul este terminat de joburi rulând în subsistem și este important să monitorizați această zonă pentru încetinirea performanței lucrului. În Navigator iSeries, puteți vizualiza job-uri și cozi de joburi asociate cu subsistemele. De asemenea, aveți aceeași funcționalitate, cu joburi și cozi de joburi de la oricare altă zonă care afișează joburi și cozi de joburi.

Operații de subsistem obișnuite

Aceste informații discută despre cele mai comune operații pe care le puteți realiza pe un subsistem.

Vizualizați atributele subsistemului:

Subsistemele au atribute. Aceste atribute dau informații despre starea curentă a subsistemului, sau despre valori identificate în descrierea subsistemului.

Când folosiți Navigator iSeries, următoarele atribute pot fi vizualizate pentru un subsistem activ:

- **Subsistem:** Numele subsistemului, la fel ca biblioteca care conține descrierea subsistemului.
- **Descrierea:** Descrierea subsistemului.
- **Starea:** Starea curentă a subsistemului. Ajutorul conține detalii despre stările posibile.
- **Joburi active:** Numărul de joburi active curent, fie că rulează sau că așteaptă să ruleze, în subsistem. Acest număr nu include jobul subsistem.
- **Numărul maxim de joburi active:** Numărul maxim de joburi care pot fi active, fie că rulează sau că sunt în așteptare să ruleze, în subsisteme.
- **Job de subsistem:** Numele jobului de subsistem, incluzând utilizatorul și numărul

Navigator iSeries:

Pentru a vedea atributele unui subsistem, urmați acești pași:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Subsisteme** → **Subsisteme active**.
2. Faceți clic dreapta pe subsistemul pe care doriți să îl vizualizați, apoi faceți clic pe **Proprietăți**.

Interfața bazată pe caractere:

Pentru a utiliza interfața bazată pe caractere, tastați următoarea comandă:

Comanda: Afișare descriere subsistem (DSPSBSD)

Exemplu: Această comandă afișează meniul de descriere subsistem pentru subsistemul QBATCH.

```
DSPSBSD QBATCH
```

Oprirea unui subsistem:

Puteți utiliza Navigatorul iSeries sau interfața bazată pe caractere pentru a opri unul sau mai multe subsisteme active și pentru a specifica ce se întâmplă cu lucrul activ care este procesat. Nici un job nou sau pași de rutare nu sunt pornite în subsistem după ce subsistemul este oprit.

Când un subsistem este oprit, puteți să specificați ce se întâmplă cu lucrul activ care este procesat de subsistem. De exemplu, puteți specifica pentru toate joburile din subsistem să fie terminate imediat (Imediat), sau puteți specifica faptul că joburile au permisiunea de termina procesarea înainte ca subsistemul să se oprească (Controlat).

Important: Este recomandat ca subsistemele să fie oprite utilizând opțiunea Controlat de câte ori este posibil. Aceasta permite joburilor active să se oprească singure. Folosiți această opțiune ca să asigurați terminarea joburilor înainte ca subsistemul să se oprească. Aceasta permite programelor care rulează ca să realizeze terminarea (procesul de terminare job). Specificarea valorii Imediat poate duce la rezultate nedorite, precum date care au fost parțial actualizate.

Există două tipuri de opriri.

Controlat (Recomandat)

Oprește subsistemul într-o manieră controlată. Joburile sunt, de asemenea, oprite într-o manieră controlată. Aceasta permite programelor care rulează să realizeze curățarea (procesarea opririi jobului). Când un job care este oprit are o procedură de tratare a semnalelor pentru semnalul asincron SIGTERM, semnalul SIGTERM este generat pentru acel job. Aplicația are specificată o durată pentru parametrul **DELAY** pentru a realiza curățarea înainte ca jobul să fie oprit.

Imediat

Oprește subsistemul imediat. Joburile sunt, de asemenea, oprite imediat. Când un job care este oprit are o procedură de tratare a semnalelor pentru semnalul asincron SIGTERM, semnalul SIGTERM este generat

pentru acel job și valoarea de sistem QENDJOBMT specifică o limită de timp. Înafară de tratarea semnalului SIGTERM, programele care rulează nu au permisiunea de a realiza nici o curățare.

Navigatorul iSeries:

Pentru a utiliza Navigator iSeries, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Subsisteme** → **Subsisteme active** .
2. Faceți clic dreapta pe subsistemul sau subsistemele pe care doriți să le opriți, apoi faceți clic pe **Oprire**.
3. Specificați opțiunile care să fie folosite când subsistemul este oprit.
4. Faceți clic pe **Oprire**.

Interfața bazată pe caractere:

Pentru a utiliza interfața bazată pe caractere, tastați următoarea comandă:

Comandă: Oprire subsistem (ENDSBS)

Exemplu: Această comandă oprește toate joburile active din subsistemul QBATCH și oprește subsistemul. Joburile active au voie 60 de secunde să realizeze procesarea oprire-job furnizată de aplicație.

```
ENDSBS SBS(QBATCH) OPTION(*CNTRLD) DELAY(60)
```

Utilizați parametrul Opțiune oprire subsistem (**ENDSBSOPT**) pentru a îmbunătăți performanța pentru oprirea unui subsistem. Dacă specificați ENDSBSOPT(*NOJOBLOG), subsistemul se va opri, dar nu se va produce un istoric de job pentru fiecare job care a fost în subsistem.

Dacă survine o problemă într-un job, dar ați specificat *NOJOBLOG, diagnosticarea problemei este posibil să fie dificilă sau imposibilă deoarece problema nu este înregistrată în istoricul jobului. Dacă ați utilizat atributul de job LOGOUTPUT(*PND) atunci istoricul jobului este pus într-o stare de așteptare dar nu este scris. Totuși, istoricul jobului este încă disponibil dacă este nevoie de el. vedeți subiectele înrudite despre istorice de job pentru informații suplimentare despre istorice de joburi aflate în așteptare.

Dacă specificați ENDSBSOPT(*CHGPTY *CHGTSL), prioritatea de rulare și felia de timp se vor modifica pentru toate joburile care s-au oprit, din subsistem. Joburile se vor termina mai puțin agresiv pentru cicluri de procesor și se vor opri cu un impact mai mic asupra joburilor care încă mai rulează în alte subsisteme.

Puteți specifica toate cele trei opțiuni (*NOJOBLOG, *CHGPTY, și *CHGTSL) în parametrul **ENDSBSOPT**, de exemplu:

```
ENDSBSOPT(*NOJOBLOG *CHGPTY *CHGTSL)
```

Notă: Dacă specificați *ALL pentru numele subsistemului și aveți joburi care rulează sub QSYSWRK, ar trebui să utilizați *CNTRLD pentru a împiedica un subsistem de la o terminare anormală.

Pornirea unui subsistem:

Comanda STRSBS (Start Subsystem - Pornire subsistem) pornește un subsistem utilizând descrierea de subsistem specificată în comandă. Când subsistemul este pornit, sistemul alocă resursele necesare și disponibile (spațiu de stocare, stații de lucru și cozi de joburi) care sunt specificate în descrierea subsistemului. Puteți porni un subsistem utilizând interfața Navigatorului iSeries sau interfața bazată pe caractere.

Navigator iSeries:

Pentru a porni un subsistem utilizând Navigatorul iSeries, faceți următoarele:

1. Expandăți **Conexiunile mele** → **server** → **Control funcționare**.

2. Faceți clic dreapta pe **Subsisteme** și faceți clic pe **Pornire subsistem**.
3. Indicați **Numele** și **Biblioteca** subsistemului de pornit și faceți clic pe **OK**.

Interfața bazată pe caractere:

Comandă: Pornire subsistem(STRSBS)

Exemplu: Această comandă pornește subsistemul utilizator care este asociat cu descrierea de subsistem TELLER din biblioteca QGPL. Numele subsistemului este TELLER.

STRSBS SBSD(QGPL/TELLER)

Crearea unei descrieri de subsistem

Puteți crea o descriere de subsistem în două moduri. Puteți copia o descriere subsistem existentă și o puteți modifica sau puteți crea o întreagă descriere nouă.

Următoarele reprezintă două abordări pe care le puteți utiliza:

1. Pentru a copia o descriere de subsistem existentă, utilizând interfața bazată pe caractere, faceți următoarele:
 - a. Crearea unui Obiect Duplicat (CRTDUPOBJ) al unei descrieri de subsistem existente. (Puteți utiliza, de asemenea, comenzile WRKOBJ (Work with Objects - Gestionare obiecte) sau WRKOBJPDM (Work with Objects using Programming Development Manager - Gestionare obiecte utilizând Managerul de dezvoltare a programării).
 - b. Modificați copia descrierii subsistemului astfel încât ea să funcționeze în maniera de care aveți nevoie. De exemplu, veți avea nevoie să înlăturați intrarea în coada de joburi deoarece ea identifică coada de joburi pe care o utilizează subsistemul original. Apoi va trebui să creați o nouă intrare în coada de joburi care specifică parametrii pe care îi va utiliza noul subsistem.

Țineți minte să examinați intrările de joburi cu pornire automat, intrările de stații de lucru, intrările de joburi prestart și intrările de comunicații și verificați că nu există conflicte între două subsisteme. De exemplu, verificați ca intrările de stații de lucru să nu cauzeze alocarea de către ambele subsisteme a acelor dispozitive de afișare.
2. Pentru a crea o descriere de subsistem complet nouă, utilizați interfața bazată pe caractere și faceți următoarele:
 - a. Creați o Descriere de subsistem (CRTSBSD).
 - b. Creați o Descriere de job (CRTJOB).
 - c. Creați o Clasă (CRTCLS) pentru Adăugarea intrărilor de job prestart (ADDPJE) și Adăugarea intrărilor de rutare (ADDRTGE).
 - d. Adăugarea Intrărilor de lucru la descrierea subsistemului.
 - Adăugarea unei intrări de stație de lucru (ADDWSE)
 - Adăugarea unei intrări de coadă de joburi (ADDJOBQE)
 - Adăugarea unei intrări de comunicații (ADDCMNE)
 - Adăugarea unei intrări de job autostart (ADDAJE)
 - Adăugarea unei intrări de job prestart (ADDPJE)
 - e. Adăugarea intrărilor de rutare (ADDRTGE) la descrierea subsistemului.

Adăugați intrări de job autostart:

Utilizați interfața bazată pe caractere pentru a adăuga o intrare de job autostart. Un job autostart pornește automat o dată cu subsistemul asociat. Aceste joburi în general realizează inițializarea lucrului care este asociat cu un subsistem particular. Joburile autostart pot de asemenea să realizeze lucru repetitiv sau furnizează funcții service centralizate pentru alte joburi din același subsistem.

Comandă: Adăugare intrare de job autostart (ADDAJE)

Exemplu: Acest exemplu adaugă o intrare de job autostart la descrierea subsistemului ABC.

```
ADDAJE SBSDB(USERLIB/ABC) JOB(START)
      JOBD(USERLIB/STARTJD)
```

Notă: Pentru ca modificările să se realizeze, subsistemul activ trebuie să fie oprit și apoi repornit.

Adăugare intrări de comunicații:

Fiecare intrare de comunicații descrie unul sau mai multe dispozitive de comunicație, tipuri de dispozitive sau locațiile aflate la distanță pentru care subsistemul va porni joburile când sunt primite cereri de pornire program. Subsistemul poate alocă un dispozitiv de comunicații dacă dispozitivul nu este alocat momentan pentru alt subsistem sau job. Un dispozitiv de comunicații care este momentan alocat poate fi, eventual, de-alocat, putând fi astfel făcut disponibil altor subsisteme. Pentru a adăuga o intrare de comunicații la descrierea subsistemului, utilizați interfața bazată pe caractere.

Comandă: Adăugare intrare comunicații (ADDCMNE)

Exemplu: Acest exemplu adaugă o intrare de comunicații pentru dispozitivul APPC numit COMDEV modul *ANY la descrierea subsistemului SBS1, care se află în biblioteca ALIB. Parametrul DFTUSR ia valoarea implicită *NONE, ceea ce înseamnă că nici un job nu va intra în sistem prin această intrare decât dacă sunt furnizate informații de securitate valide în cererea de pornire program.

```
ADDCMNE SBSDB(ALIB/SBS1) DEV(COMDEV)
```

Notă: Trebuie să specificați fie parametrul **DEV**, fie parametrul **RMTLOCNAME** dar nu amândouă.

Adăugare intrări în coada de joburi:

O intrare într-o coadă de joburi identifică o coadă de joburi din care joburile sunt selectate pentru a rula în subsistem. Joburile pornite dintr-o coadă de joburi sunt joburi batch. Puteți adăuga o intrare în coada de joburi utilizând interfața bazată pe caractere.

Puteți specifica următoarele elemente dintr-o intrare în coada de joburi.

- Numele cozii de joburi (**JOBQ**)
- Numărul maxim de joburi din coada de joburi care pot fi active în același timp (**MAXACT**)
- Ordinea în care subsistemul selectează cozile de joburi din care joburile pot fi pornite (**SEQNBR**)
- Numărul maxim de joburi care pot fi active în același timp pentru o prioritate de coadă de joburi specificată (**MAXPTYn**)

Comandă: Adaugă intrare în coadă de joburi (ADDJOBQE)

Exemplu: Această comandă adaugă o intrare în coada de joburi, pentru coada de joburi NIGHT (din biblioteca QGPL), la descrierea subsistemului NIGHTSBS conținută în biblioteca QGPL. Intrarea specifică faptul că până la trei joburi batch din coada de joburi NIGHT pot fi active simultan în subsistem. Este presupus numărul de ordine implicit 10.

```
ADDJOBQE SBSDB(QGPL/NIGHTSBS) JOBQ(QGPL/NIGHT) MAXACT(3)
```

Adăugare intrări job prestart:

Intrările de job prestart identifică joburile prestart care pot fi pornite când subsistemul este pornit sau când este introdusă comanda (STRPJ)(Start Prestart Jobs - Pornire joburi prestart). Puteți adăuga intrări de joburi prestart la descrierea subsistemului utilizând interfața bazată pe caractere.

Comandă: Adaugă Intrare job prestart (ADDPJE)

Exemplu: Următorul exemplu adaugă o intrare de job prestart la descrierea subsistemului ABC.

```
ADDPJE SBSDB(USERLIB/ABC) PGM(START)
      JOBD(USERLIB/STARTPJ)
```


Adăugare intrări de rutare:

Fiecare intrare de rutare specifică parametrului de intrare utilizați pentru a porni un pas de rutare pentru un job. Intrările de rutare identifică pool-ul subsistemului de memorie principală care trebuie utilizat, programul de control care trebuie rulat (de obicei programul furnizat de sistem QCMD) și informații suplimentare de runtime (memorate în obiectul de clasă). Pentru a adăuga o intrare de rutare la o descriere de subsistem, utilizați interfața bazată pe caractere.

Comandă: Aadaugă intrare de rutare (ADDRTGE)

Exemplu: Această comandă adaugă intrarea de rutare 46 la descrierea subsistemului PERT din biblioteca ORDLIB. Pentru a utiliza intrarea de rutare 46, datele de rutare trebuie să înceapă cu șirul de caractere WRKSTN2 din poziția 1. Pot fi activi oricâți pași de rutare prin această intrare la orice moment dat. Programul GRAPHIT din biblioteca ORDLIB trebuie să ruleze în pool-ul de stocare 2 utilizând clasa AZERO din biblioteca MYLIB.

```
ADDRTGE  SBSDB(ORDLIB/PERT)  SEQNBR(46)  CMPVAL(WRKSTN2)
          PGM(ORDLIB/GRAPHIT)  CLS(MYLIB/AZERO)  MAXACT(*NOMAX)
          POOLID (2)
```

Adăugare intrări de stație de lucru:

O intrare de spațiu de lucru este utilizată când un job este pornit în momentul în care un utilizator se semnează sau transferă un job interactiv dintr-un alt subsistem. Într-o intrare de stație de lucru puteți specifica următoarele elemente. Numele parametrilor sunt date în paranteze. Utilizați interfața bazată pe caractere pentru a adăuga intrări de stație de lucru.

- Numele sau tipul stației de lucru (**WRKSTN** sau **WRKSTNTYPE**)
- Numele descrierii jobului (**JOB**) sau numele descrierii jobului din profilul utilizator
- Numărul maxim de joburi care pot fi active în același timp prin intrare(**MAXACT**)
- Când stațiile de lucru urmează să fie alocate, fie când subsistemul este pornit, fie când un job interactiv intră în subsistem prin comanda (TFRJOB) (Transfer Job) și parametrul **AT**.

Pentru a adăuga o intrare de stație de lucru la o descriere de subsistem utilizați interfața bazată pe caractere.

Comandă: Aadaugă intrare de stație de lucru (ADDWSE)

Exemplu: Următorul exemplu adaugă intrarea de stați de lucru DSP12 la subsistemul ABC.

```
ADDWSE  SBSDB(USERLIB/ABC)  WRKSTN(DSP12)
          JOBDB(USERLIB/WSE)
```

Crearea unui fișier de afișare pentru semnare:

Fișierul de afișare pentru semnare este utilizat pentru a arăta ecrane de semnare la stațiile de lucru care sunt alocate subsistemului. Fișierul de afișare pentru semnare poate fi modificat când subsistemul este activ. Totuși, noul fișier de afișare pentru semnare nu e utilizat până data viitoare când este pornit subsistemul. Pentru a crea un fișier de afișare pentru semnare, utilizați interfața bazată pe caractere.

Poate fi creat un nou fișier de afișare pentru semnare utilizând ca punct de plecare fișierul de afișare pentru semnare livrat de IBM. Sursa pentru acest fișier de afișare pentru semnare se află în biblioteca QGPL în fișierul fizic sursă QDDSSRC. Se recomandă să creați un nou fișier fizic sursă și să copiați fișierul de afișare livrat de IBM în noul fișier fizic sursă, înainte de a face modificări. În acest mod, fișierul sursă original livrat de IBM este în continuare disponibil.

Considerente:

- Ordinea în care sunt declarate câmpurile din fișierul de afișare pentru semnare nu poate fi modificată. Poziția în care acestea sunt afișate pe ecran poate fi modificată.
- Nu modificați dimensiunea totală a buffer-elor de intrare sau ieșire. Pot apărea probleme grave dacă ordinea sau dimensiunea buffer-elor este modificată.
- Nu utilizați funcția ajutor cu specificațiile descrierilor de date (DDS) în fișierul de afișare pentru semnare.

- Specificați întotdeauna 256 în parametrul **MAXDEV**.
- Cuvintele cheie **MENUBAR** și **PULLDOWN** nu pot fi specificate în descrierea fișierului de afișare pentru semnare.
- Lungimea buffer-ului pentru fișierul de afișare trebuie să fie 318. Dacă este mai mică decât 318, subsistemul utilizează ecranul de semnare implicit, **QDSIGNON** din biblioteca **QSYS**.
- Nu poate fi ștearsă linia de copyright.
- Membrul **QDSIGNON** este fișierul de afișare pentru semnare livrat de IBM care utilizează o parolă de 10 caractere.
- Membrul **QDSIGNON2** este fișierul de afișare pentru semnare livrat de IBM care utilizează o parolă de 128 caractere.

Comandă: Creare fișier de afișare (CRTDSPF)

Pentru a putea gestiona câmpuri mai mici se poate modifica un câmp ascuns, din fișierul de afișare pentru semnare, numit **UBUFFER**. **UBUFFER** are o lungime de 128 octeți și este fixat ca ultimul câmp din fișierul de afișare. Acest câmp poate fi modificat pentru a funcționa ca buffer de intrare/ieșire astfel încât datele specificate în acest câmp să fie disponibile pentru programe de aplicații când este pornit jobul interactiv. Puteți face modificări asupra câmpului **UBUFFER** astfel încât el să conțină atâtea câmpuri mici câte sunt necesare, dacă sunt îndeplinite următoarele cerințe:

- Noile câmpuri trebuie să urmeze tuturor celorlalte câmpuri din fișierul de afișare. Locația câmpurilor pe ecran nu contează atât timp cât ordinea în care sunt puse în specificațiile de descriere a datelor (**DDS**) întrunește această cerință.
- Lungimea trebuie să totalizeze 128. Dacă lungimea câmpurilor este mai mare decât 128, unele date nu vor fi transmise.
- Toate câmpurile trebuie să fie câmpuri de intrare/ieșire (tastați **B** în sursa **DDS**) sau câmpuri ascunse (tastați **H** în sursa **DDS**).

Specificarea noului ecran de semnare:

Un subsistem utilizează fișierul de afișare pentru semnare care este specificat în parametrul **SGNDSPF** al descrierii subsistemului, pentru a crea ecranul de semnare la o stație de lucru a unui utilizator. Pentru a modifica fișierul de afișare pentru semnare de la cel implicit (**QDSIGNON**) la unul pe care l-ați creat dumneavoastră, utilizați interfața bazată pe caractere.

Notă: Utilizați o versiune de testare a unui subsistem pentru a verifica dacă ecranul este valid înainte de a încerca să modificați subsistemul de control.

Comandă: Modificare descriere subsistem (CHGSBSD)

Specificați noul fișier de afișare în parametrul **SGNDSPF**.

Exemplu: Următorul exemplu modifică fișierul implicit de afișare pentru semnare pentru subsistemul **QBATCH** cu un fișier nou numit **MYSIGNON**.

```
CHGSBSD SBS(DQSYS/QBATCH) SGNDSPF(MYSIGNON)
```

Modificarea descrierii unui subsistem

Comanda **CHGSBSD** (Change Subsystem Description - Modificare descriere subsistem) modifică atributele operaționale ale descrierii subsistemului specificat. Puteți modifica descrierea subsistemului în timp ce subsistemul este activ. Pentru a modifica o descriere de subsistem, utilizați interfața bazată pe caractere.

Notă: Nu puteți specifica valoarea ***RMV** în parametrul **POOLS** în timp ce subsistemul este activ, deoarece este posibil să fie suspendat un job.

Comandă: Modificare descriere subsistem (CHGSBSD)

Exemplu: Această comandă modifică definiția pool-ului de spațiu de stocare 2 ca re este utilizat de subsistemul **PAYCTL** cu o dimensiune a spațiului de stocare de 1500K și un nivel de activitate 3. Fișierul de afișare pentru semnare

este modificat cu fișierul de afișare COMPANYS și se află în biblioteca QGPL. Dacă subsistemul este activ când este lansată această comandă, COMPANYS nu este utilizat până următoarea dată când subsistemul este pornit.

```
CHGSBSD  SBSDB(QGPL/PAYCTL) POOLS((2 1500 3))
          SGNDSPF(QGPL/COMPANYS)
```

Modificare intrări de joburi autostart:

Puteți specifica o descriere de job diferită pentru o intrare de job autostart definită anterior. Pentru a modifica o intrare de job autostart, utilizați interfața bazată pe caractere

Comanda: Modificare intrare de job autostart (CHGAJE)

Exemplu: Următorul exemplu modifică descrierea de job utilizată de intrarea jobului autostart numită START, din subsistemul ABC, din biblioteca USERLIB.

```
CHGAJE  SBSDB(USERLIB/ABC)  JOB(START)
          JOBD(USERLIB/NEWJD)
```

Notă: Pentru ca aceste modificări să intre în vigoare, subsistemul activ trebuie să fie oprit și apoi repornit.

Modificarea intrărilor de comunicație:

Puteți modifica atributele unei intrări de comunicații existente dintr-o descriere de subsistem existentă utilizând interfața bazată pe caractere.

- Când sunt modificate parametrii (**JOBD**)(Descriere job) sau (**DFTUSR**) (Profil utilizator implicit) se modifică, de asemenea, intrarea de comunicații; totuși, valorile acestor parametri nu sunt modificate pentru nici unul din joburile active în acest moment.
- Dacă valoarea parametrului (**MAXACT**) (Maxim de joburi active) este redusă la un număr mai mic decât numărul total de joburi care sunt active prin intrarea de comunicații, nu mai este procesată nici o cerere de pornire program nouă. Joburile active continuă să ruleze; dar nici o cerere suplimentară de pornire program nu este procesată până când numărul de joburi active nu devine mai mic decât valoarea specificată pentru parametrul **MAXACT**.

Comandă: Modificare intrare comunicații (CHGCMNE)

Exemplu: Acest exemplu modifică intrarea de comunicații (din descrierea subsistemului QGPL/BAKER) pentru dispozitivul A12 și modul *ANY. Nivelul maxim de activitate este modificat cu *NOMAX ceea ce înseamnă că intrarea de comunicații nu pune nici o restricție asupra numărului de cereri de pornire program care pot fi active în același timp. Totuși, valoarea MAXJOBS din descrierea subsistemului BAKER limitează numărul total de joburi care pot fi active în subsistem. Aceasta include joburile create de cererile de pornire program. Există, de asemenea, o limită pe care utilizatorul o poate specifica asupra numărului de joburi active care pot fi rulate prin orice intrare de rutare particulară (MAXACT). Limita specificată în intrarea de rutare poate controla numărul de joburi utilizând un anumit pool sau nivelul de recursivitate al unui anumit program. În orice caz, nici una din aceste limite nu poate fi depășită ca rezultat al procesării unei cereri de pornire program.

```
CHGCMNE  SBSDB(QGPL/BAKER)  DEV(A12)  MAXACT(*NOMAX)
```

Modificarea intrărilor în coada de joburi:

Puteți modifica o intrare într-o coadă de joburi existentă din descrierea subsistemului specificat. Această comandă poate fi lansată în timp ce un subsistem este activ sau inactiv. Pentru a modifica intrarea în coada de joburi dintr-un subsistem, utilizați interfața bazată pe caractere.

Comandă: Modificare intrare în coadă de joburi (CHGJOBQE)

Exemplu: Această comandă modifică numărul maxim de joburi care pot fi active simultan din coada de joburi QBATCH din biblioteca QGPL. Numărul de ordine al intrării în coada de joburi nu se modifică. Până la patru joburi

din coada de joburi QBATCH pot fi active în același timp. Cel mult un job poate fi activ de la nivelul de prioritate 1. Nu există un număr maxim de joburi care pot fi active simultan de la nivelul de prioritate 2. Nivelurile de prioritate de la 3 la 9 nu se modifică.

```
CHGJOBQE  SBSB(QGPL/QBATCH)  JOBQ(QGPL/QBATCH)  MAXACT(4)
          MAXPTY1(1)  MAXPTY2(*NOMAX)
```

Modificarea intrărilor prestart:

Puteți modifica o intrare de job prestart din descrierea subsistemului specificat. Subsistemul este posibil să fie activ când este modificată o intrare de job prestart. Modificările făcute asupra intrării când subsistemul este activ sunt reflectate în timp. Orice job prestart nou pornit după ce este lansată comanda utilizează noile valori referitoare la job. Această comandă identifică joburile prestart care sunt pornite când este pornit subsistemul sau când este lansată comanda STRPJ (Start Prestart Jobs - Pornire joburi prestart).

Pentru a modifica intrarea de job prestart a unei descrieri de subsistem, utilizați interfața bazată pe caractere.

Comandă: Modificare intrare job prestart (CHGPJE)

Exemplu: Acest exemplu modifică intrarea de job prestart pentru programul PGM1 din biblioteca QGPL din descrierea subsistemului PJSBS conținută în biblioteca QGPL. Joburile prestart asociate cu această intrare nu sunt pornite următoarea dată când este pornită descrierea de subsistem PJSBS din biblioteca QGPL. Comanda STRPJ este necesară pentru pornirea joburilor prestart. Când mai multe joburi trebuie să fie pornite, este pornit un job suplimentar.

```
CHGPJE  SBSB(QGPL/PJSBS)  PGM(QGPL/PGM1)  STRJOBS(*NO)
        THRESHOLD(1)  ADLJOBS(1)
```

Modificarea intrărilor de rutare:

Puteți modifica o intrare de rutare din descrierea subsistemului specificat, utilizând interfața bazată pe utilizator. Intrarea de rutare specifică parametrii utilizați pentru a porni un pas de rutare pentru un job. Subsistemul asociat poate fi activ când se fac modificările.

Comandă: Modificare intrare de rutare (CHGRTGE)

Exemplu: Această comandă modifică intrarea de rutare 1478 din descrierea subsistemului ORDER aflată în biblioteca LIB5. Este utilizat același program, dar acum el rulează în pool-ul de spațiu de stocare 3 utilizând clasa SOFAST din biblioteca LIB6.

```
CHGRTGE  SBSB(LIB5/ORDER)  SEQNBR(1478)  CLS(LIB6/SOFAST)  POOLID(3)
```

Modificarea intrărilor de stație de lucru:

Puteți specifica o descriere de job diferită pentru o intrare de stație de lucru definită anterior utilizând interfața bazată pe caractere.

- Când este specificat parametrul Descriere job (**JOB**D), intrarea stației de lucru va fi modificată; totuși, valoarea acestui parametru nu este modificată pentru nici unul din joburile pornite prin această intrare, care sunt active în acel moment.
- Dacă valoarea parametrului Maxim de joburi active (**MAXACT**) este redusă la un număr mai mic decât numărul total de stații de lucru care sunt active prin intrarea stației de lucru, nu se va mai permite semnarea nici unei stații de lucru suplimentare. Stațiile de lucru active nu vor fi închise. Pot fi create joburi suplimentare pentru o stație de lucru activă, prin comanda TFRSECJOB (Transfer Secondary Job - Transfer job secundar) sau prin comanda TFRGRPJOB (Transfer to Group Job - Transfer la job de grup). Nu se va mai permite semnarea altor stații de lucru până ce numărul de stații de lucru active nu va fi mai mic decât valoarea specificată pentru parametrul **MAXACT**.

Comandă: Modificare intrare stație de lucru (CHGWSE)

Exemplu: Această comandă modifică intrarea de stație de lucru pentru stația de lucru A12 din subsistemul BAKER aflat în biblioteca cu scop general. Este creat un job pentru stația de lucru A12 atunci când este introdusă parola utilizatorului în promptul de semnare și este apăsată tasta Enter.

```
CHGWSE  SBSDB(QGPL/BAKER) WRKSTN(A12) AT(*SIGNON)
```

Modificarea ecranului de semnare:

Sistemul dumneavoastră este livrat cu fișierul implicit de afișare pentru semnare QDSIGNON, care se află în biblioteca QSYS. În situații în care aveți un mediu multilingv, este posibil să doriți să modificați ceea ce este afișat pe ecranul de semnare. Sau, poate doriți să adăugați informații despre compania dumneavoastră la ecranul de semnare. În astfel de situații trebuie mai întâi să creați un nou fișier de afișare. Pentru a face asta, utilizați interfața bazată pe caractere.

Atributul SGNDSPF din descrierea subsistemului trimite către fișierul de afișare pentru semnare pe care utilizatorul îl vede la semnarea pe subsistem.

Să recapitulăm pașii utilizați pentru modificarea ecranului de semnare:

1. Crearea unui nou fișier de afișare pentru semnare.
2. Modificarea descrierii subsistemului pentru a utiliza fișierul de afișare modificat în locul celui implicit al sistemului.
3. Testarea modificării.

Ștergerea unei descrieri de subsistem

Comanda DLTSBSD (Delete Subsystem Description - Ștergere descriere subsistem) șterge descrierile de subsistem specificate (inclusiv toate intrările de lucru sau intrările de rutare adăugate la ele) din sistem. Cozile de joburi alocate acestui subsistem de către comanda ADDJOBQE (Add Job Queue Entry - Adăugare intrare în coadă de joburi) nu sunt șterse. De fapt, când ștergeți o descriere de subsistem (SBSD), nu este șters nici unul din obiectele la care se face referire de către SBSDB.

Subsistemul asociat trebuie să fie inactiv pentru a putea fi șters. Utilizați interfața bazată pe caractere pentru a șterge un subsistem.

Comandă: Ștergere descriere subsistem (DLTSBSD)

Această comandă șterge descrierea de subsistem inactivă numită BAKER din biblioteca LIB1.

```
DLTSBSD  SBSDB(LIB1/BAKER)
```

Înlăturarea intrărilor de job autostart:

Puteți înlătura o intrare de job autostart din descrierea unui subsistem utilizând interfața bazată pe caractere.

Comandă: Înlăturarea intrării de job autostart (RMVAJE)

Exemplu: Următorul exemplu înlătură intrarea de job autostart pentru jobul START din descrierea de subsistem ABC.

```
RMVAJE  SBSDB(USERLIB/ABC)  JOB(START)
```

Notă: Pentru ca modificările să intre în vigoare subsistemul activ trebuie să fie oprit și apoi repornit.

Înlăturarea intrărilor de comunicație:

Puteți înlătura intrările de comunicație din descrierea subsistemului utilizând interfața bazată pe caractere. Toate joburile care sunt active prin intrarea de comunicații care este înlăturată trebuie să fie oprite înainte ca această comandă să poată fi rulată.

Comandă: Înlăturare intrare de comunicații (RMVCMNE)

Exemplu: Această comandă înlătură intrarea dispozitivului de comunicații pentru dispozitivul COMDEV din descrierea subsistemului SBS1 din biblioteca LIB2.

```
RMVCMNE  SBS1(LIB2/SBS1)  DEV(COMDEV)
```

Înlăturarea intrărilor de coadă de joburi:

Puteți înlătura intrări de coadă de joburi dintr-o descriere de subsistem utilizând interfața bazată pe caractere. Joburile din coada de joburi rămân în coadă când este înlăturată intrarea de coadă de joburi este înlăturată din descrierea subsistemului. O coadă de joburi nu poate fi înlăturată dacă a fost pornit vreun job activ momentan din coada de joburi.

Comandă: Înlăturare intrare de coadă de joburi (RMVJOBQE)

Exemplu: Această comandă înlătură intrarea în coada de joburi care se referă la coada de joburi BATCH2 din MYLIB din descrierea de subsistem NIGHTRUN memorată în biblioteca MYLIB.

```
RMVJOBQE  SBS1(MYLIB/NIGHTRUN)  JOBQ(MYLIB/BATCH2)
```

Înlăturarea intrărilor prestart:

Puteți înlătura intrări de job prestart din descrierea de subsistem utilizând interfața bazată pe caractere. O intrare de job prestart nu poate fi înlăturată dacă vreun job activ momentan a fost pornit utilizând acea intrare.

La înlăturarea unei intrări unde este specificat *LIBL pentru numele bibliotecii, se caută în lista de biblioteci un program cu numele specificat. Dacă este găsit un program în lista de biblioteci dar există o intrare cu un nume de bibliotecă diferit (care este găsită mai târziu în lista de biblioteci), nu este înlăturată nici o intrare. Dacă nu este găsit un program în lista de biblioteci dar există o intrare, nu este înlăturată nici o intrare.

Comandă: Înlăturare intrare de job prestart (RMVPJE)

Exemplu: Această comandă înlătură intrarea de job prestart pentru programul PGM1 (din biblioteca QGPL) din descrierea de subsistem PJE conținută în biblioteca QGPL.

```
RMVPJE  SBS1(QGPL/PJE)  PGM(QGPL/PGM1)
```

Înlăturarea intrărilor de rutare:

Puteți înlătura o intrare de rutare din descrierea de subsistem specificată utilizând interfața bazată pe caractere. Subsistemul poate fi activ la momentul când este rulată comanda. Totuși, intrarea de rutare nu poate fi înlăturată dacă există joburi active momentan care au fost pornite utilizând intrarea.

Comandă: Înlăturare intrare de rutare (RMVRTGE)

Exemplu: Această comandă înlătură intrarea de rutare 9912 din descrierea subsistemului PERT din biblioteca OR.

```
RMVRTGE  SBS1(OR/PERT)  SEQNBR(9912)
```

Înlăturarea intrărilor de stație de lucru:

Puteți înlătura o intrare de stație de lucru dintr-o descriere de subsistem utilizând interfața bazată pe caractere. Subsistemul poate fi activ la momentul când este rulată comanda. Totuși, toate joburile care sunt active prin intrarea de stație de lucru trebuie să fie oprite înainte ca aceasta să poată fi înlăturată.

Comandă: Înlăturare intrare de stație de lucru (RMVWSE)

Exemplu: Acest exemplu înlătură intrarea de stație de lucru B53 din descrierea subsistemului numită CHARLES din biblioteca LIB2.

```
RMVWSE  SBS1(LIB2/CHARLES)  WRKSTN(B53)
```

Configurarea unui subsistem interactiv

Informația din această secțiune explică cum să setați un nou subsistem interactiv.

Acești pași sunt descriși ca și când comenzile sunt introduse manual. Totuși, ar trebui să folosiți un program CL pentru a crea subsistemele dumneavoastră astfel încât să puteți recrea mai ușor configurația dumneavoastră pentru scopuri de recuperare.

Când setați un nou subsistem interactiv ar trebui să considerați câte dispozitive vor fi alocate la acel subsistem. Din moment ce subsistemul realizează funcții de gestiune dispozitiv, cum ar fi prezentarea semnarea afișajului și manevrarea recuperării erorii de dispozitiv, veți dori să limitați numărul de dispozitive alocate unui singur subsistem. Vedeți subiectul Limite de comunicații pentru informații suplimentare.

Notă: Această informație furnizează o centralizare a ceea ce este implicat în configurarea subsistemelor interactive. Rapoartele de experiență despre subsisteme conțin explicații detaliate ale fiecărui pas și opțiuni suplimentare disponibile pentru fiecare pas.

Configurarea unui subsistem interactiv: Crearea unei biblioteci:

Creați o bibliotecă în care memorați obiectele de configurare ale subsistemului dumneavoastră.

Acest exemplu utilizează SBSLIB ca bibliotecă.

```
CRTLIB SBSLIB TEXT('LIBRARY TO HOLD SUBSYSTEM CONFIGURATION OBJECTS')
```

Configurarea unui sistem interactiv: Crearea unei clase:

Crearea unei clase. Clasa definește anumite caracteristici de performanță pentru sistemul dumneavoastră interactiv.

Pentru a crea o clasă care să fie identică cu clasa QINTER, introduceți următoarea comandă:

```
CRTCLS SBSLIB/INTER1 RUNPTY(20) TIMESLICE(2000) PURGE(*YES) DFTWAIT(30)  
TEXT('Clasă de sistem interactiv personalizat')
```

Puteți utiliza clasa QINTER din QGPL pentru subsistemele dumneavoastră interactive personalizate sau puteți crea o singură clasă care să fie utilizată pentru toate subsistemele interactive sau puteți crea câte o clasă pentru fiecare subsistem interactiv.

Alegerea dumneavoastră ar trebui să depindă de faptul că doriți să personalizați unele din setările de performanță pentru un anumit subsistem. Subsistemele livrate de IBM sunt livrate cu o clasă, numele clasei fiind același cu numele subsistemului.

Dacă Nu creați o clasă pentru fiecare subsistem, cu același nume ca și subsistemul, va trebui să specificați numele clasei în comanda ADDRTGE (Add Routing Entry - Adăugare intrare de rutare). Aceasta deoarece valoarea implicită pentru parametrul CLS este *SBSD ceea ce înseamnă că numele clasei este același cu numele descrierii subsistemului.

Configurarea unui subsistem interactiv: Crearea descrierii subsistemului:

Creați descrierea subsistemului. Repetați acest pas pentru fiecare subsistem pe care aveți nevoie să îl definiți.

Următoarea sintaxă creează o descriere de subsistem cu atribute identice cu acele ale QINTER.

```
CRTSBSD SBSDB(SBSLIB/INTER1) POOLS((1 *BASE) (2 *INTERACT)) SGNDSPF(*QDSIGNON)
```

Configurarea unui subsistem interactiv: Crearea unei cozi de joburi:

Crearea unei cozi de joburi pentru subsistem utilizând același nume ca și numele subsistemului și adăugarea la descrierea subsistemului a unei intrări în coada de joburi.

Acest pas este necesar dacă aveți nevoie să utilizați comanda TFRJOB (Transfer Job) pentru a transfera joburi în subsistemele dumneavoastră personalizate.

```
CRTJOBQ JOBQ(SBSLIB/INTER1)
ADDJOBQE SBSDB(SBSLIB/INTER1) JOBQ(SBSLIB/INTER1) MAXACT(*NOMAX)
```

Configurarea unui subsistem interactiv: Adăugați o intrare de rutare:

Adăugați o intrare de rutare la subsistem.

Intrările de rutare care sunt livrate cu sistemul pentru QINTER au unele funcții suplimentare. Dacă aveți nevoie de acele funcții, adăugați acele intrări de rutare la descrierile dumneavoastră de subsistem personalizate.

```
ADDRTGE SBSDB(SBSLIB/INTER1) SEQNBR(9999) CMPVAL(*ANY) PGM(QSYS/QCMD) POOLID(2)
```

Configurați un subsistem interactiv: Adăugați intrări ale stației de lucru:

Adăugați intrări ale stației de lucru la descrierea subsistemului. Acesta este pasul cheie pentru a desemna ce dispozitive sunt alocate cărui subsistem.

Trebuie să determinați ce subsisteme ar trebui să aloce la ce dispozitive (AT(*SIGNON)). În plus, determinați dacă trebuie să alocați utilizarea TFRJOB de la un subsistem la altul (AT(*ENTER)).

```
ADDWSE SBSDB(SBSLIB/PGRM) WRKSTN(PGMR*) AT(*SIGNON)
ADDWSE SBSDB(SBSLIB/ORDERENT) WRKSTN(ORDERENT*) AT(*SIGNON)
ADDWSE SBSDB(QGPL/QINTER) WRKSTN(QPADEV*) AT(*SIGNON)
```

În acest exemplu, subsistemul și convenția de numire dispozitiv este bazată pe tipul de lucru pe care îl face utilizatorul. Programatorii toți au dispozitive care sunt numite PGMR și rulează în subsisteme PGRM. Personalul cu ordine de intrare are dispozitive care sunt numite ORDERENT și rulează în subsistemul ORDERENT. Toți ceilalți utilizatori folosesc convenția de numire implicită a sistemului a QPADEVxxxx și rulează în subsistemul QINTER livrat de IBM.

Configurarea unui subsistem interactiv: Personalizarea QINTER:

Când începeți să utilizați propriul dumneavoastră set de subsisteme, este posibil să nu aveți nevoie să utilizați QINTER. Totuși, dacă aveți un motiv pentru a continua să utilizați QINTER, trebuie să vă asigurați că QINTER este setat să NU aloce stațiile de lucru care doriți să ruleze sub celelalte subsisteme. Există două căi posibile de a face aceasta.

Înlăturați intrarea de stație de lucru *ALL din QINTER:

1. Înlăturați intrarea de stație de lucru *ALL din QINTER, și apoi adăugați intrări de stație de lucru specifice care indică ce dispozitive doriți ca QINTER să aloce. Înlăturarea intrării de tip stație de lucru *ALL se face pentru a împiedica QINTER să încerce să aloce toate stațiile de lucru.
2. Adăugați o intrare de stație de lucru pentru dispozitivele numite DSP* pentru a permite tuturor dispozitivelor de afișare atașate twinax să continue să fie alocate în QINTER.

În acest exemplu, dispozitivele de afișare atașate twinax vor continua să ruleze în QINTER; QINTER nu va încerca să aloce nici un alt dispozitiv.

```
RMWSE SBSDB(QGPL/QINTER) WRKSTNTYPE(*ALL)
ADDWSE SBSDB(QGPL/QINTER) WRKSTN(DSP*)
```

A doua metodă:

Adăugați o intrare de stație de lucru pentru a spune lui QINTER să nu aloce dispozitivele care sunt alocate altor subsisteme. Totuși, permiteți lui QINTER să continue să aloce orice alt dispozitiv care nu este alocat unui subsistem. Aceasta păstrează intrarea de tip stație de lucru *ALL în subsistemul QINTER și adaugă intrări de nume de stații de lucru cu parametrul AT pentru acele dispozitive care sunt alocate altor subsisteme.

```
ADDWSE SBSDB(QGPL/QINTER) WRKSTN(PGMR*) AT(*ENTER)
ADDWSE SBSDB(QGPL/QINTER) WRKSTN(ORDERENT*) AT(*ENTER)
```


Configurarea unui subsistem interactiv: Configurarea consolei:

O considerație finală, dar FOARTE importantă privind QINTER este intrarea *CONS a tipului stației de lucru pentru consolă. Asigurați-vă că nu împiedicați accidental pe cineva să se semneze la consolă. Împiedicați acest lucru să se întâmple neașteptându-vă nici o intrare de stație de lucru pentru consolă la subsistemele dumneavoastră interactive personalizate.

Sistemul este livrat cu subsistemul de control având intrarea de stație de lucru AT(*SIGNON) pentru consolă (intrare tip stație de lucru *CONS). QINTER are intrarea tipului de stație de lucru AT(*ENTER) pentru consolă.

Este un obicei bun să rulați întotdeauna consola în subsistemul de control și să nu transferați jobul consolei în alt subsistem interactiv. Aceasta împiedică utilizatorul de la consolă să își oprească propriul job neintenționat.

De exemplu, dacă utilizatorul de la consolă își transferă jobul la INTER1 și uită de el și mai târziu se pregătește pentru procesarea de rezervă realizând o comandă ENDSYS (End System - Oprire sistem), jobul consolei este și el oprit. Aceasta este, cel mai sigur, ceea ce operatorul nu a intenționat să facă.

Alocarea utilizatorilor unui anumit subsistem:

Această secțiune de subiect discută mai multe tehnici care pot fi utilizate pentru alocarea numelor de dispozitive și apoi asocierea acestor nume de dispozitive cu utilizatori. După ce se realizează aceasta, puteți utiliza intrările de stație de lucru pentru a duce utilizatorul pe subsistemul corect.

Sistemul are o convenție de numire implicită care este utilizată pentru sesiuni de afișare. Câteodată aceasta se dovedește insuficientă pentru rutarea intrărilor de stație de lucru prin mai multe subsisteme după profilul utilizator.

Puteți face modificări pe sistemul dumneavoastră pentru a îmbunătăți comportamentul implicit al sistemului alocând și gestionând propriile dumneavoastră convenții de numire a dispozitivelor. Există mai multe metode de a face acest lucru. Fiecare abordare are propriul set de avantaje și dezavantaje.

Inițializare dispozitiv Telnet și puncte de ieșire terminal:

Inițializare dispozitiv Telnet și puncte de ieșire terminal. Aceste puncte de ieșire furnizează abilitatea de a alocă nume de dispozitiv în funcție de clientul care se semnează pe sistem.

Punctul de ieșire vă furnizează adresa IP a clientului și numele de profil utilizator (împreună cu informații suplimentare). Puteți apoi realiza propria dumneavoastră mapare a clientului la descrierea dispozitivului care ar trebui utilizată pentru client.

Punctul de ieșire al inițializării dispozitivului mai furnizează, de asemenea, o cale de a ocoli panoul de semnare.

Avantajul de a utiliza aceste puncte de ieșire pentru a vă gestiona convenția de numire a dispozitivului dumneavoastră este că aveți control central asupra serverului iSeries pentru toți clienții.

Dezavantajul este că necesită calități de programare.

Punct de ieșire selecție dispozitiv:

Acest punct de ieșire vă permite să specificați convenția de numire utilizată pentru dispozitivele virtuale și pentru controller-ele virtuale create automat și să specificați limita creată automat utilizată pentru cereri speciale.

Cu acest punct de ieșire puteți specifica convenții de numire diferite pentru dispozitivele create automat utilizate de Telnet, de Stația de afișare Pass-through 5250, și de API-urile de terminal virtual.

În plus, puteți gestiona valoarea de sistem pentru dispozitivele Pass-through și Telnet (QAUTOVRT) într-o manieră mai precisă. De exemplu, puteți permite o valoare pentru dispozitivele create automat Telnet și o valoare diferită pentru dispozitivele de Stație de afișare 5250 Pass-through.

Acest punct de ieșire vă oferă abilitatea de a controla convențiile de numire implicite utilizate pentru dispozitive (precum QPADEV*) dar tot el nu vă va permite să specificați un anumit dispozitiv pentru un anumit utilizator. Acest punct de ieșire este cel mai util atunci când utilizați o combinație de căi pentru a vă conecta la sistem (Telnet, Stație de afișare 5250 Pass-through, WebFacing și așa mai departe) deoarece vă permite să utilizați convenții diferite de numire a dispozitivelor și o gestiune precisă QAUTOVRT pentru metode de acces diferite.

Support ID stație de lucru PC5250 (iSeries Access):

Puteți configura iSeries să se conecteze cu un nume specific de stație de lucru. Dacă faceți clic pe butonul de ajutor din această fereastră, sunt afișate diversele opțiuni pentru specificarea ID-ului stației de lucru, precum generarea unui nume nou dacă cel specificat este deja în curs de utilizare.

Un dezavantaj al acestei abordări este că trebuie să gestionați setările de configurație ale PC5250 pentru fiecare client care se conectează la serverul dumneavoastră.

Client TelnetOS/400:

Utilizând comanda Client TelnetOS/400 (STRTCPTLN sau TELNET), puteți specifica numele de dispozitiv care este utilizat pentru semnarea pe sistemul server.

Un dezavantaj pentru abordarea implicită este că necesită să vă asigurați că toate utilizările comenzilor STRTCPTLN (TELNET) specifică în mod corespunzător valoarea stației de afișare virtuală aflată la distanță. Pentru a înlătura această preocupare, puteți crea o versiune personalizată a comenzii STRTCPTLN pentru a asigura corectitudinea valorii terminalului de afișare virtual aflat la distanță și pentru a invoca comanda livrată IBM.

Crearea manuală a controlere-lor și dispozitivelor virtuale:

Puteți crea manual controlerele și dispozitivele dumneavoastră.

Pentru informații suplimentare despre crearea dispozitivelor virtuale pentru Telnet, vedeți subiectul Configurarea serverului Telnet din Centrul de informare iSeries.

Aceasta vă oferă control asupra numelor controlerelor și dispozitivelor dumneavoastră, dar nu vă furnizează abilitatea de a mapa un anumit dispozitiv la un anumit utilizator.

Crearea unui subsistem de control

IBM livrează două configurări complete de subsisteme de control: QBASE (subsistemul implicit de control) și QCTL. Doar un subsistem de control poate fi activ în sistem la un moment dat. De obicei, configurațiile subsistemului livrat de IBM ar trebui să fie suficiente pentru majoritatea necesităților unei afaceri. Totuși, puteți să vă creați propria versiune de subsistem de control și să o configurați pentru a întruni cât mai bine nevoile specifice companiei dumneavoastră.

Utilizați subsistemul de control livrat de IBM, QBASE sau QCTL, ca un model pentru a vă crea propriul subsistem de control.

Notă: Dacă vă creați propriul subsistem de control, ar trebui să utilizați un nume diferit față de QBASE sau QCTL.

Descrierea de subsistem pentru subsistemul de control ar trebui să conțină următoarele:

- O intrare de rutare care să conțină:
 - Fie *ANY sau QCMDI ca date de rutare
 - QSYS/QCMD ca program care să fie apelat

- Clasa QSYS/QCTL sau o clasă definită de utilizator. (Aceasta deoarece un utilizator, de obicei operatorul de sistem, trebuie să poată introduce comenzi care să realizeze lucruri ca eliberarea spațiului de stocare dacă a fost atins pragul de spațiu de stocare auxiliar.)
- O intrare de stație de lucru pentru consolă cu un tip *SIGNON (*SIGNON este o valoare pentru parametrul AT, specificat în comanda ADDWSE (Add Work Station Entry - Adăugare intrare de stație de lucru).)

Valoarea *SIGNON indică faptul că ecranul de semnare este afișat pe stația de lucru atunci când subsistemul este pornit. Această cerință asigură că subsistemul are un dispozitiv interactiv pentru comenzi de intrare de sistem și nivel subsistem. Comanda ENDSYS (End System - Oprire sistem) termină programul cu licență iSeries la o singură sesiune (sau ecran de semnare) la consola din subsistemul de control. O descriere de subsistem care nu conține o intrare de stație de lucru pentru consolă nu poate fi pornită ca subsistem de control.

- O intrare pentru altă stație de lucru:

Aceasta furnizează o sursă alternativă de intrare de control. Dacă se detectează o problemă de consolă în timpul unui IPL supravegheat și dacă valoarea de sistem QSCPFCONS pentru apariția unei probleme de consolă este setată pe '1', IPL-ul va continua în mod nesupravegheat. Apoi, dacă descrierea de subsistem pentru subsistemul de control conține o intrare de stație de lucru pentru altă stație de lucru, poate fi utilizată stația de lucru alternativă.

- O intrare de rutare care conține:
 - QSYS/QARDRIVE ca program care trebuie apelat,
 - și QSYS/QCTL ca clasă

După ce ați creat subsistemul de control, modificați valoarea de sistem Subsistem/bibliotecă de control (QCTLSBSD) după cum urmează (presupunând că descrierea este denumită QGPL/QCTLA):

```
CHGSYSVAL SYSVAL(QCTLSBSD) VALUE('QCTLA QGPL')
```

Modificarea intră în vigoare la următorul IPL.

Punerea sistemului în stare restricționată

Dacă toate subsistemele, inclusiv subsistemul de control sunt oprite, sistemul intră într-o condiție restricționată. Puteți pune sistemul într-o condiție restricționată utilizând una din cele două comenzi de la o stație de lucru interactivă.

Comandă: Oprire subsistem cu parametrul *ALL (ENDSBS SBS(*ALL))

Comandă: Oprire sistem (ENDSYS)

Important: Comezile ENDSBS sau ENDSYS ar trebui lansate de la un job interactiv din subsistemul de control și doar de la o stație de lucru a cărei intrare, din descrierea subsistemului de control, specifică AT(*SIGNON). Jobul interactiv de la care a fost lansată comanda rămâne activ când subsistemul de control intră în condiția restricționată. Dacă jobul care a lansat comanda este unul din cele două joburi care sunt active pe stația de lucru (utilizând tasta Cerere sistem sau comanda TFRSECJOB), nici unul dintre joburi nu este forțat să se oprească. Totuși, subsistemul de control nu se oprește pentru condiția restricționată până când nu opriți unul din joburi. Suspendarea joburilor de grup împiedică, de asemenea, subsistemul de control să se oprească (până când sunt oprite joburile de grup).

Când sistemul este în condiție restricționată, majoritatea activității din sistem este oprită și doar o stație de lucru este activă. Sistemul trebuie să fie în această condiție pentru ca comenzi precum SAVSYS (Save System - Salvare sistem) sau RCLSTG (Reclaim Storage - Pretindere spațiu de stocare) să ruleze.

Unele programe pentru diagnoza problemelor de echipament necesită, de asemenea, ca sistemul să fie într-o condiție restricționată. Pentru a opri condiția restricționată trebuie să porniți subsistemul de control din nou.

Gestionarea pool-urilor de memorie

Asigurarea că joburile obțin destulă memorie pentru a termina eficient este importantă. Dacă prea multă memorie este dată către subsistemul A și nu destulă subsistemului B, joburile din subsistemul B ar putea începe să ruleze mai slab. Informația următoare descrie operațiile diverse care sunt implicate în gestionarea pool-urilor de memorie.

Vizualizare informații pool de memorie

Puteți vizualiza informații despre pool-urile de memorie care sunt pe sistemul dumneavoastră utilizând Navigatorul iSeries sau interfața bazată pe caractere.

Navigatorul iSeries:

În Navigator iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active** sau **Pool-uri partajate**.

Containerul Pool-uri active afișează atât pool-urile partajate cât și pe cele private, atât timp cât sunt active. Containerul Pool-uri partajate afișează toate pool-urile partajate indiferent de starea lor curentă. Pool-urile private inactive nu există dincolo de definiția de pool până când nu sunt activate de către subsistem. Astfel nu pot fi vizualizate utilizând Navigator iSeries.

Interfața bazată pe caractere:

Comandă: Afișare descriere subsistem (DSPSBSD)

Utilizați opțiunea 2 - Definiții pool pentru a vizualiza toate definițiile de pool-uri private sau partajate care există în această definiție de subsistem.

Comandă: Gestionare pool-uri partajate (WRKSHRPOOL)

Determinare a numărului de subsisteme folosind un pool de memorie

Subsistemelor le este alocat un anumit procentaj din memorie pentru rularea de joburi. Este important de știut cum multe subsisteme diferite sunt trase din același pool de memorie. După ce aflați câte subsisteme sunt joburi lansate la un pool și câte joburi rulează într-un pool, veți dori să reduceți conflictul pentru resurse prin ajustarea dimensiunii și nivelul de activitate a pool-ului.

Navigator iSeries:

Pentru a folosi Navigator iSeries pentru a monitoriza numărul de subsisteme care folosesc un pool de memorie, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active** sau **Pool-uri partajate**.
2. Faceți clic dreapta pe pool-ul de memorie cu care doriți să lucrați și faceți clic pe **Subsisteme**.

De la această fereastră puteți determina numărul de subsisteme care folosesc o memorie individuală pentru a-și rula joburile.

Interfața bazată pe caractere:

Comanda: Lucrul cu subsisteme (WRKSBS)

Această comandă afișează o listă a tuturor subsistemelor și pool-urile lor corespunzătoare.

Determinarea numărului de joburi dintr-un pool de memorie

Navigator iSeries vă furnizează o cale să afișați rapid o listă de joburi care rulează curent într-un pool de memorie.

Pentru a determina numărul de joburi dintr-un pool de memorie, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active** sau **Pool-uri partajate**.

2. Faceți clic dreapta pe pool-ul de memorie pe care doriți să-l folosiți și faceți clic pe **Joburi**. O fereastră apare care arată o listă de joburi din cadrul pool-ului de memorie.

Puteți de asemenea vizualiza numărul de fire de execuție dintr-un pool de memorie prin vizualizarea coloanei Contor fir execuție. Contorul fir de execuție furnizează informații adiționale despre activitatea dintr-un pool de memorie.

Din acest moment, puteți executa aceleași funcții asupra joburilor ca și când ați fi în zona Joburi server sau Joburi active.

Determinarea pool-ului în care rulează un job singular

Dacă aveți un job care nu se realizează după cum v-ați așteptat este posibil să doriți să verificați în ce pool rulează jobul. Pentru a determina în ce pool rulează un singur job, utilizați Navigatorul iSeries sau interfața bazată pe caractere.

După ce ați identificat pool-ul în care rulează jobul, puteți vizualiza informații despre pool-ul de memorie și puteți determina dacă trebuie făcute schimbări. De exemplu, dacă are loc prea multă paginare, este posibil ca pool-ul de memorie să trebuiască să fie mai mare. O altă posibilitate pentru performanța redusă este că în pool sunt prea multe alte joburi și trebuie să rutați acest job către alt pool.

Navigator iSeries:

Pentru a utiliza Navigatorul iSeries , faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active** sau **Joburi de server** , în funcție de tipul de job cu care doriți să lucrați.
2. Găsiți jobul al cărui pool de memorie doriți să îl vedeți.
3. Faceți clic dreapta pe **Numele jobului** și faceți clic pe **Proprietăți**.
4. Faceți clic pe fișa **Resurse**. Fereastra Proprietăți job - Resurse afișează informații specifice despre pool-ul de memorie al jobului.

Interfața bazată pe caractere:

Comandă: Gestionare job (WRKJOB)

Opțiunea 1: Afișați atributele de stare job

Câmpul ID pool subsistem conține numele pool-ului definit pentru subsistemul în care rulează jobul. Acest câmp este blank pentru job urile care nu sunt active la momentul la care este cerută afișarea. Mai este blank, de asemenea, pentru joburi de sistem (tip SYS), joburi de monitor de subsistem (tip SBS) care nu rulează într-un subsistem și joburi batch imediate (BCI) care rulează în pool-ul de memorie Bază.

Comandă: Gestionare joburi active(WRKACTJOB)

Puteți utiliza comanda WRKACTJOB pentru a vedea ID-ul de pool sistem pentru un job activ.

Gestionarea parametrilor de reglare pentru pool-uri partajate

Pentru a gestiona parametrii de reglare pentru pool-uri partajate, folosiți Navigator iSeries sau comenzi de interfață bazată pe caractere.

Navigator iSeries:

Pentru a accesa parametrii de reglare prin folosirea Navigatorului iSeries, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active sau pool-uri partajate**.
2. Faceți clic dreapta pe pool-ul pe care doriți să-l reglați și faceți clic pe **Proprietăți**.
3. Faceți clic pe fișa **Reglare**.

De la Proprietăți partajate - Fereastra de reglare pe care puteți să ajustați manual valori specifice cum ar fi procentul de alocare de pool, pagini lipsă pe secundă, și prioritate.

Interfața bazată pe caractere:

Comanda: Lucrul cu Pool-uri partajate (WRKSHRPOOL)

Selectați **Opțiunea 11 - Afișare date ajustate** .

Gestionarea configurării unui pool

Pentru a modifica dimensiunea unui pool, nivelul de activitate sau opțiunea de paginație, folosiți Navigator iSeries sau comenzile de interfață bazată pe caractere.

Navigator iSeries:

Pentru a accesa valoarea unei configurații de partajat prin folosirea Navigatorului iSeries, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active sau pool-uri partajate**.
2. Faceți clic dreapta pe pool-ul pe care doriți să-l ajustați și faceți clic pe **Proprietăți**.
3. Faceți clic pe fișa **Configurație**.

De la Proprietăți partajate - Fereastra de configurare pe care puteți să ajustați manual valori specifice cum ar fi dimensiunea de pool, nivel de activitate sau opțiunea de paginație.

Interfața bazată pe caractere:

Comanda: Lucrul cu Pool-uri partajate (WRKSHRPOOL)

Modificarea dimensiunii pool-ului de memorie

Dimensiunea unui pool de memorie afectează direct cantitatea de muncă pe care o poate procesa un subsistem. Cu cât un subsistem are mai multă memorie, cu atât mai multă muncă poate, potențial, să realizeze. Este important să vă monitorizați sistemul cu atenție înainte de a începe să modificați parametrii pool-urilor dumneavoastră de memorie. De asemenea, veți reverifica periodic aceste niveluri deoarece este posibil să fie necesare unele reajustări.

Asigurați-vă că ați oprit tuner-ul sistemului înainte de a începe să modificați manual dimensiunile pool-urilor de memorie. Programul de ajustare (tuner) sistem ajustează automat dimensiunea pool-urile de memorie partajate pe cantitatea de lucru pe care sistemul o face. Dacă tuner-ul sistemului nu este oprit, modificările pe care le faceți manual este posibil să fie modificate automat de către tuner.

Opriti tuner-ul sistemului punând valoarea de sistem QPFRADJ (ajustare automată a pool-urilor de memorie și a nivelurilor de activitate) pe 0. (0 = Fără ajustări)

Navigatorul iSeries:

1. În Navigator iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Pool-uri de memorie** → **Pool-uri active sau Pool-uri partajate**.
2. Faceți clic dreapta pe pool-ul de memorie în care doriți să lucrați (de exemplu, Interactiv) și faceți clic pe **Proprietăți**. Fereastra **Proprietăți pool de memorie** apare.
3. Din fișa Configurare a ferestrei **Proprietăți**, puteți modifica cantitatea de memorie definită. Memoria definită este cantitatea maximă de memorie pe care o poate utiliza pool-ul. Numărul pe care îl puneți aici ar trebui să reflecte cantitatea de memorie de care credeți că va avea nevoie pool-ul pentru a suporta subsistemele pe care le servește.

Notă: Pool-ul de bază este singurul pool de memorie care nu are o cantitate de memorie definită. În schimb, el are o cantitate minimă de memorie de care are nevoie pentru a rula. Pool-ul de bază conține orice nu este alocat în altă parte. De exemplu, este posibil să aveți 1000 MB de memorie în sistemul dumneavoastră, din care 250 MB sunt

alocați pool-ului Mașină și 250 MB sunt alocați pool-ului Interactiv. 500 MB nu sunt alocați pentru nimic. Această memorie nealocată este stocată de pool-ul Bază până când este nevoie de ea.

Fiți prudenți când mutați memorie. Mutarea memoriei de pe un pool pe altul poate corecta un subsistem, dar poate cauza probleme pentru alte subsisteme, care în schimb, poate înrăutăți performanța sistemului.

Interfața bazată pe caractere:

Comandă: Modificare valoarea de sistem (CHGSYSVAL)

Exemplu: Următorul exemplu modifică dimensiunea pool-ului Mașină.

```
CHGSYSVAL QMCHPOOL 'dimensiune-nouă-în-KB'
```

Aceasta corespunde pool-ului 1 din ecranul WRKSYSSTS.

Exemplu: Următorul exemplu modifică dimensiunea minimă a pool-ului de bază.

```
CHGSYSVAL QBASPOOL 'dimensiune-minimă-nouă-în-KB'
```

Aceasta îi corespunde pool-ului 2 din ecranul WRKSYSSTS.

Notă: Valoarea de sistem QBASPOOL controlează doar dimensiunea minimă a pool-ului de bază. Pool-ul Bază conține tot spațiul de stocare care nu este alocat altor pool-uri.

Modificați dimensiunea unui pool partajat:

Comandă: Modificare pool de spațiu de stocare partajat (CHGSHRPOOL)

Modificările asupra pool-urilor partajate intră imediat în vigoare dacă pool-ul partajat este activ și este disponibil suficient spațiu de stocare.

Comandă: Gestionare pool-uri de spațiu de stocare partajate (WRKSHRPOOL)

Această comandă vă oferă acces la numele și informațiile de stare ale pool-urilor partajate. Utilizând opțiunile din meniu puteți modifica valori pentru dimensiunea unui pool și nivelurile maxime de activitate.

Crearea unui pool de memorie privat

Pool-urile de memorie private (cunoscute și ca pool-uri de memorie definite de utilizator) pot fi utilizate de către subsistemele livrate de IBM sau de către subsistemele definite de utilizator. Puteți defini un maxim de 10 definiții de pool de memorie pentru un subsistem. Creați un pool de memorie privat în descrierea subsistemului.

Pentru a crea un pool de memorie privat, utilizați interfața bazată pe caractere.

Comandă: Creare descriere subsistem (CRTSBSD) parametrul **POOLS**.

Comandă: Modificare descriere subsistem (CHGSBSD) parametrul **POOLS**.

Notă: Deși fiecare descriere de subsistem poate avea până la 10 pool-uri de memorie definite de utilizator, există o limitare de funcționare care face să nu poată fi active mai mult de 64 pool-uri de memorie simultan. (Aceasta include Pool-ul de memorie de bază și Pool-ul de memorie al mașinii.) Dacă limita maximă de alocare este atinsă înainte ca toate pool-urile de memorie pentru un subsistem să fie alocate, este utilizat Pool-ul de memorie de bază pentru toți pașii de rutare care încă mai necesită un pool de memorie.

Gestionarea cozilor de joburi

Așa cum gestionați lucrul pe sistemul dumneavoastră, ați putea găsi necesar să manevrați joburi care sunt în așteptare într-o coadă de joburi. Poate cineva are nevoie imediat de o rulare de job și jobul stă într-o coadă la un nivel minim de

prioritate. Sau poate trebuie să realizați puțină întreținere la subsistemul dumneavoastră și doriți să mutați toate joburile într-o coadă care nu este asociată în mod special cu acel subsistem.

Informația următoare descrie cum să realizați aceste tipuri operații de gestiune .

Alocați coada de joburi unui subsistem

Pentru a aloca o coadă de joburi unei descrieri de subsistem, utilizați interfața bazată pe caractere.

Comandă: Adăugare intrare în coadă job (ADDJOBQE)

Parametrii din această comandă specifică:

- Numărul de joburi care pot fi active în același timp în această coadă de joburi (**MAXACT**)
- În ce ordine tratează subsistemul munca de la această coadă de joburi (**SEQNBR**)
- Câte joburi pot fi active la un moment dat pentru fiecare din cele nouă niveluri de prioritate (**MAXPTYn**) (n=de la 1 la 9)

Exemplu: Următorul exemplu adaugă o intrare în coada de joburi pentru coada de joburi JOBQA din descrierea de subsistem TEST. Nu există un număr maxim de joburi care pot fi active în această coadă de joburi și lucrul este procesat cu numărul de ordine cinci.

```
ADDJOBQE SBS(D) TEST) JOBQ(LIBA/JOBQA) MAXACT(*NOMAX) SEQNBR(5)
```

Cum tratează un subsistem mai multe cozi de joburi:

Pentru a ilustra cum un subsistem manevrează mai multe cozi de joburi, considerați acest scenariu.

Coada de job A (SEQNBR=10)

Jobul 1
Jobul 2
Jobul 3

Coada de job B (SEQNBR=20)

Jobul 4
Jobul 5
Jobul 6

Coada de job C (SEQNBR=30)

Jobul 7
Jobul 8
Jobul 9

Fiecare intrare în coada de joburi din acest scenariu este specificată ca MAXACT(*NOMAX). Subsistemul mai întâi selectează joburile din coada de joburi **A** deoarece intrarea în coada de joburi are cel mai mic număr de ordine. Dacă numărul maxim de joburi din subsistem este 3 (parametrul MAXJOBS(3) din comanda CRTSBSD (Create Subsystem Description - Creare descriere subsistem)), el poate selecta toate joburile din coada de joburi **A** pentru a fi active simultan.

Când oricare din cele trei joburi s-a terminat, nivelul de activitate nu mai este la maxim; de aceea este selectat un job nou din coada de joburi **B** deoarece el are următorul număr de ordine cel mai mic (presupunând că nu au fost adăugate joburi noi la coada de joburi **A**). Deoarece fiecare intrare în coada de joburi specifică MAXACT(*NOMAX), valoarea MAXACT nu împiedică pornirea joburilor. Dacă fiecare intrare în coada de joburi ar fi specificat MAXACT(1), atunci joburile 1, 4, și 7 ar fi fost pornite. Dacă intrarea în coada de joburi **A** ar fi fost specificată ca MAXACT(2), atunci joburile 1, 2 și 4 ar fi fost pornite.

Modificarea numărului de joburi care rulează simultan într-o coadă de joburi

Subsistemul QBASE este livrat cu o intrare în coadă de joburi pentru coada de joburi QBATCH. Această intrare permite să ruleze doar un job batch la un moment dat. Dacă doriți ca mai multe joburi batch din coada de joburi să ruleze simultan atunci trebuie să modificați intrarea în coada de joburi.

Pentru a modifica numărul de joburi dintr-o coadă de joburi, care rulează simultan, utilizați interfața bazată pe caractere.

Comandă: Modificare intrare în coadă de joburi (CHGJOBQE)

Exemplu: Comanda următoare permite ca două joburi batch din coada de joburi QBATCH să ruleze simultan în subsistemul QBASE. (Această comandă poate fi lansată în orice moment și se execută imediat.)

```
CHGJOBQE SBSDB(QBASE) JOBQ(QBATCH) MAXACT(2)
```

Curățarea unei cozi de joburi

Când curățați o coadă de joburi, fiecare job din coadă este șters. Aceasta include orice job care se află într-o stare de reținere. Puteți utiliza Navigator iSeries sau interfața bazată pe caractere pentru a curăța o coadă de joburi. Joburile care rulează nu sunt afectate deoarece ele sunt considerate joburi active și nu mai sunt în coadă.

Navigatorul iSeries:

Pentru a utiliza Navigator iSeries pentru a curăța o coadă de joburi, urmați acești pași:

1. Expandați **Conexiunile mele** → **server** → **Cozi de joburi** → **Cozi de joburi active sau Toate cozile de joburi**.
2. Faceți clic dreapta pe coada de joburi și faceți clic pe **Curățare**. Apare fereastra Confirmare curățare unde puteți specifica faptul că doriți să fie produs un istoric de job când coada este curățată.

Interfața bazată pe caractere:

Comandă: Curățare coadă de joburi (CLRJOBQ)

Exemplu: Această comandă înlătură toate joburile care rulează momentan în coada de joburi livrată de IBM, QBATCH. Nu este afectat nici un job care momentan este citit.

```
CLRJOBQ JOBQ(QGPL/QBATCH)
```

Crearea cozilor de joburi

Pentru a crea o coadă de joburi, utilizați interfața bazată pe caractere.

Comandă: Creare coadă de joburi (CRTJOBQ)

Exemplu: Următorul exemplu creează o coadă de joburi numită JOBQA în biblioteca LIBA:

```
CRTJOBQ JOBQ(LIBA/JOBQA) TEXT('testare coadă de joburi')
```

După ce creați o coadă de joburi, ea trebuie să fie alocată unui subsistem înainte ca joburile să poată fi rulate. Pentru a alocă o coadă de joburi unui subsistem, adăugați o intrare în coada de joburi la descrierea subsistemului.

Ștergerea unei cozi de joburi

Pentru a șterge o coadă de joburi, utilizați interfața bazată pe caractere.

Restricții:

- Coada de joburi care este ștersă nu poate conține intrări. Toate joburile din coadă trebuie să fie terminate, șterse sau mutate într-o altă coadă de joburi.
- Un subsistem nu poate fi activ la coada de joburi.

Există mai mult de o cale pentru a șterge o coadă de joburi. Deși aici sunt menționate doar două metode, comanda WRKJOBQ este metoda recomandată deoarece ea arată starea și numărătoarea jobului.

Comandă: Gestionare coadă de joburi WRKJOBQ

Dacă numărul de joburi este 0, atunci puteți utiliza opțiunea 4=Ștergere pentru a șterge coada de joburi din bibliotecă.

Utilizați DLTJOBQ cu scripturi automatizate și curățați mediile. Aveți grijă când utilizați această metodă deoarece comportamentul implicit al acestei comenzi este să caute lista de biblioteci și să șteargă prima coadă de joburi care se potrivește cu numele specificat. Dacă aveți două cozi de joburi cu același nume în biblioteci diferite, este posibil să sfârșiți prin a o șterge pe cea bună. Puteți suprascrie acest comportament specificând o anumită bibliotecă.

Comandă: Ștergere coadă de joburi DLTJOBQ

Exemplu: Această comandă șterge coada de joburi SPECIALJQ din bibliotecă SPECIALLIB.

```
DLTJOBQ JOBQ(SPECIALLIB/SPECIALJQ)
```

Determinarea subsistemului care are alocată o coadă de joburi

Puteți determina ce subsistem are alocată coada de joburi utilizând interfața Navigatorului iSeries sau interfața bazată pe caractere. Aceasta este utilă când vi se pare necesar să ștergeți coada de joburi din moment ce nu puteți șterge o coadă de joburi la care este activ un subsistem.

Navigatorul iSeries:

Pentru a vedea ce subsistem are alocată o coadă de joburi, faceți următoarele:

1. În Navigatorul iSeries, expandați **Control funcționare** → **Cozi de joburi** → **Toate cozile de joburi**.
2. Localizați coada de joburi în panoul corect al Navigatorului iSeries. Veți vedea subsistemul care are alocată coada de joburi în coloana Subsistem.

(Dacă nu vedeți coloana Subsistem, adăugați-o la ecran). **Faceți clic dreapta pe Toate cozile de joburi** → **Personalizarea acestei vizualizări** → **Coloane**.)

3. Sau puteți face clic dreapta pe coada de joburi și să faceți clic pe **Proprietăți**. Subsistemul este menționat în pagina General a ferestrei Proprietăți coadă de joburi.

Interfața bazată pe caractere:

Comandă: WRKJOBQ JOBQ(LIBA/JOBQA) unde JOBQA este numele unei cozi de joburi

1. Tastați comanda WRKJOBQ JOBQ(LIBA/JOBQA). Apare ecranul Gestionare coadă de joburi. Funcția de descriere subsistem apare în zona tastelor funcționale a ecranului când coada de joburi este alocată unui sistem.
2. Apăsați tasta funcțională de descriere subsistem. Apare ecranul Gestionare descriere de subsistem arătând subsistemul căruia îi este alocată coada de joburi.

Reținerea unei cozi de joburi

Când puneți o coadă de joburi în așteptare preveniți procesarea tuturor joburilor care sunt în acest moment în așteptare în coada de joburi. Punând o coadă de joburi în așteptare nu are nici un efect asupra joburilor care sunt în rulare. Joburi adiționale pot fi puse în coada de joburi în timp ce este reținută, dar nu sunt procesate.

Pentru a pune o coadă de joburi în așteptare, puteți folosi Navigator iSeries sau interfața bazată pe caractere.

Navigator iSeries:

În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare** → **Cozi de joburi** → **Cozi de joburi active** → **clic dreapta pe coadă** → **Reține**.

Interfața bazată pe caractere:

Comandă: Reține coada de joburi (HLDJOBQ)

În acest exemplu, coada de joburi QBATCH este plasată în așteptare. Toate joburile care nu rulează la momentul la care comanda este lansată sunt reținute până când coada este eliberată sau curățată.

HLDJOBQ JOBQ(QBATCH)

Eliberarea unei cozi de joburi

Când eliberați o coadă de joburi, toate joburile care erau reținute ca rezultat al reținerii cozii de joburi sunt, de asemenea, eliberate. Dacă un job individual a fost reținut înainte să fie reținută coada de joburi, atunci jobul nu este eliberat.

Pentru a elibera o coadă de joburi, utilizați Navigatorul iSeries sau interfața bazată pe caractere.

Navigatorul iSeries:

În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Cozi de joburi** → **Toate cozile de joburi** → **Faceți clic dreapta pe coadă** → **Eliberare**.

Interfața bazată pe caractere:

Comandă: Eliberare coadă de joburi (RLSJOBQ)

Acest exemplu eliberează coada de joburi QBATCH.

RLSJOBQ JOBQ(QBATCH)

Mutarea unui job într-o coadă de joburi diferită

Există multe motive pentru care ați putea dori să mutați un job în altă coadă. De exemplu, unele joburi devin restante în coadă din cauza unui job care are o durată mare de rulare. Probabil timpul de rulare planificat al jobului este în conflict cu un nou job care are o prioritate mai mare. O metodă de a gestiona această situație este să fie mutate joburile aflate în așteptare în altă coadă care nu este atât de ocupată.

Puteți utiliza fie interfața Navigatorului iSeries sau interfața bazată pe caractere pentru a muta jobul dintr-o coadă în alta.

Navigatorul iSeries:

Pentru a utiliza Navigatorul iSeries, faceți următoarele:

1. În Navigatorul iSeries, expandați **Control funcționare** → **Cozi de joburi** → **Toate cozile de joburi**.
2. Localizați și deschideți coada care conține jobul momentan.
3. Faceți clic dreapta pe jobul pe care doriți să îl mutați. Se deschide fereastra Mutare unde puteți specifica coada destinație.

Notă: Dacă doriți să mutați mai mult de un job din această coadă, țineți apăsată tasta CTRL în timp ce faceți clic pe fiecare job. Apoi faceți clic dreapta și apoi faceți clic pe **Mutare**.

- Joburile care așteaptă să ruleze sunt mutați în aceeași poziție relativă în coada destinație (de exemplu, joburile cu o prioritate de coadă de joburi de 3 sunt mutate după toate celelalte joburi cu prioritatea 3 care așteaptă să ruleze în coada destinație).
- Joburile care sunt reținute rămân reținute și sunt puse în aceeași poziție relativă în coada destinație (de exemplu, joburile reținute cu prioritatea de coadă de joburi 3 sunt mutate după toate celelalte joburi reținute cu prioritatea 3 din coada destinație).
- Joburile care sunt planificate să ruleze sunt mutate în coada destinație și orele lor planificate rămân nemodificate.

Interfața bazată pe caractere:

Comandă: Modificare job (CHGJOB)

Exemplu: Următorul exemplu mută jobul JOBA în coada de joburi JOBQB

Punerea unui job în coada de joburi

Joburile sunt puse în coada de joburi fie prin mutarea unui job existent dintr-o coadă în alta, sau prin lansarea unui nou job. Utilizați Navigatorul iSeries pentru a face mutarea între cozi. Utilizați interfața bazată pe caractere pentru a lansa un nou job.

Navigatorul iSeries:

Pentru a utiliza interfața Navigatorului iSeries, jobul trebuie să existe deja în altă coadă de joburi. Apoi puteți muta jobul dintr-o coadă în alta. (Pentru a pune un job nou într-o coadă de joburi, utilizați interfața de linie de comandă.)

1. În Navigatorul iSeries, expandați **Control funcționare** → **Cozi de joburi** → **Toate cozile de joburi**.
2. Faceți clic dreapta pe jobul pe care doriți să îl mutați. Se deschide fereastra Mutare unde puteți specifica coada de destinație.

Interfața bazată pe caractere:

Următoarea este o listă de metode ale interfeței bazată pe caractere pentru punerea unui job nou într-o coadă de joburi nouă.

- Lansare job (SBMJOB): Permite unui job care rulează să pună un alt job într-o altă coadă care să fie rulat mai târziu ca job batch. Doar un element de date de cerere poate fi pus în coada de mesaje a noului job. Datele de cerere pot fi o comandă CL dacă intrarea de rutare utilizată pentru job specifică un program de procesare a comenzilor CL (cum ar fi programul QCMD livrat de IBM).
- Adăugare intrare de planificare job (ADDJOBSCDE): Sistemul lansează automat un job în coada de joburi la ora și data specificate în intrarea de planificare job.
- Lansare joburi de bază de date (SBMDBJOB): Lansează joburi în cozi de joburi astfel încât ele să ruleze ca joburi batch. Fluxul de intrare este citit fie dintr-un fișier fizic de bază de date sau dintr-un fișier bază de date logic care are un format cu o singură înregistrare. Această comandă vă permite să specificați numele fișierului bază de date și a membrului său, numele cozii de joburi care va fi utilizată și să decideți dacă joburile care sunt lansate pot fi afișate de către comanda WRKSBMJOB (Work with Submitted Jobs - Gestionare joburi lansate).
- Pornire cititor bază de date (STRDBRDR): Citește un flux de intrare batch dintr-o bază de date și pune unul sau mai multe joburi în cozi de joburi.
- Transfer job (TFRJOB): Mută jobul curent în altă coadă de joburi dintr-un subsistem activ.
- Transfer job batch (TFRBCHJOB): Mută jobul curent în altă coadă de joburi.

Căutarea unui anumit job în toate cozile de joburi

Puteți utiliza fie Navigatorul iSeries sau interfața bazată pe caractere pentru a căuta un anumit job în cozile de joburi.

Navigatorul iSeries:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Operații de bază** → **Faceți clic dreapta pe Joburi** → **Personalizarea acestei vizualizări** → **Includere**.
2. Utilizați fereastra Joburi - Includere pentru a limita numărul de joburi care sunt afișate. Asigurați-vă că, câmpul **Coadă de job** este setat pe Toate.
3. Când faceți clic pe **OK** vor fi afișate toate joburile care îndeplinesc criteriile dumneavoastră.

Interfața bazată pe caractere:

Comandă: Gestionare cozi de joburi (WRKJOBQ)

Exemplu: Următorul exemplu creează o listă a tuturor joburilor din coada de joburi JOBQA.

```
WRKJOBQ JOBQ(LIBA/JOBQA)
```

Găsirea unui job când nu cunoașteți numele cozii de joburi:

Dacă nu cunoașteți numele cozii de joburi, faceți următoarele:

1. Introduceți comanda fără parametrul **JOBQ**. Apare fereastra Gestionare toate cozile de joburi, cu o listă a tuturor cozilor de joburi la care sunteți autorizat.
2. Scanați această listă până vedeți numele cozii de joburi care este posibil să conțină jobul pe care încercați să îl găsiți.

După ce ați găsit un job într-o coadă de joburi, puteți să vă uitați la job introducând opțiunea de gestionare, pentru jobul pe care doriți să îl vizualizați. Apare ecranul Gestionare job. Acest ecran furnizează mai multe opțiuni pentru vizualizarea întregii informații disponibile pentru jobul pe care l-ați selectat.

Dacă știți ce job căutați, următoarea comandă vă va duce direct la ecranul jobului.

```
WRKJOB JOB(număr/utilizator/nume) OPTION(*DFNA)
```

Dacă nu știți cu siguranță ce job căutați, comenzile WRKSBMJOB (Work with Submitted Jobs - Gestionare joburi lansate în execuție) sau WRKUSRJOB (Work with User Jobs - Gestionare joburi de utilizator) vă pot ajuta.

Specificarea priorității pentru coada de joburi

Pentru a specifica ordinea în care cozile de joburi sunt procesate de către subsistem, folosiți interfața bazată pe caractere.

Comanda: Adăugați Intrarea în coada de joburi (**ADDJOBQE**)

Parametrii din această comandă specifică:

- Numărul de joburi care pot fi active în același timp în această coadă de joburi (**MAXACT**)
- În ce ordine subsistemele tratează lucrul de la această coadă de joburi (**SEQNBR**)
- Câte joburi pot active la un moment dat pentru fiecare din cele nouă niveluri de prioritate (**MAXPTYn**) (n=1 până la 9)

Gestionarea cozilor de ieșire

Cozile de ieșire vă ajută să gestionați ieșirea la imprimantă creată când un job se termină. Este important să înțelegeți cum să mențineți efectiv cozile de ieșire ca să procesați fără probleme ieșirea la imprimantă.

Ieșirea la imprimantă aflată în coada de ieșire. Coada de ieșire determină ordinea în care ieșirea la imprimantă va fi procesată de un dispozitiv de tipărire. Prin gestionarea cozilor de ieșire, puteți asigura procesarea normală a ieșiri la imprimantă.

Crearea unei cozi de ieșire

Comanda CRTOUTQ (Create Output Queue - Creare coadă de ieșire) creează o coadă de ieșire nouă pentru fișiere spool. Este pusă o intrare în coada de ieșire pentru fiecare fișier spool. Ordinea în care sunt scrise fișierele în dispozitivul de ieșire este determinată de prioritatea de ieșire a fișierului spool și valoarea specificată în parametrul **SEQ** (Order of files on queue prompt - Ordinea fișierelor în promptul cozii). Utilizați interfața bazată pe caractere pentru a crea o coadă de ieșire.

Comandă: CRTOUTQ (Create Output Queue - Creare coadă de ieșire)

Exemplu: Această comandă creează o coadă de ieșire numită DEPTAPRT și o pune în biblioteca curentă. Deoarece este specificat AUT(*EXCLUDE) și este presupus OPRCTL(*YES), coada de ieșire poate fi utilizată și controlată doar de utilizatorul care a creat coada și de utilizatorii care au autorizare de control job sau autorizare de control spool. Deoarece este specificat SEQ(*FIFO), fișierele spool sunt puse în coadă în ordinea first-in first-out (primul-intrat primul-ieșit). Dacă utilizatorii din Departamentul A sunt autorizați să utilizeze această coadă de ieșire, trebuie utilizată comanda GRTOBJAUT (Grant Object Authority - Acordare autorizare obiect) pentru a le acorda acestora suficientă autorizare. Datele conținute în fișierele din această coadă pot fi afișate doar de către utilizatorii care sunt proprietarii

acestor fișiere, de către proprietarul cozii, de către utilizatorii cu autorizare de control job sau de către utilizatorii cu autorizare de control spool. În mod implicit, nu este tipărit nici un separator de joburi la începutul ieșirii pentru fiecare job.

```
CRTOUTQ  OUTQ(DEPTAPRT) AUT(*EXCLUDE) SEQ(*FIFO)
          TEXT('SPECIAL PRINTER FILES FOR DEPTA')
```

Exemplu: Următorul este un alt exemplu de cum puteți crea o coadă de ieșire.

```
CRTOUTQ  OUTQ(QGPL/JONES) +
          TEXT('Coadă de ieșire pentru Mike Jones')
```

Alocarea cozii de ieșire unui job sau unei descrieri de job

Înainte de a putea să utilizați o coadă de ieșire nou creată, trebuie să o alocați unui job sau unei descrieri de job. Puteți aloca coada de ieșire utilizând Navigatorul iSeries sau interfața bazată pe caractere.

Navigatorul iSeries:

Pentru a utiliza Navigatorul iSeries pentru a aloca coada de ieșire unui job, faceți următoarele.

În Navigatorul iSeries, expandați **Control funcționare** → **Joburi active** → **Faceți clic dreapta pe un job și selectați Proprietăți** → **faceți clic pe fișa Ieșire imprimantă** .

Interfața bazată pe caractere:

Puteți, de asemenea, modifica descrierea de job pentru a utiliza noua coadă de ieșire. Astfel, toate joburile care utilizează descrierea de job vor utiliza noua coadă de ieșire. Utilizați interfața bazată pe caractere pentru a aloca o coadă de ieșire unei descrieri de job.

Comandă: Modificare descriere job CHGJOB

Următorul exemplu va modifica descrierea de job AMJOBS astfel încât să fie utilizată coada de ieșire QPRINT.

```
CHGJOB JOB(AMJOBS/AMJOBS) OUTQ(*LIBL/QPRINT)
```

Accesarea ieșirii de imprimantă

Deoarece aveți posibilitatea de a detașa ieșirea imprimantei de la un job, o dată ce acesta termină de rulat (separând complet ieșirea imprimantei de job), puteți accesa ieșirea imprimantei dumneavoastră din Navigatorul iSeries prin Operații de bază sau prin Control funcționare.

Navigator iSeries:

Pentru a accesa ieșirea imprimantă a unui job prin Operații de bază, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Operații de bază** → **Job**.
2. Faceți clic dreapta pe jobul pentru care doriți să afișați ieșirea imprimantă și faceți clic pe **Ieșire imprimantă**. Apare fereastra Ieșire imprimantă.

Pentru a accesa ieșirea imprimantei prin folderul Cozi de ieșire, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Cozi de ieșire**.
2. Selectați coada de ieșire pe care doriți să o afișați ieșirii imprimantă (de exemplu, Qprint2). Va apare ieșirea imprimantă în coada de ieșire.

Interfață bazată pe caractere:

Comandă: Gestionare coadă de ieșire (WRKOUTQ <nume coadă de ieșire>)

Comandă: Gestionare fișiere spool (WRKSPLF JOB(nume job calificat))

Ștergerea cozilor de ieșire

Când un job creează ieșirea imprimantă el este trimis la o coadă de ieșire ca să fie tipărit. Cel mai probabil nu veți tipări toată ieșirea imprimantă creată. Navigator iSeries vă dă posibilitatea de a vă curăța cozile de ieșire utilizând opțiunea **Curățare**. Ștergerea unei cozi de ieșire va șterge toate ieșirile din coadă.

Navigator iSeries:

Ca să ștergeți o coadă de ieșire, urmați acești pași:

1. În Navigator iSeries , expandați **Conexiunile mele** → **server** → **Control funcționare** → **Cozi de ieșire**.
2. Faceți clic dreapta pe coada de ieșire pe care doriți să o curățați și faceți clic pe **Curățare**.

Interfața bazată pe caractere:

Comandă: Curățare coadă de ieșire (CLROUTQ)

Această comandă înlătură intrările pentru toate fișierele spool din coada de ieșire, QPRINT, care așteaptă să fie tipărite sau sunt reținute. Intrările pentru fișierul care este tipărit în acest moment și fișierele care încă mai primesc date de la programe care momentan rulează nu sunt afectate.

```
CLROUTQ  OUTQ(QPRINT)
```

Ștergerea unei cozi de ieșire

Utilizați interfața bazată pe caractere pentru a șterge o coadă de ieșire.

Înainte ca o coadă de ieșire să poată fi ștearsă, ea trebuie să îndeplinească următoarele cerințe.

Coadă de ieșire care este ștearsă nu poate să conțină nici o intrare. Ieșirea pentru fiecare fișier trebuie tipărită, ștearsă sau mutată într-o coadă de ieșire diferită. Un subsistem nu poate fi activ. Coadă nu poate fi în curs de utilizare de către un scriitor de punere în spool. Coadă nu poate fi ștearsă dacă ea a fost creată de către sistem pentru o anumită imprimantă.

Comandă: Ștergere coadă de ieșire (DLTOUTQ)

Această comandă șterge coada de ieșire PUNCH2 din sistem.

```
DLTOUTQ  OUTQ(PUNCH2)
```

Vizualizarea cozilor de ieșire din sistem

Cozile de ieșire determină ordinea în care ieșirea imprimantă este trimisă la dispozitivul imprimantă. Puteți vizualiza cozi de ieșire folosind Navigatorul iSeries.

Pentru a vizualiza cozile de ieșire din sistem, faceți următoarele:

1. În Navigator iSeries, expandați **serverul** → **Conexiunile mele** → **Control funcționare**.
2. Faceți clic pe **Cozi de ieșire**.

În Navigator iSeries, puteți personaliza lista de cozi de ieșire pe care o vizualizați prin folosirea ferestrei Include. Fereastra Include vă permite să puneți limitări pe ceea ce este afișat în Navigator iSeries. De exemplu, puteți rula Include pentru a afișa numai anumite cozi de ieșire.

Pentru a folosi funcția de includere, faceți clic pe meniul Vizualizare, și apoi **Personalizați această Vizualizare**.

Gestionarea istoricelor de job

Majoritatea joburilor de pe iSeries au un istoric de job asociat cu el. Istoricul de job vă spun multe lucruri diferite cum ar fi când pornește un job, când se oprește un job, ce comenzi rulează, observații de eșuare și mesaje de eroare. Aceste informații vă dau o idee bună despre modul cum rulează ciclul jobului.

Următoarele informații discută diversele operații pe care le puteți realiza când lucrați cu istorice de joburi.

Gestionarea serverului de istorice de joburi

Subsistemul QSYSWRK controlează serverul de istorice de joburi. Totuși, există câteva operații pe care le puteți realiza pentru a personaliza sau gestiona serverul de istorice de joburi.

Reconfigurarea serverului de istorice de joburi:

După cum este livrat, serverul de istorice de joburi rulează în QSYSWRK. QSYSWRK este activ continuu. Pentru a îmbunătăți performanțele, este posibil să doriți să vă reconfigurați serverul de istorice de joburi să ruleze într-un subsistem diferit.

Pentru a reconfigura serverul de istorice de joburi să ruleze într-un subsistem diferit, utilizați interfața bazată pe caractere și urmați acești pași.

1. Adăugați la descrierea subsistemului dumneavoastră o intrare de rutare identică celei din QSYSWRK. Aceasta este intrarea de rutare Seq Nbr 500, Programul QWCJLSVR, Biblioteca QSYS, Valoare de comparare 'QJOBLOGSVR', Poziția de start 1.
2. Modificați coada de joburi, specificată în descrierea de job QJOBLOGSVR, cu o coadă de joburi care este prezentă pe subsistemul dumneavoastră.
3. Adăugați intrarea jobului autostart QJOBLOGAJ (împreună cu intrarea de rutare, dacă este necesar) la subsistemul dumneavoastră. Aceasta va face ca serverul de istorice de joburi să pornească automat când subsistemul dumneavoastră pornește.
 - Sau, dacă preferați, puteți înlocui intrarea de job autostart cu o apelare a comenzii STRLOGSVR în programul startup.
4. Înlăturați intrarea de job autostart QJOBLOGAJ din QSYSWRK.

Ca alt exemplu de reconfigurare a unui server de istorice de joburi, puteți utiliza comanda (CHGCLS)(Change Class - Modificare clasă) pentru a modifica prioritatea de Rulare (RUNPTY) specificată în clasa QJOBLOGSVR (din biblioteca QSYS).

```
CHGCLS CLS(QSYS/QJOBLOGSVR) RUNPTY(50)
```

Oprirea serverului de istorice de joburi:

Comanda ENDLOGSVR (End Job Log Server - Oprire server de istorice de joburi) este utilizată pentru a opri server(e) de istorice de joburi. Serverul de istoric de joburi scrie istorice de joburi pentru joburi al căror istoric job este într-o stare de așteptare. Dacă sunt mai multe joburi ale serverului de istorice de joburi active la momentul când această comandă este lansată, toate joburile serverului de istoric de joburi vor fi oprite.

Trebuie să aveți autorizare specială de control job (*JOBCTL) pentru a utiliza această comandă.

Important: Dacă doriți doar să opriți producerea unui anumit istoric de job, de exemplu, este prea lung sau consumă prea multe resurse, vedeți subiectul înrudit ***Oprirea producerii unui anumit istoric de job.***

Când utilizați comanda ENDLOGSVR, puteți specifica dacă doriți ca serverul să se oprească imediat (nu este recomandat) sau într-o manieră controlată.

Navigatorul iSeries:

1. Din Navigatorul iSeries, faceți clic dreapta pe sistemul punct final în care lucrează serverul de istorice de joburi și faceți clic pe **Rulare comandă**.
2. În Comanda de rulat: tastați ENDLOGSVR.
3. Fereastra Oprire server de istorice de joburi apare pentru a vă ajuta să specificați parametrii acestei comenzi. Terminați în fereastră și faceți clic pe **OK**. Fereastra se închide și vă întoarceți la fereastra Rulare comandă.
4. Puteți acum fie să faceți clic pe **OK** pentru a rula comanda imediat, sau puteți face clic pe **Planificare** pentru a planifica momentul când doriți să ruleze comanda.

Interfața bazată pe caractere:

Comandă: Oprește server de istorice de joburi (ENDLOGSVR)

Pornirea serverului de istorice de joburi

În mod implicit, serverul de istorice de joburi pornește automat când pornește subsistemul QSYSWRK. Puteți porni manual un server de istorice de joburi utilizând comanda STRLOGSVR (Start Job Log Server - Pornire server de istorice de joburi).

Când utilizați comanda STRLOGSVR, puteți specifica numărul de servere de istorice de joburi suplimentare pe care doriți să le porniți sau puteți lăsa sistemul să calculeze numărul necesar în locul dumneavoastră. Dacă numărul de servere cerute depășește numărul maxim de servere active permis, doar numărul de servere egal cu diferența între numărul maxim și numărul curent de servere active vor fi pornite. Numărul maxim de servere de istorice de joburi care pot fi active sau pot fi în coada de joburi la un moment dat este 30.

Navigatorul iSeries:

Pentru a utiliza Navigatorul iSeries, faceți următoarele:

1. Din Navigatorul iSeries, faceți clic dreapta pe sistemul din punctul final unde se află serverele de istorice de joburi și selectați **Rulare comandă**.
2. În câmpul **Comanda de rulat** : tastați STRLOGSVR.
3. Faceți clic pe **Prompt**.
4. Este afișată fereastra Pornire server de istorice de joburi pentru a vă ajuta să specificați parametrii acestei comenzi. Terminați în fereastră și faceți clic pe **OK**. Fereastra se închide și vă întoarceți la fereastra Rulare comandă.
5. Puteți acum fie să faceți clic pe **OK** pentru a rula comanda imediat, sau puteți face clic pe **Planificare** pentru a planifica momentul când doriți să ruleze comanda.

Interfața bazată pe caractere:

Comandă: **Pornire server de istorice** (STRLOGSVR)

Cum se afișează un istoric de job

Puteți vedea un istoric de job din orice loc din Control funcționare de unde accesați joburi, de exemplu prin zona Subsistem sau zona Pool de memorie. Puteți utiliza Navigatorul iSeries sau interfața bazată pe caractere, pentru a afișa istorice de joburi.

Navigatorul iSeries:

Pentru a accesa istoricul de job pentru un job activ sau un job de server, faceți următoarele:

1. În Navigatorul iSeries, expandați **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active** sau **Joburi de server**.
2. Faceți clic dreapta pe un job (de exemplu, Qbatch) și faceți clic pe **Istoric job**. Pentru informații suplimentare, faceți referire la ajutorul din fereastra Istoric job.
Pentru a vedea detalii suplimentare ale unui mesaj, faceți clic dreapta pe un mesaj și faceți clic pe **Proprietăți**. Fereastra Proprietăți mesaj furnizează informații detaliate despre mesaj. Această fereastră arată detaliile mesajului precum și ajutorul pentru mesaj. Ajutorul de mesaj detaliat vă oferă informații pentru a rezolva problema.

Lista următoare descrie căi adiționale de a accesa istorice de job:

- **Operații de bază** → **Imprimantă**
- **Operații de bază** → **Joburi** → **Faceți clic dreapta pe un job** → **Ieșire imprimantă**
- **Control funcționare** → **Joburi active** → **Faceți clic dreapta pe un job** → **Ieșire imprimantă**
- **Control funcționare** → **Cozi de ieșire**

- **Utilizatori și grupuri** → **Toți utilizatorii** → **Faceți clic dreapta pe un utilizator** → **Obiecte utilizator** → **Ieșire imprimantă**

Interfață bazată pe caractere:

- **Comandă:** Gestionare istorice de job (WRKJOBLOG)
- **Comandă:** Gestionare job WRKJOB OPTION(*JOBLOG).
- **Comandă:** Gestionare job (WRKJOB) JOB (<nume job calificat>) OPTION(*SPLF)
- **Comandă:** Gestionare coadă de ieșire (WRKOUTQ)
- **Comandă:** Gestionare fișier spool (WRKSPLF)

Dacă nu aveți destule informații pentru a utiliza comenzile de mai sus, comanda (WRKUSRJOB)(Work with User Jobs - Gestionare joburi utilizator) sau comanda (WRKSBMJOB) (Work with Submitted Jobs - Gestionare joburi lansate) ar putea fi de ajutor.

Ce trebuie făcut când istoricul de job nu se afișează

În Navigatorul iSeries, pentru a găsi și afișa un istoric de job, fie pentru un job batch sau un job interactiv, faceți clic dreapta pe job și apoi faceți clic pe **Istoric job** din meniu. Totuși, în funcție de starea jobului dumneavoastră sau de cum au fost setate valorile istoricului de job în descrierea jobului, istoricul dumneavoastră de job poate fi în coada de ieșire sau poate fi într-o stare de istoric job în așteptare sau poate să nu fie disponibil.

Următorii sunt câțiva pași care trebuie făcuți dacă opțiunea din meniu Istoric job nu este disponibilă pentru jobul dumneavoastră.

Indiciu: Setări afișarea pe coloană pentru Joburi active (sau Joburi de server) pentru a include Starea. Aceasta va înlesni determinarea rapidă a locului unde trebuie căutat istoricul dumneavoastră de job.

Pentru a accesa un istoric de job: **Conexiunile mele** → **server** → **Control funcționare** → **Joburi active sau Joburi de server** → **Faceți clic dreapta pe job și selectați Istoric job.**

Dacă opțiunea de meniu Istoric job nu este disponibilă sau dacă primiți un mesaj de eroare care declare că sistemul nu este capabil să extragă istoricul de job, luați în considerare următoarele:

1. Verificați starea jobului.

Opțiune	Descriere
Rulează	Verificați fereastra Proprietăți job - Istoric job și asigurați-vă că este bifată caseta Producere istoric de job . Dacă nu este bifată, atunci nu este produs nici un istoric de job.
Terminat la	Acest job nu s-a oprit într-o manieră normală. Este posibil să fie din cauza unei erori sau a unei intervenții a utilizatorului. Faceți clic dreapta pe job și faceți clic pe Ieșire imprimantă . Dacă nu vedeți istoricul dumneavoastră de job acolo, verificați fereastra Proprietăți job - Istoric job și asigurați-vă că este selectată caseta de bifare Producere ieșire imprimantă pentru istoric de job.
Terminat - Ieșirea de imprimantă este disponibilă	Acest job s-a oprit normal. Faceți clic dreapta pe job și apoi faceți clic pe Ieșire imprimantă . Dacă nu vedeți istoricul dumneavoastră acolo, verificați fereastra Proprietăți job - Istoric job și asigurați-vă că a fost bifat câmpul Creare ieșire imprimantă pentru istoricul de job dacă jobul se oprește normal .
Terminat - Istoric job în așteptare	Istoricul de job nu va fi produs. Istoricul de job rămâne în așteptare până când este înlăturat. Va trebui să utilizați comanda DSPJOBLOG (Display Job Log - Afișare istoric job) pentru a vedea istoricul de job în așteptare.

2. Istoricul de job poate să fi fost pus în spool într-o coadă de ieșire și a fost tipărit, în acest caz istoricul de job a fost înlăturat din sistem.
3. O altă posibilitate este că istoricul de job a fost șters de către alt utilizator.

Specificați coada de ieșire pentru un istoric de job

Implicit fișierul de imprimantă care este folosit pentru un job de spool este QPJOBLOG. Puteți avea multiple fișiere imprimantă QPJOBLOG pe sistemul dumneavoastră. În QSYS coada de ieșire pe care atributul OUTQ o folosește este QEZJOBLOG, din biblioteca QUSRSYS. Când sistemul creează un istoric de job, el caută pentru fișierul imprimantă QPJOBLOG din lista de biblioteci ale jobului. Prima care este găsită este cea pe care o folosește. Utilizați interfața bazată pe caractere pentru a ajusta aceste setări.

1. Modificați atributul fișierului imprimantă QPJOBLOG OUTQ cu *JOB.
 - a. **Comanda:** Modificați fișierul imprimantă CHGPRTF FIȘIER(QPJOBLOG) OUTQ(*JOB)
2. Modificați atributul OUTQ al jobului cu coada de ieșire pe care o doriți. Puteți să faceți asta folosind interfața bazată pe caractere sau Navigator iSeries.
 - a. **Comanda:** Modificare job CHGJOB OUTQ(MYLIB/MYOUTQ)
 - b. Navigator iSeries: **Control funcționare** → **Joburi active** → **clic dreapta pe job și selectați Proprietăți** → **Fișă imprimantă**

Oprirea producerii unui anumit istoric de job

Dacă doriți să opriți doar producerea unui anumit istoric de job, nu utilizați comanda ENDLOGSVR (End Job Log Server - Oprire server de istorice de joburi). Comanda ENDLOGSVR oprește toate serverele de istorice de joburi care duce la oprirea producerii tuturor istoricelor de joburi.

În loc, utilizați procedura următoare pentru a opri producerea unui anumit istoric de job.

1. Din Navigatorul iSeries, faceți clic dreapta pe jobul pentru care doriți să opriți producerea istoricului și faceți clic pe **Proprietăți**. (**Conexiunile mele** → **server** → **Control funcționare** → **Joburi active sau Joburi de server**)
2. Faceți clic pe fișa **Istoric job**.
3. Debifați caseta **Producere istoric de job** și faceți clic pe **OK**.

Producerea istoricului de job va înceta și istoricul de job va fi într-o stare de istoric job în așteptare.

Împiedicarea producerii unui istoric de job

Împiedicarea producerii unui istoric de job este utilă dacă știți deja că nu veți avea nevoie de istoricul de job și doriți să conservați resursele sistemului. Când specificați că nu doriți să se producă un istoric de job, istoricul de job nu va fi produs și rămâne în așteptare până când este înlăturat fie de comanda QWTRMVJL (Remove Pending Job Log-Înlăturare istoric de job aflat în așteptare) sau comanda ENDJOB (End Job - Oprire job).

Pentru a împiedica producerea unui istoric de joburi, faceți următoarele:

1. În Navigatorul iSeries deschideți fereastra **Proprietăți job - Istoric job**. (**Conexiunile mele** → **server** → **Control funcționare** → **Joburi active (sau Joburi de sistem)** → **Faceți clic dreapta pe job** → **Proprietăți** → **fișa Istoric job**)
2. Debifați caseta **Producere istoric de job** și faceți clic pe **OK**.

Controlul informațiilor dintr-un istoric de job

Când se lucrează cu probleme, este posibil să doriți să înregistrați cantitatea maximă de informații pentru joburile care au probleme frecvente. În mod alternativ, este posibil să nu doriți să creați un istoric de joburi pentru joburile care s-au efectuat normal. Sau este posibil să doriți să excludeți mesajele informative.

Puteți controla ce informații sunt adăugate la istoricul de job setând nivelul de mesaj, gravitatea mesajului sau valorile nivelului de text de mesaj din descrierea de job. Totuși, dacă doriți să controlați ce informații sunt scrise în istoricul de job al unui anumit job, utilizați fereastra **Proprietăți job - Istoric job** din Navigatorul iSeries.

Această fereastră vă permite să controlați următoarele:

- Dacă istoricul jobului este produs și ce metodă este folosită pentru a-l produce
- Ce trebuie făcut când s-a atins dimensiunea maximă
- Dacă să fie înregistrate în istoric comenzi de la programe CL
- Dacă să se păstreze mesajele în istoricul de job și care mesaje anumite ar trebui păstrate (nivel de înregistrare în istoric și gravitatea mesajului)
- Dacă să se creeze ieșire de imprimantă pentru un istoric de job dacă jobul se termină normal și ce să se tipărească

Pentru a accesa fereastra **Proprietăți job - Istoric job** faceți următoarele:

1. Din Navigator iSeries deschideți fereastra **Proprietăți job** a jobului și faceți clic pe fișa **Istoric job. Conexiunile mele → server → Control funcționare → Job activ → Faceți clic dreapta pe job → Proprietăți.**
2. Pentru o explicație detaliată a diferitelor opțiuni care sunt disponibile în această fereastră, faceți referire la ajutorul online.

Modificarea nivelului de istoric pentru un job:

Nivelul de istoric este un nivel numeric alocat unei combinații specifice de tipuri de mesaje care sunt înregistrate în istoric. Puteți modifica nivelul de istoric din descrierea jobului utilizând interfața bazată pe caractere. Totuși, dacă doriți să modificați nivelul de istoric al unui anumit job, utilizați fereastra **Proprietăți job - Istoric job** din Navigatorul iSeries.

Pentru a accesa fereastra **Proprietăți job - Istoric job** faceți următoarele:

1. Din Navigator iSeries deschideți fereastra **Proprietăți job** a jobului și faceți clic pe fișa **Istoric job. (Conexiunile mele → server → Control funcționare → Job activ → Faceți clic dreapta pe job → Proprietăți)**
2. Pentru o explicație detaliată a diferitelor opțiuni disponibile în această fereastră, faceți referire la ajutorul online.

Controlul informațiilor înregistrate în istoricele de job batch:

Pentru aplicațiilor dumneavoastră batch, este posibil să doriți să modificați cantitatea de informații înregistrate în istoric. Nivelul istoricului (LOG(40 *NOLIST)) specificat în descrierea jobului pentru subsistemul QBATCH livrat de IBM furnizează un istoric complet dacă jobul se termină anormal. Dacă jobul se termină normal nu se produce nici un istoric pentru job.

Controlul istoricului de joburi la nivelul cozii de joburi (QBATCH) este făcut ajustând setările istoricului de job pentru jobul subsistemului QBATCH. Aveți aceleași opțiuni pentru controlul modului în care sunt produse istoricele de joburi la nivelul jobului de subsistem pe care le aveți și pentru nivelul joburilor individuale.

Pentru a ajusta setările istoricului de job pentru subsistemul cozii de joburi, faceți următoarele:

În Navigatorul iSeries deschideți fereastra **Proprietăți subsistem - Istoric job** pentru subsistemul cozii de joburi. (**Control funcționare → Sub sisteme → Sub sisteme active → QBATCH → Faceți clic dreapta pe jobul QBATCH → Proprietăți → Fișă Istoric job**)

Notă: Dacă debifați câmpul **Producere istoric de job (*PND)** pentru subsistem, istoricul de job specific subsistemului nu va fi menționat cu cealaltă ieșire de imprimantă. VEți avea atunci nevoie să utilizați comanda (DSPJOBLOG) (Display Job Log - Afișare istoric job) pentru a vizualiza istoricul de job aflat în așteptare.

Dacă jobul batch rulează un program CL, comenzile programului CL sunt înregistrate în istoric doar dacă este specificat LOGCLPGM(*YES) în comanda (CRTCLPGM) (Create Control Language Program - Creare program de control limbaj) sau în comanda (CHGPGM) (Change Program - Modificare program).

Ștergerea fișierelor de ieșire istoric job

Istoricul de job sunt înlăturate din sistem când un job se încheie normal sau când sunt lansate API-ul QWTRMVJL (Remove Pending Job Log - Înlăturare istoric job în așteptare) sau comanda ENDJOB (End Job - Oprire job). În plus,

dacă se specifică "curățare istorice de joburi incomplete" în IPL, toate joburile din istoricul de joburi aflat în așteptare vor fi înlăturate din sistem în timpul unui IPL. Toate fișierele de ieșire de istoric job rămase pot fi găsite în **Operații de bază** → **Ieșire imprimantă**.

Pentru a șterge istorice de joburi aflate în Ieșire imprimantă, faceți clic dreapta pe numele fișierului istoricului de job pe care doriți să îl ștergeți și faceți clic pe **Ștergere**.

Cum să determinați dacă nu este primejdios să ștergeți un istoric de job

Echilibrul între păstrarea istoricelor de joburi sau ștergerea lor este o adevărată provocare. Istoricul de job sunt lucruri care trebuie păstrate ca să puteți depăși o problemă. Istoricul de joburi sunt lucruri pe care nu doriți să le păstrați din cauza că vă dezorganizează sistemul. Când decideți este istoric de job să ștergeți sau ce istorice de joburi să împiedicați să se producă, luați în considerare următoarele indicații:

- Este acesta un job pe care să îl puteți corecta cu ușurință fără să aveți nevoie de istoricul jobului?
- Este acesta un job similar cu alte joburi din sistem? Dacă eșuează, este posibil să mai eșueze și joburile similare? Dacă este așa, atunci este posibil să doriți ca doar unul din joburi să producă un istoric de job.

Producerea ieșirii de imprimantă din istoricul de job în așteptare

Joburi care nu au câmpul **Producere istoric job** din setarea din Navigator iSeries **Proprietăți job - Istoric job** selectat, nu produc istorice de job. În schimb istoricul de job este în starea istoric job în așteptare. Pentru a produce o ieșire imprimantă dintr-un istoric de job care este în starea istoric de job în așteptare, folosiți interfața bazată pe caractere.

Comanda: Afișare istoric job (DSPJOBLOG)

Curățarea istoricului de job în așteptare

Există puține căi prin care să curățați sau să înlăturați joburile a căror istoric este în așteptare. Puteți opri jobul cu o valoare 0 pentru parametrul Maxim de intrări în istoric **LOGMLT**. Dacă jobul este deja oprit puteți rula API-ul QWTRMVJL (Remove Pending Job Log - Înlăturare istoric de job în așteptare). Puteți utiliza, de asemenea, comanda WRKJOBLOG (Work with Job Logs - Gestionare istorice de job).

Pentru a opri un job cu **LOGMLT** pus pe 0, utilizați Navigatorul iSeries sau interfața bazată pe caractere.

Navigatorul iSeries:

1. În Navigatorul iSeries, expandați **Control funcționare** → **Joburi active**.
2. Localizați jobul pe care doriți să îl opriți.
3. Faceți clic dreapta pe job și faceți clic pe **Ștergere/Oprire**.
4. În fereastra Confirmare Ștergere/Oprire, setați **Ștergere ieșire imprimantă** pe Nu.
5. Terminați în fereastra Confirmare Ștergere/Oprire și faceți clic pe **Ștergere**.

Interfața bazată pe caractere:

Comandă: Oprire job (ENDJOB LOGMLT(0))

Gestionarea contabilizării de job

Funcția de contabilizare job nu este activă implicit. Ea necesită câțiva pași inițiali pentru a o seta. Următoarele informații descriu cum să setați o contabilizare de job și să realizați unele din cele mai comune operații asociate cu contabilizarea de job.

Setarea contabilizării unui job

Pentru setarea contabilizării de job, utilizați interfața bazată pe caractere.

1. Creați un receptor de jurnal. Receptorul de jurnal poate fi creat cu orice nume și bibliotecă pe care le alegeți. Este recomandat să îi dați un nume cu o convenție de numire precum ACGJRN1 astfel încât receptorii suplimentari (precum ACGJRN2, ACGJRN3) să poată fi creați cu comanda CHGJRN JRNRCV(*GEN) (Change Journal - Modificare jurnal).

- a. **Comandă:** Creare receptor jurnal (CRTJRNRCV)
CRTJRNRCV JRNRCV(USERLIB/ACGJRN1)
2. Crearea jurnalului de contabilizare job. Numele jurnalului trebuie să fie QSYS/QACGJRN și aveți nevoie de autorizare pentru a adăuga obiecte la biblioteca QSYS.

- a. **Comandă:** Creare Jurnal (CRTJRN)
CRTJRN JRN(QSYS/QACGJRN) JRNRCV(USERLIB/ACGJRN1) AUT(*EXCLUDE)

Receptorul de jurnal ar trebui să fie același cu receptorul creat la pasul 1. Autorizarea poate fi setată la orice alegeți, dar se recomandă *EXCLUDE din moment ce datele colectate pot fi utilizate pentru a taxa utilizatorii pentru folosirea resurselor.

3. Modificați valoarea de sistem QACGLV (informații jurnal de contabilizare). Valoarea de sistem poate fi setată să jurnalizeze informațiile de contabilizare job sau informațiile despre imprimantă, sau ambele. *JOB produce intrări de jurnal de job (JB), în timp ce *PRINT produce intrările de jurnal Tipărire directă (DP) sau Tipărire pusă în spool (SP). Valoarea *NONE înseamnă că nu se realizează nici o jurnalizare pentru jurnalul QACGJRN. Datele de contabilizare job vor fi jurnalizate doar pentru joburile care sunt pornite după ce valoarea de sistem a fost setată la altă valoare decât *NONE.

- a. **Comandă:** Gestionare valori de sistem (WRKSYSVAL) sau Modifică valoarea de sistem (CHGSYSVAL)
CHGSYSVAL SYSVAL(QACGLV) VALUE('*JOB *PRINT')

4. Setati parametrul **ACGCDE**, de cod de contabilizare, pentru fie care profil utilizator. Codul de contabilizare poate fi setat la orice șir alfanumeric de până la 15 caractere lungime. Dacă este important să se determine utilizatorul curent pentru analiza pe care o faceți dumneavoastră unei intrări de jurnal de contabilizare job, este recomandat să setați parametrul **ACGCDE** pe numele profilului utilizator.

- a. **Comandă:** Modificare profil utilizator (CHGUSRPRF) sau Creare profil utilizator (CRTUSRPRF)
CHGUSRPRF USRPRF(USERID1) ACGCDE(USERID1)

Codul de contabilizare poate fi, de asemenea, specificat pentru un grup de utilizatori, folosind comenzile CHGJOB (Change Job Description - Modificare descriere de job) sau CRTJOB (Create Job Description - Creare descriere de job).

Codul implicit de contabilizare pentru descrierile de job este *USRPRF, ceea ce înseamnă că utilizează codul de contabilizare din profilul utilizatorului jobului. Dacă se specifică altă valoare decât *USRPRF în descrierea jobului, ea va avea prioritate față de codul de contabilizare specificat în profilul utilizator

Se poate scrie un program simplu pentru modificarea tuturor profilurilor utilizator dintr-un sistem astfel încât codul lor de contabilizare să se potrivească cu numele profilului utilizator.

Controlul alocării codurilor de contabilizare

Un aspect important al fiecărei aplicații de procesare a datelor este asigurarea că sunt specificate câmpurile corecte de control. Pentru codurile de contabilizare a joburilor, aceasta poate necesita o funcție complexă de verificare a validității care nu verifică doar existența codurilor autentice dar verifică și ce utilizatori au permisiunea de a utiliza anumite coduri.

Codurile de contabilizare pot fi alocate în următoarele zone:

- Profil utilizator
- Descriere de job
- Într-un job (comanda CHGACGCDE (Change Accounting Code - Modificare coduri de contabilizare))

Dacă este important să controlați alocarea codurilor de contabilizare, luați în considerare următoarele:

1. Înainte ca un cod de contabilizare să fie pus într-un profil utilizator, asigurați-vă de validitatea codului pentru un anumit utilizator.
2. Controlați modificarea codurilor de contabilizare din comanda CHGJOB (Change Job Description - Modificare descriere de job) dând autorizarea responsabilului cu securitatea doar comenzii CHGACGCDE.

- Sau, utilizați comanda CHGACGCDE pentru a permite utilizatorilor să modifice codul de contabilizare job al propriului job sau al altor joburi. Pentru a modifica alt job, utilizatorul trebuie să aibă, de asemenea, autorizarea specială *JOBCTL.
3. Utilizați un program și o comandă CL pentru a împiedica modificarea codurilor de contabilizare pentru un job din coada de joburi sau pentru un job pentru a modifica codul de contabilizare al altui job. De exemplu, comanda CHGACGCDE va fi autorizată privat și inclusă într-un program CL unde ea a modificat doar jobul curent (ca atunci când este specificat JOB(*)). Această comandă va fi autorizată în mod corespunzător.

Afișarea datelor colectate

După colectarea datelor în jurnalul de contabilizare a jobului, trebuie să scrieți intrările de jurnal la un fișier și să le afișați.

Pentru a face aceasta, urmați acești pași:

Notă: În exemplul următor, numele jurnalului de contabilizare a jobului este QACGJRN.

1. Creați o copie a modelului de fișier de ieșire al sistemului livrat pentru jurnalul de contabilizare. QAJBACG4 este modelul fișierului de ieșire pentru formatul fișierului de ieșire *TYPE4.
 - a. **Comanda:** Creare obiect duplicat (CRTDUPOBJ)


```
CRTDUPOBJ OBJ(QAJBACG4) FROMLIB(QSYS) OBJTYPE(*FILE) TOLIB(QTEMP)
NEWOBJ(MYJBACG4)
```
2. Creați un dump pentru intrările de jurnal către fișierul de ieșire pe care doar ce l-ați creat. În exemplul următor numai 'JB' sau intrările de jurnal ale tipului de job sunt puse în dump.
 - a. **Comanda:** Afișare jurnal (DSPJRN)


```
DSPJRN JRN(QACGJRN) ENTTYP(JB) OUTPUT(*OUTFILE) OUTFILFMT(*TYPE4)
OUTFILE(QTEMP/MYJBACG4)
```
3. Porniți o sesiune SQL. Apoi folosiți comanda SELECT din cadrul sesiunii SQL pentru a alege câmpurile pe care doriți să le afișați.
 - a. **Comanda:** Pornire limbaj de interogare structurată (STRSQL)


```
STRSQL
SELECTARE JAJOB, JAUSER, JAUSPF, JACDE, JACPU DE LA QTEMP/MYJBACG4
```

Puteți afișa interactiv o listă de nume a câmpurilor sau la un fișier prin crearea și rularea unei interogări folosind comanda Lucrul cu interogările(WRKQRY).

Convertirea intrărilor de jurnal de contabilizare job

Puteți utiliza parametrul **OUTFILE** din comanda DSPJRN (Display Journal - Afișare jurnal) pentru a scrie intrările de jurnal de contabilizare joburi într-un fișier bază de date pe care îl procesați.

Parametrul **OUTFILE** vă permite să denumiți un fișier sau un membru. Dacă membrul există, el este curățat înainte ca înregistrările să fie scrise. Dacă membrul nu există, el este adăugat. Dacă fișierul nu există, este creat un fișier utilizând formatul de înregistrare QJORDJE. Acest format definește câmpurile de antet standard pentru fiecare intrare de jurnal, dar datele de contabilizare job sunt definite ca un singur câmp mare.

Pentru a evita necesitatea procesării datelor de contabilizare ca pe un singur câmp mare, sunt livrate două fișiere de referință de câmp pentru a vă ajuta la procesarea intrărilor de jurnal de contabilizare joburi. Fișierul QSYS/QAJBACG4 conține formatul de înregistrare QWTJAJBE și este utilizat pentru intrări JB. Fișierul QSYS/QAPTACG5 conține formatul de înregistrare QSPJAPTE și este utilizat pentru intrări DP sau SP. Același format este utilizat pentru toate intrările de fișier de imprimantă indiferent dacă ieșirea este SP (pus în spool) sau DP (nepus în spool). Intrarea DP pentru fișierele tipărite direct conține unele câmpuri care nu sunt utilizate; aceste câmpuri conțin blancuri.

Următoarele reprezintă unele abordări pe care le-ați putea utiliza:

- Intrările de bază JB și DP sau intrările SP pot fi procesate prin crearea a două fișiere de ieșire utilizând formatele de fișier de referință câmp livrate și rulând comanda DSPJRN o dată pentru JB și o dată pentru DP sau SP. Aceasta vă permite să definiți un fișier logic peste cele două fișiere fizice și să utilizați un program de limbaj de nivel înalt pentru procesarea fișierului descris extern.
- Puteți procesa doar intrările JB creând un fișier utilizând unul din fișierele de referință câmp livrate (QSYS/QAJBACG4) pentru a crea un fișier descris extern. Acest fișier poate fi apoi procesat de utilitatea de interogare sau de un program cu limbaj de nivel înalt.
- Puteți converti ambele tipuri de intrări de jurnal utilizând formatul implicit DSPJRN sau QJORDJE. Puteți apoi utiliza un fișier descris de program pentru a procesa intrările de jurnal dintr-un program cu limbaj de nivel înalt.

Următorul DDS definește un fișier fizic pentru intrările jurnalului JB utilizând fișierul de referință câmp QAJBACG4 din QSYS. Puteți crea fișierul (utilizând comanda CRTPF (Create Physical File - Creare fișier fizic)) cu același nume (QAJBACG4) ca fișierul model.

```
R QWTJAJBE FORMAT(QSYS/QAJBACG4)
```

Următorul DDS definește un fișier fizic pentru intrările de jurnal DP sau SP utilizând fișierul de referință câmp QAPTACG5 din QSYS. Puteți crea fișierul (utilizând comanda CRTPF) cu același nume (QAPTACG5) ca și fișierul model.

```
R QSPJAPTE FORMAT(QSYS/QAPTACG5)
```

Puteți specifica un câmp cheie în orice fișier fizic; totuși, în acest exemplu, este utilizat pentru ordonare un fișier logic. Dacă creați două fișiere fizice (unul pentru JB și unul pentru DP sau SP) cu membrii de același nume, puteți lansa următoarele comenzi DSPJRN pentru convertirea intrărilor. Presupuneți că ați creat fișierele fizice cu aceleași nume ca fișierele model din biblioteca dumneavoastră YYYY.

```
DSPJRN JRN(QACGJRN) JRNCD(A) ENTYP(JB)
OUTPUT(*OUTFILE) OUTFILE(YYYY/QAJBACG4)
DSPJRN JRN(QACGJRN) JRNCD(A) ENTYP(SP DP)
OUTPUT(*OUTFILE) OUTFILE(YYYY/QAPTACG5)
```

Puteți controla criteriile de utilizare și selecție ale comenzii DSPJRN astfel încât să nu convertiți aceleași intrări de mai multe ori. De exemplu, puteți selecta toate intrările dintr-un anumit interval de date. Puteți converti toate intrările la un punct de separație pentru analiza de contabilizare a jobului dumneavoastră, de exemplu, lunar. În timpul lunii este posibil să se fi utilizat unul sau mai mulți receptori de jurnal. Notați că fiecare utilizare a comenzii DSPJRN asupra aceluiași membru face ca membrul să fie curățat înainte să se adauge orice altă intrare nouă. Nu utilizați parametrul **JOB** al comenzii DSPJRN deoarece unele intrări sunt făcute pentru job de către jobul de sistem și de aceea nu apar cum vă așteptați dumneavoastră.

Permiterea procesării ambelor Fișiere fizice:

Introduceți următorul DDS pentru a crea un fișier logic pentru a permite procesarea ambelor fișiere fizice. Aceasta vă permite să citiți un singur fișier în ordinea codului de contabilizare și să tipăriți un raport utilizând un program cu limbaj de nivel înalt:

```
R QWTJAJBE PFILE(YYYY/QAJBACG4)
K JACDE
R QSPJAPTE PFILE(YYYY/QAPTACG5)
K JACDE
```

Procesarea unei înregistrări de contabilizare job de bază:

Dacă doriți să utilizați un fișier logic pentru procesarea doar a înregistrării de contabilizare job de bază în ordinea codului de contabilizare, puteți introduce următorul DDS pentru un fișier logic:

```
R QWTJAJBE PFILE(YYYY/QAJBACG4)
K JACDE
K JAUSER
```


Acest fișier logic poate fi procesat de o utilitate de interogare sau de un program cu limbaj de nivel înalt. Dacă are loc o oprire anormală a sistemului, numele calificat de job din primii 30 octeți ai câmpului JARES din intrarea de jurnal descriu jobul sistemului care a scris intrarea la următorul IPL și nu jobul care a utilizat resursele. Din acest motiv, orice analiză realizată asupra intrărilor JB ar trebui să utilizeze câmpurile JAJOB, JAUSER și JANBR.

Recuperare și contabilizare job

Dacă un job se termină anormal, intrarea de contabilizare finală este scrisă și toate intrările de contabilizare scrise anterior apar în jurnal. Dacă are loc o oprire anormală a sistemului, următoarele date de contabilizare s-au pierdut la următorul pas de rutare sau la ultimul segment sfârșit-de-contabilizare, cel care a survenit mai recent.

- Informații despre numărul de linii și pagini tipărite
- Număr de fișiere create
- Operații de punere, scoatere și actualizare bază de date
- Operații de scriere și citire comunicații
- Operații de I/E auxiliare
- Timp tranzacție
- Număr de câmpuri de tranzacție
- Timp activ
- Timp suspendat

După o oprire anormală a sistemului, timpul de terminare job din jurnal nu va fi același cu cel din mesajul CPF1164. Mesajul utilizează timpul cel mai apropiat celui de oprire a sistemului, dar intrările jurnalului de contabilizare job sunt trimise către jurnal în timpul IPL-ului și momentul de terminare job este timpul curent al sistemului, care este mai târziu decât momentul la care a survenit oprirea anormală a sistemului.

Dacă sistemul se termină anormal, unele din intrările jurnalului se pot pierde. Acestea sunt intrările care sunt scrise în jurnal dar nu sunt forțate pe disc (aceasta este echivalent cu FORCE(*NO) în comanda SNDJRNE (Send Journal Entry - Trimitere intrare jurnal)). Ele includ următoarele:

- Intrări JB cauzate de o comandă CHGACGCDE (Change Accounting Code - Modificare cod de contabilizare)
- Intrări DP și SP

De fiecare dată când un job se termină, ultima intrare de cod de contabilizare este forțată pe disc (ca și când s-ar fi specificat FORCE(*YES) în comanda SNDJRNE). De câte ori o intrare de contabilizare este forțată pe disc, toate intrările de mai devreme din jurnal, indiferent de ce job le-a produs, sunt forțate pe disc.

Excepție

Dacă este specificată numai contabilizare *PRINT pe sistem, nu va fi realizată nici o intrare de jurnal de terminare job FORCE(*YES). De aceea, dacă este scrisă o intrare de contabilizare critică de către comanda CHGACGCDE și vreți să vă asigurați că ea nu se va pierde în cazul unei opriri anormale a sistemului, puteți emite o comandă SNDJRNE și specifica opțiunea FORCE(*YES). Dacă vor fi, de asemenea, jurnalizate fișiere în jurnalul de contabilizare, orice modificare a bazei de date este întotdeauna forțată în jurnal și aceasta duce la forțarea tuturor intrărilor de contabilizare de mai devreme.

Dacă are loc o oprire anormală a sistemului sau modificați un cod de contabilizare al altui job decât cel propriu, numele jobului calificat din primii 30 de octeți ai câmpului JARES din intrarea de jurnal descriu jobul de sistem care a scris intrarea JB la următorul IPL și nu jobul care a utilizat resursele. Câmpurile JAJOB, JAUSER și JANBR ar trebui utilizate în scopuri de analiză.

Jurnal de contabilizare joburi deteriorat sau receptor jurnal deteriorat:

Dacă are loc o deteriorare asupra jurnalului sau asupra receptorului său curent astfel încât intrările de contabilizare nu pot fi jurnalizate, este trimis un mesaj CPF1302 către coada de mesaje QSYSOPR și datele de contabilizare sunt scrise

în istoricul QHST din mesajul CPF1303. Jobul care încearcă să trimită intrarea de jurnal continuă în mod normal. Recuperarea dintr-un jurnal deteriorat sau dintr-un receptor de jurnal deteriorat este aceeași ca pentru celelalte jurnale.

Jurnalul QACGJRN nu ar trebui să fie alocat de către alt job. Dacă jurnalul este alocat de către alt job, intrarea de jurnal este modificată cu text de mesaj și trimisă în istoricul QHST ca reprezentând mesajul CPF1303.

Puteți utiliza parametrul OUTFILE din comanda DSPNJRN (Display Journal - Afișare jurnal) pentru a scrie intrările jurnalului de contabilizare într-un fișier bază de date pe care îl puteți procesa.

Puteți, de asemenea, utiliza comanda RCVJRNE (Receive Journal Entry - Recepționare intrare de jurnal) în jurnalul QACGJRN pentru a recepționa intrările așa cum sunt scrise în jurnalul QACGJRN. Dacă jurnalul de contabilizare joburi sau receptoarele de jurnal sunt deteriorate, sistemul continuă să opereze și să înregistreze datele în istoricul sistemului. Pentru a recupera din jurnalul sau receptorul de jurnal deteriorat, utilizați comanda WRKJRN (Work with Journal - Gestionare jurnal). După recuperarea jurnalului sau receptorului de jurnal deteriorat, modificați valoarea de sistem QACGLVL (Informații de contabilizare a jurnalului) cu o valoare corespunzătoare pentru instalarea dumneavoastră. (Doar dacă nu modificați valoarea de sistem QACGLVL, atunci sistemul nu înregistrează informații de contabilizare în noul receptor de jurnal.)

Accesarea Mesajului CPF1303:

Pentru a accesa informațiile din mesajul CPF1303, creați un program cu limbaj de nivel înalt.

Pentru a defini înregistrările care se potrivesc mesajului CPF1303, includeți următoarele câmpuri:

Oră sistem Car (8)
Număr înregistrare mesaj Bin (4)
Nume job calificat Car (26)
Tip intrare (JB, DP sau SP) Car (2)
Lungimea datelor Bin (2)

Urmat de câmpurile:

JAJOB până la JASPN pentru intrări JB
JAJOB până la JABYTE pentru intrări SP și DP

Pentru un exemplu de program, faceți referire la secțiunea din cartea Programarea CL care discută procesarea fișierului QHST pentru mesajul de terminare job.

Mesajul CPF1164 conține întotdeauna trei înregistrări și mesajul CPF1303 conține întotdeauna patru înregistrări. Informațiile conținute în câmpurile de prefix de jurnal standard nu este inclusă în acest mesaj. Tot ce este necesar sunt informațiile care privesc oprirea jobului, data și ora. Aceste informații pot fi găsite în înregistrarea 1 a mesajului CPF1303.

Referință

Este posibil să găsiți aceste articole din Centrul de informare utile în timp ce utilizați controlul de funcționare.

(Centrul de informare IBM iSeries, Versiunea 5 Ediția4 (V5R4) → Gestiune sisteme → Control funcționare → Referință)

Tabelă de job de server

Utilizați tabela de server ca o referință pentru a afla cum sunt mapate unele cu altele serverele, joburile de server, descrierile de job și subsistemele.

Detector de valoare sistem

Utilizați detectorul de valoare de sistem OS/400 pentru a afla informații, repede, despre valorile de sistem.

Puteți căuta categorii de valori de sistem așa cum apar ele în Navigatorul iSeries sau puteți căuta numele de valori de sistem pe care le-ați utilizat în interfața bazată pe caractere.

Interfața de programare a aplicației (API) de control de funcționare

Pagina API-urilor de Control funcționare afișează o listă de API-uri care extrag și manipulează; joburi, pool-uri de spațiu de stocare ale subsistemului, cozi de joburi ale subsistemului, zone de date, atribute de rețea, starea sistemului, valori de sistem și înregistratori de șir. Mai este inclusă, de asemenea, o listă de programe de ieșire din Control funcționare.

Detectoare de cod de referință sistem (SRC) IPL

Utilizați detectorul SRC IPL pentru a găsi informații despre mesajele SRC care sunt afișate pe serverul dumneavoastră când realizați un IPL. SRC-urile indică starea IPL-ului și sunt deseori utile în analiza problemelor. Puteți căuta un SRC după nume, sau puteți afișa o listă a celor mai comune SRC-uri.

Depanare

Folosiți informația din acest subiect pentru a vă ajuta să depanați câteva din cele mai obișnuite probleme care apar în gestionarea lucrului.

Jobul meu este în impas

Acest subiect discută motive posibile pentru care un job ar putea fi suspendat.

Jobul așteaptă să obțină o blocare a unui obiect

Modul de diagnosticare:	Vedeți starea jobului din Navigatorul iSeries; vedeți Determinarea stării unui job. Un job care așteaptă să primească o blocare va avea o stare a <i>Așteaptă pentru blocare</i> .
Recuperare:	Vedeți lista obiectelor blocate pentru job pentru a determina pentru ce obiect așteaptă jobul să obțină o blocare. Atunci folosiți acțiunea Posesorii blocajului împotriva obiectului ca să determinați care joburi deja posedă blocajul. Atunci trebuie să determinați de ce acest job reține blocajul, și ce poate fi terminat pentru eliberarea blocajului.

Jobul este reținut

Modul de diagnosticare:	Vedeți starea unui job din Navigatorul iSeries; vedeți Determinarea stării unui job
Recuperare:	Faceți clic dreapta pe job și apoi clic pe Eliberare .

Următoarele sunt posibilele motive de ce un job dintr-o coadă de joburi ar putea fi în impas:

Coadă de joburi este reținută

Modul de diagnosticare:	Vedeți starea unei cozi de joburi din Navigatorul iSeries;
Recuperare:	<ol style="list-style-type: none">1. Mutarea jobului într-o coadă de joburi care nu este reținută, vedeți Mutarea joburilor în alte cozi de joburi.2. Eliberarea cozii de joburi. Pentru a face astfel, faceți clic dreapta pe job și faceți clic pe Eliberare.

Coadă de joburi nu a fost alocată de un subsistem activ

Modul de diagnosticare:	Vedeți starea cozii de joburi din Navigatorul iSeries.
Recuperare:	<ol style="list-style-type: none">1. Mutarea jobului într-o coadă de joburi care este alocată de un subsistem activ, vedeți Mutarea joburilor în alte cozi de joburi.2. Pornirea unui subsistem care conține o intrare în coada de joburi pentru această coadă de joburi, vedeți Pornirea unui subsistem.3. Adăugarea unei intrări în coada de joburi pe un subsistem activ folosind comanda de adăugare a intrării în coada de joburi, ADDJOBQE.

A fost atins maximul subsistemului

- Modul de diagnosticare: Vedeți valoarea maximului de joburi active pentru subsistem din Navigatorul iSeries. Pentru a face asta, faceți clic dreapta pe subsistem și clic pe **Proprietăți**.
- Recuperare:
1. Mutarea jobului într-o coadă de joburi diferită, vedeți Mutarea joburilor în alte cozi de joburi.
 2. Incrementarea valorii maxime. Pentru a face aceasta, utilizați comanda CHGSBSD (Change Subsystem Description - Modificare descriere subsistem).

A fost atins maximul cozii de joburi

- Modul de diagnosticare: Vedeți valoarea maximă de joburi active pentru coada de joburi din Navigatorul iSeries. Pentru a face astfel, faceți clic dreapta pe coada de joburi și apoi clic pe **Proprietăți**. Apoi selectați fișa **Activitate**.
- Recuperare:
1. Mutarea jobului într-o coadă de joburi diferită; vedeți Mutarea joburilor în cozi de joburi diferite.
 2. Incrementarea valorii maxime. Pentru a face astfel, utilizați comanda CHGJOBQE (Change Job Queue Entry - Modificare intrare coadă de joburi).

Valoarea maximă pentru nivelul de prioritate a fost atinsă

- Modul de diagnosticare: Determinarea priorității cozii de joburi ale jobului vizualizând prioritățile sale. Apoi vizualizați valorile maximului de joburi active, în funcție de prioritatea de job, pentru coada de joburi din Navigatorul iSeries. Pentru a face astfel, faceți clic dreapta pe coada de joburi și clic pe **Proprietăți**. Apoi selectați fișa Activitate și faceți clic pe butonul **Avansat**.
- Recuperare:
1. Mutarea jobului într-o coadă de joburi diferită; vedeți Mutarea joburilor în cozi de joburi diferite.
 2. Modificați prioritatea din coada de joburi a jobului; vedeți Specificarea priorității unui job într-o coadă de joburi.
 3. Incrementarea valorii maxime. Pentru a face astfel, utilizați comanda CHGJOBQE (Change Job Queue Entry - Modificare intrare în coada de joburi).

Jobul meu are o performanță scăzută

Informațiile din acest articol discută motive posibile pentru care un job poate avea o performanță redusă.

Memorie insuficientă

- Modul de diagnosticare: Vedeți proprietățile jobului care determină în care pool de memorie rulează jobul. Apoi vizualizați proprietățile pool-ului de memorie din Navigatorul iSeries, vedeți Verificarea utilizării pool-ului de memorie. O rată mare de pagini lipsă la un pool indică faptul că nu există memorie suficientă în pool sau că prea multe joburi sunt în acel pool care intră în competiție pentru memorie.
- Recuperare:
1. Porniți reglarea sistem dacă nu o folosiți deja. Valoarea sistem QPFRADJ ajustează automat pool-urile de memorie și nivelurile de activitate.
 2. Dacă este posibil, reglați manual pool-ul cu care lucrați prin creșterea cantității de memorie din pool sau prin reducerea nivelului de activitate pentru pool-ul de memorie. Este posibil, de asemenea, să doriți să verificați pool-ul mașinii pentru a verifica dacă cantitatea de memorie care este utilizată nu afectează toate joburile din sistem.

Nivel de activitate prea scăzut

- Modul de diagnosticare: Vedeți proprietățile jobului pentru a determina starea sa și în care pool de memorie rulează jobul. Dacă jobul arată o stare de *Așteptare pentru nivel de activitate*, atunci vizualizați proprietățile pool-ului de memorie din Navigatorul iSeries, vedeți Verificarea utilizării pool-ului de memorie. O rată mare de tranziții către starea ineligibilă din pool indică faptul că prea multe joburi din pool au intrat în competiție pentru memorie.
- Recuperare:
1. Porniți reglarea sistem dacă nu o folosiți deja. Valoarea sistem QPFRADJ ajustează automat pool-urile de memorie și nivelurile de activitate.
 2. Reglați manual pool-ul prin creșterea nivelului de activitate pentru pool-ul de memorie.

Resurse CPU insuficiente

- Modul de diagnosticare: Vizualizați coloana CPU % pentru job și alte joburi din lista Joburi active a Navigatorului iSeries. Dacă sistemul este foarte ocupat, jobul dumneavoastră este posibil să nu primească destule resurse CPU pentru a termina de lucru.
- Recuperare:
1. Dacă este posibil, terminați sau rețineți lucrul nenecesar din sistemul dumneavoastră.
 2. Dacă câteva joburi folosesc intens CPU, modificați prioritatea de rulare a acestor joburi (o valoare de prioritate de rulare mai mare echivalează cu o prioritate la rulare mai mică pentru job).

Opțiunea de paginare pool de memorie

- Modul de diagnosticare: Dacă o aplicație este disc intensivă, dacă CPU-ul este subutilizat și dacă există suficientă memorie, utilizarea cache-ului expert poate avea avantaje.
- Recuperare: Cache-ul expert poate fi pornit de către Navigatorul iSeries modificând opțiunea Paginare, pentru un pool de memorie partajat, cu Calculat. Opțiunea Paginare este localizată în fișa **Configurație** a paginii **Proprietăți** a pool-ului de memorie și este disponibilă doar în pool-urile partajate (nu pool-urile private).

Prioritate mică de rulare job

- Modul de diagnosticare: vedeți proprietățile jobului pentru a determina prioritatea la rulare a unui job față de alte joburi din sistem.
- Recuperare: Dacă jobul are o prioritate de rulare scăzută (un număr mai mare) față de alte joburi și nu folosește prea mult CPU din cauza joburilor cu prioritate mai mare (număr mai mic) care folosesc cel mai mult CPU, ar trebui să creșteți prioritatea de rulare a jobului, vedeți Proprietăți job. De asemenea, pe un sistem cu o utilizare ridicată a CPU și un job cu o prioritate de rulare mică, setarea valorilor de sistem Ajustarea dinamică a priorităților de job în benzile de prioritate (QDYNPTYSCD) și Ajustarea dinamică a priorităților de job a joburilor interactive (QDYNPTYADJ) poate fi utilă.

Pentru informații suplimentare despre performanță, vedeți Performanță. Dacă doriți informații suplimentare despre cum să reglați performanța în sistemul dumneavoastră, vedeți Reglare performanță.

Investigarea jobului prestart

Acest subiect furnizează pași care să vă ajute să răspundeți la întrebarea "Cum găsesc adevăratul utilizator al unui job prestart și cum determin resursele utilizate de către acel job prestart?"

Navigator iSeries

Puteți utiliza vizualizările de Control funcționare sau monitoarele de Administrare centrală ale Navigatorului iSeries pentru o analiză în timp real a ceea ce se întâmplă pe sistemul dumneavoastră.

1. Utilizați vizualizarea Joburi de server pentru a vedea joburile de server active și utilizatorul curent. (**Conexiunile mele** → **server** → **Control funcționare** → **Joburi de server**)

- Faceți clic dreapta pe containerul Joburi de server și selectați **Personalizarea acestei vizualizări** → **Coloane** și asigurați-vă că utilizatorul curent, Timpul total al unității centrale de procesare și Timpul total DB al unității centrale sunt în lista **Coloane care vor fi afișate**.
- Dacă lista dumneavoastră de joburi de server active este mare, puteți limita ceea ce este afișat după un nume de job, utilizator curent sau stare. Faceți clic dreapta pe containerul Joburi de server și faceți clic pe **Personalizarea acestei vizualizări** → **Includere**.
- Puteți sorta ordinea de afișare a listei de joburi de server active făcând clic pe anteturile coloanelor.
- Puteți seta ecranul să se reîmprospăteze automat la intervale regulate. (**Conexiunile mele** → **Faceți clic dreapta pe server** → **Personalizarea acestei vizualizări** → **Auto reîmprospătare**) Aceasta vă poate ajuta să observați orice modificări imediate.

O dată ce ați localizat jobul care vă interesa, puteți face clic dreapta pe job și accesa stiva de apeluri a jobului, istoricul jobului, statisticile performanței anterioare, ultima declarație SQL și proprietățile jobului.

2. Utilizați Administrarea centrală pentru a seta un monitor de sistem care va monitoriza utilizarea generală a CPU (Unitatea centrală de procesare). (**Administrare centrală** → **Monitoare** → **Faceți clic dreapta pe Sistem** → **Monitor nou**)
 - În timp ce monitorul rulează puteți face clic pe unul din puncte pentru a vedea următorul nivel de detalii. De exemplu, în timpul monitorizării Utilizării CPU puteți afișa o listă de joburi care au gradul cel mai mare de utilizare a CPU. Puteți, apoi, să faceți clic pe un job care utilizează multă CPU și să faceți clic pe **Proprietăți** pentru a afișa proprietățile jobului. (Vedeți ajutorul online pentru informații suplimentare despre cum să utilizați monitorul de sistem.)
3. Utilizați Administrare centrală pentru a seta un monitor de job să monitorizeze joburi de server corespunzătoare și să vă notifice când aceste joburi încep să consume resurse excesive. (**Administrare centrală** → **Monitoare** → **Faceți clic dreapta pe Job** → **Monitor nou**)

| Interfața bazată pe caractere

| **Comandă:** Gestionare joburi active (WRKACTJOB)

| Această comandă afișează utilizatorul curent al firului de execuție inițial (care este chiar jobul în cazul în care acesta are doar un fir de execuție). Acestea sunt aceleași date care sunt arătate în GUI.

Informații înrudite pentru gestiunea lucrului

Control funcționare este complex. El este întretesut cu multe alte componente. Următoarea este o listă de subiecte care se găsesc în Centrul de informare IBM iSeries , Versiunea 5 Ediția 4 (V5R4). Aceste subiecte vă pot ajuta să vă descurcați cu părțile sistemului dumneavoastră care sunt legate de, dar înafara arenei Controlului de funcționare.

(**Centrul de informare IBM iSeries, Versiunea 5 ediția 4 (V5R4)** → **Gestiune sistem** → **Control funcționare** → **Informații legate de Controlul de funcționare**)

Lucrul cu rețelele

Înțelegerea tehnologiilor de rețea este o parte vitală a soluției totale de e-business a companiei dumneavoastră. Învățați cum să conectați afacerea dumneavoastră la Internet, cum să configurați e-mail-ul și cum să serviți obiecte multimedia clienților browser-ului dumneavoastră Web. Puteți integra servicii de fișiere și imprimantă, gestiunea profilului utilizator și operații de rețea. Găsiți informații despre serverul Windows(R) care poate fi integrat în server și citiți despre ofertele de securitate care vă pot ajuta să vă protejați resursele.

Extragere atribute rețea (API)

API-ul Retrieve Network Attributes (QWCRNETA) vă permite să extrageți atribute de rețea.

Reglarea performanței

Înțelegerea tuturor proceselor diferite care afectează performanța sistemului poate fi provocatoare pentru un utilizator neexperimentat. Rezolvarea problemelor de performanță necesită utilizarea efectivă a unei suite mari de unelte, fiecare cu propriul set unic de cerințe și funcții suportate. Chiar și după ce ați adunat și analizat

datele de performanță, poate fi complicat de știut ce anume trebuie făcut cu acele informații. poate fi complicat de știut ce anume trebuie făcut cu acele informații. Acest subiect vă va ghida prin operațiile și uneltele asociate cu gestionarea performanței.

Explorator de performanță

Exploratorul de performanță este o colecție de date care ajută utilizatorul să identifice cauzele problemelor de performanță care nu pot fi identificate prin colectarea datelor utilizând Serviciile de colecție sau prin realizarea unei analize generale.

Gestionare timp

Gestionarea timpului vă permite să gestionați timpul pe serverul dumneavoastră IBM eServer™ i5. În componenta de gestionare timp a Navigatorului iSeries™ Navigator puteți lucra cu funcțiile de fus orar și de reglare timp.

Valorile de sistem5/OS

Valorile sistemului sunt piese de informații care afectează mediul de operare în întregul sistem iSeries™. Valorile de sistem nu sunt obiecte din sistem. Mai degrabă, valorile sistemului conțin informații de control pentru funcționarea anumitor părți ale sistemului.

Anexa. Observații

Aceste informații au fost elaborate pentru produse și servicii oferite în S.U.A.

Este posibil ca IBM să nu ofere în alte țări produsele, serviciile sau caracteristicile discutate în acest document. Luați legătura cu reprezentantul IBM local pentru informații despre produsele și serviciile disponibile în zona dumneavoastră. Referirea la un produs, program sau serviciu IBM nu înseamnă că se afirmă sau că se sugerează faptul că poate fi folosit numai acel produs, program sau serviciu IBM. Poate fi folosit în loc orice produs, program sau serviciu care este echivalent din punct de vedere funcțional și care nu încalcă dreptul de proprietate intelectuală al IBM. Însă evaluarea și verificarea modului în care funcționează un produs, program sau serviciu non-IBM ține de responsabilitatea utilizatorului.

IBM poate avea brevete sau aplicații în curs de brevetare care să acopere subiectele descrise în acest document. Furnizarea acestui document nu vă acordă nici o licență asupra acestor patente. Puteți trimite întrebări cu privire la licențe, în scris, la:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pentru întrebări privind licența pentru informațiile DBCS (double-byte), contactați departamentul de Proprietate intelectuală IBM din țara dumneavoastră sau trimiteți întrebările în scris la:

IBM World Trade Asia Corporation
Licențe
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

Următorul paragraf nu se aplică în cazul Marii Britanii sau al altor țări unde asemenea prevederi sunt incompatibile cu legile locale: INTERNATIONAL BUSINESS MACHINES CORPORATION OFERĂ ACEASTĂ PUBLICAȚIE “CA ATARE”, FĂRĂ NICI UN FEL DE GARANȚIE EXPLICITĂ SAU IMPLICITĂ, INCLUZÂND, DAR FĂRĂ A SE LIMITA LA GARANȚIILE IMPLICITE DE NEÎNCĂLCARE A UNOR DREPTURI SAU NORME, DE VANDABILITATE SAU DE POTRIVIRE CU UN ANUMIT SCOP. Unele state nu permit declinarea responsabilității pentru garanțiile exprese sau implicite în anumite tranzacții și de aceea este posibil ca aceste clauze să nu fie valabile în cazul dumneavoastră.

Aceste informații pot conține greșeli tehnice sau erori de tipar. Se efectuează modificări periodice la informațiile incluse aici; aceste modificări vor fi încorporate în noi ediții ale publicației. IBM poate aduce îmbunătățiri și/sau modificări produsului (produselor) și/sau programului (programele) descrise în această publicație în orice moment, fără notificare.

Orice fel de referințe din aceste informații către situri Web non-IBM sunt furnizate doar pentru conveniență și nu servește în nici un caz ca aprobare a acelor situri Web. Materialele de pe siturile Web respective nu fac parte din materialele pentru acest produs IBM, iar utilizarea acestor situri Web se face pe propriul risc.

IBM poate utiliza sau distribui oricare dintre informațiile pe care le furnizați, în orice mod considerat adecvat, fără ca aceasta să implice vreo obligație față de dumneavoastră.

Posesorii de licențe pentru acest program care doresc să obțină informații despre el în scopul de a permite: (I) schimbul de informații între programe create independent și alte programe (inclusiv acesta) și (II) utilizarea mutuală a informațiilor care au fost schimbate, vor contacta:

IBM Corporation

Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Aceste informații pot fi disponibile cu condiția respectării termenilor și condițiilor, iar în unele cazuri cu plata unor taxe.

- | Programul cu licență descris în aceste informații și toate materialele cu licență disponibile pentru acesta sunt furnizate
- | de către IBM conform termenilor din Contractul IBM cu Clientul, Contractul de Licență IBM pentru Programele
- | Internaționale, Contractul de Licență IBM pentru Codul Mașină, sau orice contract echivalent dintre noi.

Toate datele de performanță conținute aici au fost determinate într-un mediu controlat. Prin urmare, rezultatele obținute în alte medii de operare pot varia semnificativ. Este posibil ca unele măsurători să fi fost realizate pe sisteme de nivel evoluat și nu există nici o garanție că aceste măsurători vor fi identice pe sisteme general disponibile. Mai mult, este posibil ca anumite măsurători să fi fost estimate prin extrapolare. Rezultatele reale pot varia. Utilizatorii acestui document ar trebui să verifice datele aplicabile pentru mediul lor specific.

Informațiile în legătură cu produsele non-IBM au fost obținute de la furnizorii acelor produse, din anunțurile publicate de aceștia sau din alte surse publice disponibile. IBM nu a testat acele produse și nu poate confirma acuratețea performanței, compatibilitatea sau orice alte pretenții legate de produse non-IBM. Întrebările privind capacitățile produselor non-IBM se pot adresa furnizorilor acelor produse.

Toate declarațiile privind orientarea viitoare sau intențiile IBM sunt supuse modificării sau retractării fără o înștiințare prealabilă și reprezintă doar ținte și obiective.

Aceste informații conțin exemple de date și rapoarte folosite în operații de afaceri zilnice. Pentru a fi cât mai complete, exemplele includ nume de persoane, de companii, de mărci și de produse. Toate aceste nume sunt fictive și orice asemănare cu nume și adrese utilizate de o întreprindere reală este pur întâmplătoare.

LICENȚĂ DE COPYRIGHT:

Aceste informații cuprind exemple de programe de aplicație în limbaj sursă, care ilustrează tehnici de programare pe diverse platforme de operare. Puteți copia, modifica și distribui aceste programe-exemplu în orice formă fără necesitatea unei plăți către IBM, în scopul dezvoltării, utilizării, marketingului sau distribuirii programelor de aplicație în concordanță cu interfața de programare a aplicației pentru platforma de operare pentru care sunt scrise programele-exemplu. Aceste exemple nu au fost testate complet în toate condițiile. Prin urmare, IBM nu poate garanta sau sugera că aceste programe vor fi fiabile, practice sau funcționale.

Fiecare copie sau orice porțiune din aceste exemple de program sau orice lucrare derivată din acestea trebuie să includă un anunț de copyright de genul următor:

© (numele companiei dumneavoastră) (anul). Porțiuni din acest cod sunt derivate din Programe exemplu ale IBM Corp.
© Copyright IBM Corp. _introduceți anul sau anii_. Toate drepturile rezervate.

Această publicație Control funcționare certifică Interfețele de programare proiectate care permit clientului să scrie programe pentru a obține serviciile IBM i5/OS.

Dacă vedeți aceste informații folosind o copie electronică, fotografiile și ilustrațiile color s-ar putea să nu apară.

Mărci comerciale

Următorii termeni sunt mărci comerciale deținute de International Business Machines Corporation în Statele Unite, în alte țări sau ambele:

- | eServer
- | i5/OS
- | IBM
- | iSeries
- | System i5

- | Intel, Intel Inside (logo-uri), MMX și Pentium sunt mărci comerciale ale Intel Corporation în Statele Unite, în alte țări sau în ambele.

Microsoft, Windows, Windows NT și logo-ul Windows sunt mărci înregistrate deținute de Microsoft Corporation în Statele Unite, în alte țări sau ambele.

Java și toate mărcile comerciale bazate pe Java sunt mărci comerciale deținute de Sun Microsystems, Inc. în Statele Unite, în alte țări sau ambele.

- | Linux este marcă comercială a Linus Torvalds în Statele Unite, în alte țări sau ambele.

UNIX este o marcă comercială înregistrată deținută de The Open Group în Statele Unite și în alte țări.

Alte nume de companii, de produse și de servicii pot fi mărci comerciale sau mărci de serviciu ale altora.

Termenii și condițiile

Permisunile pentru utilizarea acestor publicații sunt acordate în conformitate cu următorii termeni și condiții.

Utilizare personală: Puteți reproduce aceste publicații pentru utilizarea personală, necomercială, cu condiția ca toate anunțurile de proprietate să fie păstrate. Nu puteți distribui, afișa sau realiza lucrări derivate din aceste publicații sau dintr-o porțiune a lor fără consimțământul explicit al IBM.

Utilizare comercială: Puteți reproduce, distribui și afișa aceste publicații doar în cadrul întreprinderii dumneavoastră, cu condiția ca toate anunțurile de proprietate să fie păstrate. Nu puteți să realizați lucrări derivate din aceste informații, nici să reproduceți, să distribuiți sau să afișați aceste informații sau o porțiune a lor în afara întreprinderii dumneavoastră fără consimțământul explicit de la IBM.

În afara celor acordate expres prin această permisiune, nu se acordă nici o altă permisiune, licență sau drept, explicite sau implicite, pentru aceste publicații sau orice informații, date, software sau alte elemente pe care le conțin și care reprezintă o proprietate intelectuală.

IBM își păstrează dreptul de a retrage permisiunile acordate aici oricând, la discreția sa, dacă folosirea publicațiilor este în detrimentul intereselor sale sau, după cum este determinat de IBM sau dacă instrucțiunile de mai sus nu sunt urmate corespunzător.

Nu puteți descărca, exporta sau reexporta aceste informații decât în deplină conformitate cu legile și regulamentele aplicabile, inclusiv toate legile și regulamentele de export ale Statelor Unite.

IBM NU OFERĂ GARANȚII DESPRE CONȚINUTUL ACESTOR PUBLICAȚII. PUBLICAȚIILE SUNT FURNIZATE "CA ATARE", FĂRĂ NICI UN FEL DE GARANȚIE, EXPLICITĂ SAU IMPLICITĂ, INCLUZÂND, DAR FĂRĂ A SE LIMITA LA ELE, GARANȚIILE IMPLICITE DE VANDABILITATE, DE NEÎCĂLCARE A UNOR DREPTURI SAU NORME ȘI DE POTRIVIRE PENTRU UN ANUMIT SCOP.



Tipărit în S.U.A.