



Systemy IBM - iSeries

Manažment systémov - Klastre

*Verzia 5, vydanie 4*







Systemy IBM - iSeries

Manažment systémov - Klastre

*Verzia 5, vydanie 4*

**Poznámka**

Pred použitím týchto informácií a produktu, ktorého sa týkajú, si prečítajte informácie v časti “Právne informácie”, na strane 149.

**Siedme vydanie (február 2006)**

Toto vydanie sa týka verzie 5, vydania 4, modifikácie 0 produktu IBM i5/OS (číslo produktu 5722-SS1) a všetkých nasledujúcich vydaní a modifikácií, ak nie je v nových vydaniach uvedené inak. Táto verzia nie je určená pre všetky modely RISC (reduced instruction set computer) ani pre všetky modely CISC.

© Copyright International Business Machines Corporation 1998, 2006. Všetky práva vyhradené.

# Obsah

<b>Klastre . . . . .</b>	<b>1</b>		Monitorovanie administratívnej domény klastra. . . . .	108
Novinky vo V5R4 . . . . .	1		Monitorovanie stavu klastra . . . . .	109
Vytlačiteľné PDF . . . . .	2		Výkon klastra . . . . .	110
Koncepty klastrov . . . . .	2		Ukončenie klastrových úloh . . . . .	111
Výhody klastrov. . . . .	3		Monitorovanie a riadenie prostriedkov (RMC) . . . . .	111
Ako pracuje klaster . . . . .	3		Úloha štruktúry a užívateľské fronty . . . . .	112
Základy klastra . . . . .	4		Udržiavanie užívateľských profilov vo všetkých uzloch . . . . .	114
Časti klastra . . . . .	7		Zálohovanie a obnova klastrov. . . . .	114
Udalosti klastra . . . . .	17		Uloženie konfigurácií klastra . . . . .	115
Klastrové aplikácie . . . . .	28		Príklad: Konfigurácie klastra . . . . .	115
Plán pre klaster. . . . .	71		Príklad: Jednoduchý dvojuzlový klaster . . . . .	115
Riešenia konfigurácie a správy klastrov . . . . .	72		Príklad: Štvoruzlový klaster . . . . .	116
Požiadavky klastra. . . . .	79		Príklad: Klaster s prepínateľným diskom s použitím nezávislých diskových oblastí . . . . .	118
Návrh klastra . . . . .	81		Príklad: Administratívna doména klastra na manažovanie partnerských prostriedkov . . . . .	119
Bezpečnosť klastra. . . . .	88		Príklad: Nezávislé diskové oblasti s geografickým zrkadlením . . . . .	120
Zoznam klastrovej konfigurácie. . . . .	90		Riešenie problémov s klastrami . . . . .	120
Server INETD . . . . .	93		Identifikácia problémov s klastrami . . . . .	121
Laditeľné parametre komunikácií klastra . . . . .	94		Získanie informácií pre obnovu klastra . . . . .	122
Kontrolný zoznam pre zrušenie konfigurácie klastra. . . . .	95		Zisťovanie problému pomocou príkazu DMPCLUTRC (Dump Cluster Trace) . . . . .	122
Plán pre administratívnu doménu klastra . . . . .	96		Zisťovanie problému pomocou makra CLUSTERINFO . . . . .	126
Konfigurácia klastrov. . . . .	97		Bežné problémy s klastrom. . . . .	132
Vytvorenie klastra . . . . .	97		Chyby oddielov . . . . .	134
Riadenie klastrov . . . . .	98		Obnova klastra . . . . .	138
Pridanie uzla do klastra . . . . .	99		Často kladené otázky o manažovaní klastrov v Navigátore iSeries . . . . .	141
Spustenie klastrového uzla . . . . .	100		Kam volať pre podporu klastrov . . . . .	147
Ukončenie klastrového uzla . . . . .	101		Súvisiace informácie pre klastre . . . . .	147
Prispôbenie klastrovej verzie pre klaster . . . . .	101			
Vymazanie klastra . . . . .	102			
Vytvorenie CRG . . . . .	102			
Spustenie CRG . . . . .	103			
Zmena domény obnovy pre skupinu klastrových prostriedkov . . . . .	104			
Vykonanie prepnutia. . . . .	104			
Pridanie uzla do domény pre zariadenie . . . . .	105			
Odstránenie uzla z domény pre zariadenie . . . . .	106			
Ako systémová udalosť ovplyvňuje klaster . . . . .	107			
Vytvorenie administratívnej domény klastra . . . . .	107			
Pridanie položiek monitorovaných prostriedkov . . . . .	107			
			<b>Príloha. Právne informácie. . . . .</b>	<b>149</b>
			Informácie o programovom rozhraní . . . . .	150
			Ochranné známky . . . . .	151
			Pojmy a podmienky . . . . .	151



---

## Klastre

Klastre vám umožňujú efektívne zoskupiť systémy iSeries a vytvoriť tak prostredie, ktoré poskytuje vašim kritickým aplikáciám, zariadeniam a údajom dostupnosť blížiacu sa k hodnote 100 percent.

Klastre vám tiež poskytujú zjednodušenie riadenia systému a zvýšenie škálovateľnosti, takže môžete bez prerušenia pridávať nové komponenty, ak vaše podnikanie rastie.

Použitím príkladov kódu súhlasíte s podmienkami v časti Informácie o licencií na kód a právne vyhlásenia.

---

### Novinky vo V5R4

Dozviete sa, čo je nové v tomto vydaní.

#### Podpora administratívnych domén klastra

*Administratívna doména klastra* monitoruje a synchronizuje zmeny vybraných prostriedkov v klastri. Administratívna doména klastra zabezpečuje jednoduchšiu správu a synchronizáciu atribútov prostriedkov, ktoré sa zdieľajú v klastri, napríklad premenných prostredia alebo užívateľských profilov. Viac informácií o administratívnej doméne klastra nájdete v týchto témach:

- “Administratívna doména klastra” na strane 8
- “Plán pre administratívnu doménu klastra” na strane 96
- “Kontrolný zoznam pre administratívnu doménu klastra” na strane 96
- “Vytvorenie administratívnej domény klastra” na strane 107

#### Podpora skupiny prostriedkov klastra (CRG) s partnerskými uzlami

Všetky rozhrania CRG boli vylepšené, aby podporovali skupinu prostriedkov klastra (CRG) s partnerskými uzlami. *Skupina prostriedkov klastra (CRG) s partnerskými uzlami* je neprepínateľné CRG, v ktorom má každý uzol domény obnovy rovnocennú rolu v obnove prostriedkov, priradených k danému CRG. Viac informácií nájdete v týchto témach:

- Skupina prostriedkov klastra
- “Vytvorenie CRG” na strane 102
- “Spustenie CRG” na strane 103



#### Vylepšenia klastra

Bolo vykonaných viacero vylepšení, ktoré zdokonaľujú voľby pre vypnutie systému a riešenie problémov v prostredí s klastrami. Tieto vylepšenia zahŕňajú:

- Systematický prístup k ukončeniu klastrovania v uzle klastra, keď sa ukončia všetky podsystémy, alebo keď sa ukončí alebo vypne systém. Detaily nájdete v téme “Ako systémová udalosť ovplyvňuje klastri” na strane 107.
- Možnosť nakonfigurovať nové CRG aplikácie s aktívnou adresou IP pre prevzatie. Viac informácií nájdete v téme “Vytvorenie CRG aplikácie s aktívnou adresou IP pre prevzatie” na strane 103.
- Možnosť zobraziť celý klastri a priradené skupiny CRG z aktívneho uzla a odstrániť tak problémy s klastrom. Viac detailov nájdete v téme “Získanie informácií pre obnovu klastra” na strane 122.
- Boli pridané nové informácie o ladiacich nástrojoch a výsledky, ktoré generujú. Pomocou týchto nástrojov a ich výsledkov môžete určiť riešenie problému v klastri. Viac detailov nájdete v týchto témach:
  - “Zisťovanie problému pomocou príkazu DMPCLUTRC (Dump Cluster Trace)” na strane 122
  - “Zisťovanie problému pomocou makra CLUSTERINFO” na strane 126

## I Ako zistiť, čo je nové alebo zmenené

I Na označenie technických zmien používajú tieto informácie:

- I • Obrázok  na označenie začiatku nových alebo zmenených informácií.
- I • Obrázok  na označenie konca nových alebo zmenených informácií.

I Ak chcete nájsť ďalšie informácie o novinkách alebo zmenách v tomto vydaní, pozrite si časť Poznámka pre užívateľov.




---

## Vytlačiteľné PDF

Pomocou tejto témy môžete zobraziť a vytlačiť tieto informácie vo verzii PDF.

Ak chcete zobraziť alebo prevziať tento dokument vo verzii PDF, vyberte odkaz Klastre (približne 938 KB).

## Dokumenty Redbook

- Clustering and IASPs for Higher Availability  (približne 6.4 MB) Tento dokument Redbook obsahuje prehľad technológie klastrov a prepinateľných diskov, dostupnej v serveroch iSeries.
- iSeries Independent ASPs: A Guide to Moving Applications to IASPs  (približne 3.4 MB) Tento dokument Redbook postupne predstavuje nezávislé oblasti ASP v serveroch iSeries.
- Roadmap to Availability on the iSeries 4000  (približne 626 KB) Tento dokument Redpaper postupne predstavuje nezávislé oblasti ASP v serveroch iSeries.

## Webové lokality


- High Availability and Clusters  ([www.ibm.com/servers/eserver/series/ha](http://www.ibm.com/servers/eserver/series/ha))  
Lokalita IBM pre vysokú dostupnosť a klastre

## Uloženie súborov PDF

Ak chcete uložiť PDF vo vašej pracovnej stanici za účelom zobrazenia alebo tlače:

1. Pravým tlačidlom myši kliknite na PDF vo vašom prehliadači (kliknite pravým tlačidlom myši na odkaz hore)
2. Ak používate prehliadač Internet Explorer, kliknite na **Save Target As**. Ak používate prehliadač Netscape Communicator, kliknite na **Save Link As**.
3. Prejdite do adresára, do ktorého chcete uložiť PDF.
4. Kliknite na **Save**.

## Prevzatie programu Adobe Acrobat Reader

Na zobrazenie alebo tlač týchto dokumentov PDF potrebujete Adobe Acrobat Reader. Kópiu tohto programu môžete prevziať z webovej lokality spoločnosti Adobe ([www.adobe.com/products/acrobat/readstep.html](http://www.adobe.com/products/acrobat/readstep.html)) .

---

## Koncepty klastrov

Naučte sa plne chápať, ako klastre fungujú. Prečítajte si o výhodách klastrov a o tom, v čom môžu byť pre vás dôležité, ako aj o dôležitých konceptoch klastrov.

Klaster iSeries je kolekcia alebo skupina jedného alebo viacerých systémov alebo logických oddielov, ktoré spolu fungujú ako jeden systém. Systémy v klastri, nazývané uzly klastra, navzájom spolupracujú a poskytujú jedno výpočtové riešenie. Klastrovanie iSeries podporuje v jednom klastri až 128 uzlov. To vám umožňuje efektívne zoskupiť systémy iSeries a vytvorí tak prostredie, ktoré poskytuje vašim kritickým aplikáciám a údajom dostupnosť blížiacu sa k hodnote 100 percent. To vám pomôže zabezpečiť, že vaše kritické systémy a aplikácie budú dostupné 24 hodín denne, 7



dni v týždni. Klastre vám tiež poskytujú zjednodušenie riadenia systému a zvýšenie škálovateľnosti, takže môžete bez prerušenia pridávať nové komponenty, ak vaše podnikanie rastie.

## Výhody klastrov

Klastre poskytujú riešenie, ak vaše podnikanie vyžaduje funkčné systémy 24 hodín denne, 7 dní v týždni.

- | Použitím klastrovania môžete výrazne znížiť počet a trvanie neplánovaných výpadkov ako aj trvanie plánovaných
- | výpadkov, čím zabezpečíte, že vaše systémy, údaje a aplikácie budú nepretržite dostupné.

Najvýznamnejšie výhody, ktoré vášmu podnikaniu klastre prinášajú, sú:

### Nepretržitá dostupnosť

Klastre zabezpečia, že vaše systémy, údaje a aplikácie zostanú nepretržite dostupné.

### Zjednodušená správa

- | Skupinu systémov môžete manažovať ako jediný systém alebo jedinú databázu, bez nutnosti prihlasovať sa k
- | samostatným systémom. Pomocou administratívnej domény klastra môžete jednoducho manažovať
- | prostriedky, ktoré sa zdieľajú v rámci klastra.

### Zvýšená rozšíriteľnosť

Pridávajte hladko jednotlivé komponenty tak, ako si to vyžaduje rast vášho podnikania.

#### Súvisiace koncepty

“Zlyhanie” na strane 17

K *núdzovému prepnutiu* dochádza, keď server v klastru vykoná pri zlyhaní systému automatické prepnutie na jeden alebo viac záložných serverov.

#### Súvisiace úlohy

“Prepnutie” na strane 20

K *prepnutiu* dochádza, keď manuálne prepnete prístup k prostriedku z jedného servera na iný.

## Ako pracuje klaster

- | Infraštruktúra klastrov v systéme i5/OS, nazývaná služby klastrových prostriedkov, poskytuje obnoviteľnosť pre vaše
- | kritické prostriedky. Tieto prostriedky môžu zahŕňať údaje, aplikácie, zariadenia a iné prostriedky, ku ktorým pristupuje
- | viacero klientov.

- | Ak dôjde k výpadku systému alebo lokality, k funkciám dostupným v systéme v klastru je možné pristupovať cez iné
- | systémy, ktoré boli definované v klastru. K údajom je možné pristupovať podľa dvoch modelov: model s
- | primárnymi/záložnými uzlami a model s partnerskými uzlami. Viac detailov o skupinách prostriedkov klastrov (CRG),
- | ktoré môžete vytvoriť na základe týchto modelov, nájdete v téme Skupina prostriedkov klastra.

#### Súvisiace koncepty

“Zlyhanie” na strane 17

K *núdzovému prepnutiu* dochádza, keď server v klastru vykoná pri zlyhaní systému automatické prepnutie na jeden alebo viac záložných serverov.

“Replikácia” na strane 25

*Replikácia* vytvorí kópiu niečoho v reálnom čase. To znamená, že skopíruje objekty z jedného uzla v klastru na jeden alebo viac iných uzlov v klastru.

“Obnoviteľné zariadenia” na strane 15

*Obnoviteľné zariadenia* sú fyzické zdroje predstavované konfiguračným objektom, akým je napríklad opis zariadenia, ktoré je prístupné z viac, než jedného uzla v klastru.

“Obnoviteľné údaje” na strane 15

*Obnoviteľné údaje* sú údaje, ktoré sú replikované na viac ako jednom uzli klastra.

“Opätovné pripojenie” na strane 20

*Opätovné pripojenie* znamená, že uzol sa stane aktívnym členom klastra po tom, čo bol nezúčastneným členom.

“Porovnanie logickej replikácie, prepínaných diskov a zrkadlenia lokalít” na strane 85

Táto téma poskytuje prehľad rôznych technológií obnoviteľnosti údajov, ktoré môžete použiť v klastroch na zvýšenie dostupnosti.

### Súvisiace úlohy

“Prepnutie” na strane 20

K *prepnutiu* dochádza, keď manuálne prepnete prístup k prostriedku z jedného servera na iný.

## Základy klastra

Pred začatím navrhovania a prispôsobovania klastra vašim potrebám by ste mali pochopiť základné koncepty klastrovania.

S klastrom súvisia dva základné koncepty: uzly klastra a skupina prostriedkov klastra. *Uzol klastra* je systém alebo logický oddiel iSeries, ktorý je členom klastra. Pri vytváraní klastra určíte systémy a logické oddiely, ktoré chcete zahrnúť do klastra ako uzly. *Skupina prostriedkov klastra (CRG)* slúži ako riadiaci objekt pre kolekciu obnoviteľných prostriedkov. CRG môže obsahovať podmnožinu alebo všetky uzly v klastri. Klastri iSeries podporuje štyri typy skupín CRG: pre aplikáciu, údaje, zariadenie a partnera. Tieto typy skupín CRG majú dva spoločné prvky: doménu obnovy a ukončovací program.

*Doména obnovy* definuje rolu každého uzla v CRG. Keď vytvoríte CRG v klastri, objekt CRG sa vytvorí vo všetkých uzloch patriacich k doméne obnovy. Poskytnutý je však iba jeden systémový obraz objektu CRG, ku ktorému môžete pristupovať z ľubovoľného uzla v doméne obnovy CRG. To znamená, že ľubovoľné zmeny v CRG sa urobia vo všetkých uzloch v doméne obnovy.

*Ukončovací program* sa volá pre CRG počas udalostí, ktoré súvisia s klastrom. Jednou z takýchto udalostí je presun prístupového bodu z jedného uzla do iného.

Existujú dva modely skupín CRG, ktoré môžete vytvoriť v klastri: model s primárnymi/záložnými uzlami a model s partnerskými uzlami. V modeli s primárnymi/záložnými uzlami môžu byť uzly domény obnovy CRG definované takto:

- *Primárny uzol* je uzol klastra, ktorý predstavuje primárny prístupový bod k obnoviteľnému prostriedku klastra.
- *Záložný uzol* je uzol klastra, ktorý prevezme rolu primárneho prístupu, ak súčasný primárny uzol zlyhá alebo ak sa inicializuje manuálne prepnutie.
- *Replikovaný uzol* je uzol klastra, ktorý obsahuje kópie prostriedkov klastra, ale nemôže prevziať rolu primárneho ani záložného uzla.

V modeli s partnerskými uzlami definuje doména obnovy CRG partnerský vzťah medzi uzlami. Uzly v doméne obnovy CRG s partnerskými uzlami môžu byť definované takto:

- *Partnerský uzol* je uzol klastra, ktorý môže byť aktívnym prístupovým bodom k prostriedkom klastra.
- *Replikovaný uzol* je uzol klastra, ktorý obsahuje kópie prostriedkov klastra. Uzly, ktoré sú v CRG s partnerskými uzlami definované ako replikované, reprezentujú neaktívny prístupový bod k prostriedkom klastra.

V CRG s partnerskými uzlami sú uzly v doméne obnovy rovnocenné s ohľadom na rolu, ktorú zohrávajú v obnove. Keďže každý uzol v CRG s partnerskými uzlami má v podstate rovnakú rolu, koncepty prepnutia a núdzového prepnutia sa tu nepoužívajú. Uzly sú v partnerskom vzťahu a keď jeden z nich zlyhá, ostatné budú pokračovať v prevádzke.

Môžete tiež vytvoriť administratívnu doménu klastra, ktorú reprezentuje CRG s partnerskými uzlami. Uzly v administratívnej doméne klastra sú všetky partnerskými uzlami v doméne obnovy CRG. Nie sú tu žiadne replikované uzly.

V príklade nižšie je predstavené jedno CRG každého typu:



### CRG údajov

Uzol 1, Uzol 2 a Uzol 3 predstavuje CRG údajov. To znamená, že doména obnovy pre CRG údajov má zadanú rolu pre Uzol 1 (primárny), Uzol 2 (prvá záloha) a Uzol 3 (druhá záloha). V uvedenom príklade slúži Uzol 1 ako primárny bod prístupu. Uzol 2 je definovaný ako prvá záloha v doméne obnovy. To znamená, že Uzol 2 obsahuje kópiu prostriedku, ktorý sa pomocou logickej replikácie udržiava aktuálny. Ak dôjde k núdzovému prepnutiu, Uzol 2 sa stane primárnym prístupovým bodom.

### CRG aplikácie

Uzol 4 a uzol 5 predstavuje CRG aplikácie. To znamená, že doména obnovy pre CRG aplikácie má určený Uzol 4 a Uzol 5. V tomto príklade predstavuje Uzol 4 primárny prístupový bod. Ak dôjde k núdzovému prepnutiu, Uzol 5 sa stane primárnym prístupovým bodom k aplikácii. Vyžaduje náhradnú adresu IP.

### | CRG s partnerskými uzlami

| Uzol 6 a Uzol 7 predstavuje CRG s partnerskými uzlami. To znamená, že doména obnovy pre CRG s partnerskými uzlami má určený Uzol 6 a Uzol 7. V tomto príklade môžu byť uzly 6 a 7 partnerskými alebo replikovanými uzlami. Ak je toto administratívna doména klastra reprezentovaná pomocou CRG s partnerskými uzlami, prostriedky, ktoré monitoruje administratívna doména klastra budú synchronizovať všetky zmeny v doméne reprezentovanej Uzlom 6 a Uzlom 7, bez ohľadu na to, v ktorom uzle zmena nastala.

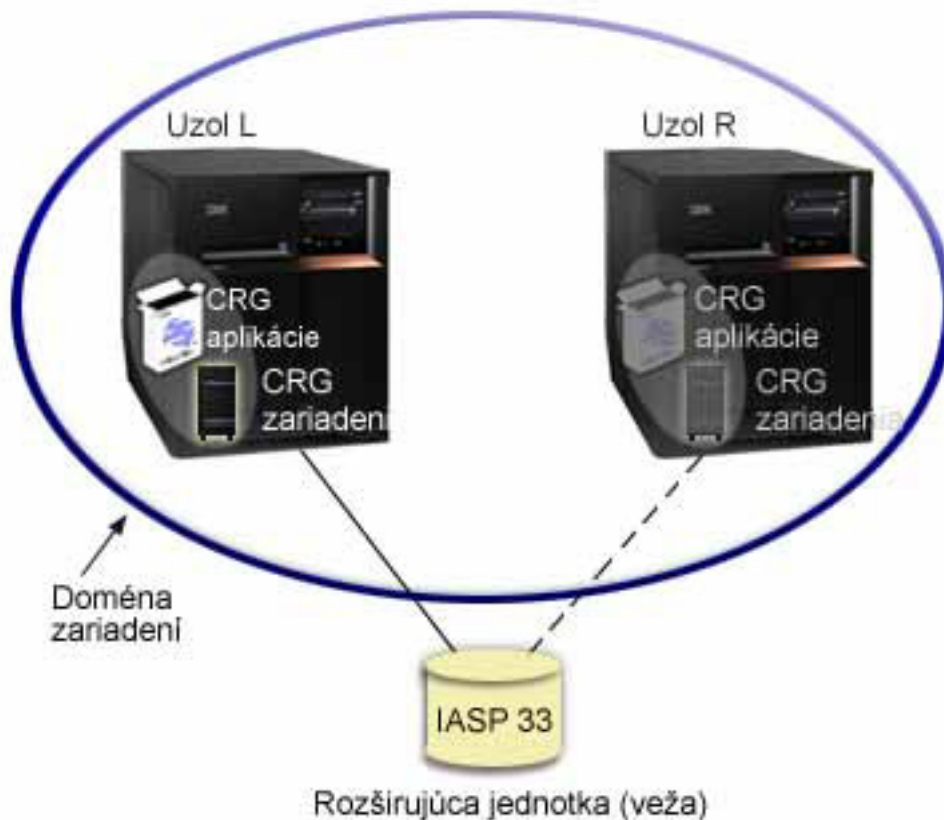
### CRG zariadenia

Uzol 2 a Uzol 3 predstavuje CRG zariadenia. To znamená, že doména obnovy pre CRG zariadenia má určený Uzol 2 a Uzol 3. V tomto príklade predstavuje Uzol 2 primárny prístupový bod. To znamená, že k

obnoviteľnému zariadeniu, ktoré vlastní CRG zariadenia, je možné aktuálne pristupovať z Uzla 2. Ak dôjde k núdzovému prepnutiu, Uzol 3 sa stane primárnym prístupovým bodom k zariadeniu.

CRG pre zariadenia vyžaduje odolné zariadenie nazývané nezávislá disková oblasť (tiež nazývané nezávislá pomocná pamäťová oblasť alebo nezávislé ASP), nakonfigurované na externé zariadenie, expanznú jednotku (vežu), alebo IOP v logickom oddiele.

Uzly v doméne obnovy zariadenia CRG musia taktiež patriť do tej istej domény zariadenia. Nasledujúci príklad predstavuje zariadenie CRG s Uzlom L a Uzlom R v ich doméne obnovy. Oba uzly taktiež patria do tej istej domény zariadenia.



### Súvisiace koncepty

“Uzol klastra” na strane 7

*Uzol klastra* je systém alebo logický oddiel iSeries, ktorý je členom klastra.

“Skupina klastrových prostriedkov” na strane 7

| *Skupina prostriedkov klastra (CRG)* je systémový objekt i5/OS predstavujúci množinu zoskupených prostriedkov  
| klastra, ktorá sa používa na manažovanie udalostí v prostredí s klastrami. Skupina prostriedkov klastra opisuje  
| doménu obnovy a poskytuje názov ukončovacieho programu pre skupinu prostriedkov klastra, ktorý sa zavolá pri  
| výskyte určitých udalostí klastra.

“Doména obnovy” na strane 11

| *Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za  
| spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

“Ukončovacie programy skupiny klastrových prostriedkov” na strane 10

*Ukončovací program pre skupinu prostriedkov klastra* sa zavolá, keď dôjde k udalosti pre CRG, ktorá súvisí s  
klastrom.

Nezávislé diskové oblasti

“Domény zariadení” na strane 16

*Doména zariadení* je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení. Konkrétne sa môžu uzly v doméne zariadení zúčastňovať akcie prepnutia pre určitú kolekciu obnoviteľných prostriedkov zariadení.

## Časti klastra

- | *Klaster* iSeries je kolekcia jedného alebo viacerých systémov alebo oddielov, ktoré spolu fungujú ako jeden systém.
- | Tieto informácie vás oboznámia s jednotlivými prvkami a ich vzájomnými vzťahmi.

### Uzol klastra

*Uzol klastra* je systém alebo logický oddiel iSeries, ktorý je členom klastra.

Každý uzol klastra je identifikovaný 8-znakovým názvom uzla klastra, ktorý je priradený k jednej alebo viacerým adresám IP, reprezentujúcim systém iSeries. Pri konfigurácii klastra môžete ako názov klastra použiť akýkoľvek názov. Odporúčame však použiť pre uzol rovnaký názov, ako je názov hostiteľa alebo systému.

Spojenia klastra využívajú zostavu protokolu TCP/IP na poskytovanie ciest spojenia medzi službami klastra na každom uzle klastra. Množina uzlov klastra, nakonfigurovaných ako súčasť klastra, sa označuje ako zoznam členov klastra.

### Skupina klastrových prostriedkov

- | *Skupina prostriedkov klastra (CRG)* je systémový objekt i5/OS predstavujúci množinu zoskupených prostriedkov klastra, ktorá sa používa na manažovanie udalostí v prostredí s klastrami. Skupina prostriedkov klastra opisuje doménu obnovy a poskytuje názov ukončovacieho programu pre skupinu prostriedkov klastra, ktorý sa zavolá pri výskyte určitých udalostí klastra.
- | Klaster poskytuje dve možnosti definovania vzťahov medzi uzlami v klastrí: model s primárnymi/záložnými uzlami a model s partnerskými uzlami. Každý z týchto modelov je možné použiť spolu alebo samostatne, podľa potrieb vášho prostredia.

### Model s primárnymi/záložnými uzlami

- | Všetky skupiny prostriedkov klastra tejto kategórie definujú v doméne obnovy uzly so špecifickými rolami: primárny, záložný alebo replikovaný uzol. Primárne a záložné uzly predstavujú dostupné prístupové body pre prostriedky klastra.
- | V danom čase bude však aktívnym prístupovým bodom len jeden uzol. Bude ním primárny uzol. Replikované uzly nemôžu byť prístupovými bodmi. To môžete zmeniť, ak priradíte replikovanému uzlu rolu záložného uzla. Skupiny prostriedkov klastra používajúce model s primárnymi/záložnými uzlami sa definujú ako skupiny s obnoviteľnými údajmi, obnoviteľnou aplikáciou alebo obnoviteľným zariadením. Obnoviteľnosť údajov umožňuje, aby boli viaceré kópie údajov udržiavané na viac ako jednom uzle v klastrí a umožňuje zmeniť prístupový bod na záložný uzol.
- | Obnoviteľnosť aplikácií umožňuje, aby bol aplikačný program reštartovaný na rovnakom, alebo na inom uzle klastra.
- | Obnoviteľnosť zariadení umožňuje, aby bolo zariadenie presunuté (prepnuté) na záložný uzol.

S každou skupinou klastrových prostriedkov údajov, alebo aplikácií, je spojený ukončovací program. Pre skupinu klastrových prostriedkov obnoviteľných zariadení je ukončovací program voliteľný.

V Navigátore iSeries sa skupiny prostriedkov klastra označujú odlišne.

- | • CRG zariadenia sa označuje ako **prepínateľné zariadenie**.
- | • CRG aplikácie sa označuje ako **prepínateľná aplikácia**.
- | • Na CRG údajov sa odkazuje ako na **skupinu prepínateľných údajov**.

### Model s partnerskými uzlami

- | Všetky skupiny prostriedkov klastra tejto kategórie definujú v doméne obnovy uzly s rolou partnerského alebo replikovaného uzla. Partnerské uzly môžu byť prístupovými bodmi pre skupinu prostriedkov klastra. Keď je skupina prostriedkov klastra spustená, všetky uzly definované ako partnerské budú prístupovými bodmi. Replikované uzly nemôžu byť prístupovými bodmi. To môžete zmeniť, ak priradíte replikovanému uzlu rolu partnerského uzla. V CRG s

- | partnerskými uzlami obsahuje každý uzol replikované údaje, ktoré sa nachádzajú vo všetkých uzloch. Keď v CRG s partnerskými uzlami zlyhá uzol, bod zlyhania sa oznámi ostatným uzlom a tie potom pokračujú v prevádzke od tohto bodu zlyhania.

Administratívnu doménu klastra reprezentuje CRG s partnerskými uzlami s doménou obnovy, pozostávajúcou len s partnerských uzlov.

#### **Súvisiace koncepty**

“Doména obnovy” na strane 11

- | *Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

“Ukončovacie programy skupiny klastrových prostriedkov” na strane 10

*Ukončovaci program pre skupinu prostriedkov klastra* sa zavolá, keď dôjde k udalosti pre CRG, ktorá súvisí s klastrom.

### **Riadenie spracovania skupín klastrových prostriedkov**

- | Keď dôjde k zlyhaniu uzla, vykoná sa núdzové prepnutie. Pri postupnom prepínaní skupín CRG sa najprv núdzovo prepnú skupiny CRG zariadení, potom skupiny CRG údajov a nakoniec skupiny CRG aplikácií. V prípade CRG s partnerskými uzlami sa nepoužíva poradie, ale pri zlyhaní sa upozorní každý uzol.

- | Ak si skontrolujete stav CRG, môžete si preveriť, že skupiny klastrových prostriedkov dokončili nápravu zlyhania alebo prepnutie.

Môžete tiež použiť blokovanie na udržanie aplikácie pokiaľ sú údaje dostupné na spracovanie. Pokiaľ sú spracovávané skupiny klastrových prostriedkov obnoviteľných údajov, môžete chcieť blokovať prístup k údajom, reprezentovaný údajmi CRG. Prístup môžete zablokovať pomocou API QxdaBlockEDRS (Block EDRS Access) a API QxdaCheckEDRSBlock (Check EDRS Block Status). Ak má dôjsť k núdzovému prepnutiu, pomocou týchto rozhraní API môžete zablokovať a odblokovať prístup z ukončovacieho programu pre skupinu prostriedkov klastra.

#### **Súvisiace koncepty**

“Zlyhanie” na strane 17

K *núdzovému prepnutiu* dochádza, keď server v klastru vykoná pri zlyhaní systému automatické prepnutie na jeden alebo viac záložných serverov.

“Doména obnovy” na strane 11

- | *Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

“Ukončovacie programy skupiny klastrových prostriedkov” na strane 10

*Ukončovaci program pre skupinu prostriedkov klastra* sa zavolá, keď dôjde k udalosti pre CRG, ktorá súvisí s klastrom.

#### **Súvisiace úlohy**

“Prepnutie” na strane 20

K *prepnutiu* dochádza, keď manuálne prepnete prístup k prostriedku z jedného servera na iný.

### **Administratívna doména klastra**

- | *Administratívna doména klastra* sa používa na manažovanie prostriedkov, ktoré je nutné v uzloch prostredia s klastrami udržiavať konzistentné.

- | V každom uzle, ktorý predstavuje prístupový bod k obnoviteľným údajom, aplikáciám a zariadeniam, môže byť nutné definovať určité prevádzkové alebo konfiguračné parametre. Ak sa vykoná zmena jedného z týchto parametrov v ľubovoľnom uzle, ktorý môže predstavovať prístupový bod pre niektorý prostriedok, túto zmenu je nutné rozšíriť do všetkých ostatných uzlov, ktoré môžu predstavovať potenciálny prístupový bod. Administratívna doména klastra poskytuje možnosť identifikovať prostriedky, ktoré je nutné v uzloch domény udržiavať konzistentné. Potom monitoruje zmeny týchto prostriedkov a synchronizuje ich v rámci aktívnej domény. Administratívnu doménu klastra reprezentuje CRG s partnerskými uzlami. Pri vytvorení administratívnej domény klastra vytvorí systém CRG s partnerskými uzlami. Názvom CRG s partnerskými uzlami sa stane názov administratívnej domény klastra. Uzly, ktoré tvoria administratívnu doménu klastra, definuje doména obnovy CRG s partnerskými uzlami. Všetky uzly sú partnerské uzly.



| V administratívnej doméne klastra nie sú povolené replikované uzly. V rámci klastra môže byť uzol definovaný len v jednej administratívnej doméne klastra. Viac informácií o úlohách súvisiacich s administratívnou doménou klastra nájdete v týchto témach:

- | 1. “Plán pre administratívnu doménu klastra” na strane 96
- | 2. “Kontrolný zoznam pre administratívnu doménu klastra” na strane 96
- | 3. “Vytvorenie administratívnej domény klastra” na strane 107
- | 4. “Pridanie položiek monitorovaných prostriedkov” na strane 107
- | 5. “Spustenie CRG” na strane 103

| Po vytvorení administratívnej domény klastra sa na jej manažovanie používajú normálne funkcie pre CRG. Ak chcete napríklad pridať uzol do administratívnej domény, musíte do domény obnovy CRG pridať uzol s rolou "partner". Ak chcete spustiť administratívnu doménu klastra, spustíte CRG s partnerskými uzlami.

| Proces synchronizácie zmien môžete riadiť spúšťaním a ukončovaním CRG. Keď ukončíte CRG, vykonané zmeny monitorovaného prostriedku v ľubovoľnom uzle sa nerozšíria do zvyšku domény. Keď spustíte skupinu CRG, zmeny monitorovaného prostriedku, ktoré sa vykonali počas jej neaktivity, sa rozšíria do zvyšku domény. Keď je CRG aktívne, vykonané zmeny ľubovoľného monitorovaného prostriedku v aktívom uzle sa dynamicky rozšíria, takže prostriedok zostane v rámci administratívnej domény konzistentný. Detaily nájdete v téme “Monitorovanie administratívnej domény klastra” na strane 108.

| Ak chcete pridať uzol do administratívnej domény klastra, musíte pridať uzol klastra do domény obnovy CRG s partnerskými uzlami. Keď pridáte uzol do domény, v novom uzle sa vytvoria všetky manažované prostriedky (ak ešte neexistujú) a zosynchronizujú sa so zvyškom administratívnej domény.

| Ak vymažete administratívnu doménu klastra, z každého uzla v doméne sa odstránia všetky prostriedky, ktoré sú v nej definované; skutočný prostriedok sa však zo systému neodstráni. Detaily nájdete v téme Monitorované prostriedky.

## | **Monitorované prostriedky**

| *Monitorované prostriedky* sú typy systémových prostriedkov, ktoré je možné manažovať pomocou administratívnej domény klastra. Tieto prostriedky sú v administratívnej doméne klastra reprezentované ako *položky monitorovaných prostriedkov (MRE)*.

| Prostriedky, ktoré sa synchronizujú pomocou administratívnej domény klastra, sú reprezentované položkami monitorovaných prostriedkov (MRE). Po pridaní MRE do administratívnej domény klastra sa budú vykonané zmeny prostriedku v ľubovoľnom uzle administratívnej domény klastra šíriť do všetkých uzlov v aktívnej doméne. Na manažovanie položiek MRE v administratívnej doméne klastra môžete použiť tri rozhrania API integrovaného prevádzkového prostredia:

- | • API QfpadAddMonitoredResourceEntry (Add Monitored Resource Entry)
- | • API QfpadRmvMonitoredResourceEntry (Remove Monitored Resource Entry)
- | • API QfpadRtvMonitoredResourceInfo (Retrieve Monitored Resource Information)

| Do administratívnej domény klastra môžete pridať položku monitorovaného prostriedku pre tieto typy prostriedkov:

- | • Systémové hodnoty
- | • Uživateľské profily
- | • Opisy úloh
- | • Trieda
- | • Opis zariadenia nezávislej diskovej oblasti
- | • Sieťové atribúty
- | • Systémové premenné prostredia
- | • Atribúty TCP/IP

MRE je možné pridať do administratívnej domény klastra, len ak sú všetky uzly v doméne aktívne a sú súčasťou skupiny. MRE nie je možné pridať, ak je administratívna doména klastra rozdelená. Keď pridáte MRE a CRG s partnerskými uzlami je spustené, zmeny prostriedku, ktorý reprezentuje MRE, sa budú šíriť do všetkých aktívnych uzlov v doméne. Ak je CRG ukončené, nevybavené zmeny sa rozšíria do aktívnej domény po jeho opätovnom spustení.

K MRE je priradený globálny stav. Keď má prostriedok reprezentovaný pomocou MRE rovnaké hodnoty všetkých monitorovaných atribútov vo všetkých uzloch aktívnej domény, globálny stav tohto prostriedku je konzistentný. Ak sa administratívna doména klastra pokúsi zaktualizovať prostriedok v jednom alebo viacerých uzloch a aktualizácia zlyhá, globálny stav prostriedku bude nekonzistentný. Keď je globálny stav nekonzistentný, administrátor musí zistiť príčinu zlyhania a odstrániť ju. Administratívna doména klastra sa pokúsi o obnovenie synchronizácie prostriedku pri jeho ďalšej aktualizácii (pravdepodobne keď administrátor zmení prostriedok v dôsledku opravy problému, ktorý spôsobil zlyhanie aktualizácie) alebo po reštartovaní CRG.

Keď je CRG administratívnej domény klastra ukončené, globálny stav všetkých položiek MRE sa nastaví na nekonzistentný. Je to preto, že v čase, keď je CRG ukončené, je možné vykonať zmeny monitorovaných prostriedkov v rôznych uzloch a tie sa môžu stať nekonzistentnými.

Ak prostriedok reprezentovaný pomocou MRE predstavuje systémový objekt, nemali by ste ho vymazať, premenovať ani presunúť do inej knižnice bez toho, aby ste najprv odstránili MRE. Ak sa prostriedok vymaže, premenuje alebo presunie do inej knižnice, globálny stav MRE bude nekonzistentný a zmeny prostriedku, ktoré sa potom vykonajú v ľubovoľnom uzle, sa nerozšíria do administratívnej domény klastra.

Keď sa do administratívnej domény klastra pridá uzol, všetky položky MRE z aktívnej domény sa skopirujú do nového uzla. Vytvorí sa všetky prostriedky reprezentované pomocou MRE, ktoré v novom uzle neexistujú a hodnoty atribútov sa nastaví na rovnaké hodnoty ako v aktívnej administratívnej doméne klastra.

Ak sa v administratívnej doméne klastra nachádzajú uzly, ktoré nie sú aktívne, všetky zmeny prostriedkov, vykonané v aktívnej doméne, sa do nich rozšíria, keď sa uzly znova pripoja do aktívnej domény. Ak je administratívna doména klastra rozložená, zmeny sa budú naďalej synchronizovať medzi aktívnymi uzlami v každom oddiele. Keď sa uzly znova zlúčia, administratívna doména klastra rozšíri všetky vykonané zmeny v každom klasteri, aby boli v aktívnej doméne prostriedky konzistentné. Ak sa v rôznych oddieloch vykonalo viacero zmien rovnakého prostriedku, po zlúčení oddielov sa v administratívnej doméne klastra spracuje každá zo zmien, ich poradie sa však nedá vopred určiť.

#### Súvisiace koncepty

“Administratívna doména klastra” na strane 8  
*Administratívna doména klastra* sa používa na manažovanie prostriedkov, ktoré je nutné v uzloch prostredia s klasterami udržiavať konzistentné.

## Ukončovacie programy skupiny klastrových prostriedkov

*Ukončovací program pre skupinu prostriedkov klastra* sa zavolá, keď dôjde k udalosti pre CRG, ktorá súvisí s klastrom.

Ukončovací program je pre obnoviteľné zariadenie CRG voliteľný, ale je povinný pre ostatné typy CRG. Keď sa používa ukončovací program pre skupinu prostriedkov klastra, volá sa pri výskyte celoklastrových udalostí, vrátane týchto:

- Uzol opustí klaster neočakávane.
- Uzol opustí klaster ako dôsledok API `QcstEndClusterNode` (End Cluster Node) alebo API `QcstRemoveClusterNodeEntry` (Remove Cluster Node Entry).
- Klaster sa vymaže ako dôsledok API `QcstDeleteCluster` (Delete Cluster).
- Uzol sa aktivuje pomocou API `QcstStartClusterNode` (Start Cluster Node).
- Je obnovená komunikácia s uzlom delenia.

Ukončovacie programy píše a poskytujú obchodní partneri IBM, ktorí tvoria softvér strednej úrovne pre klastre a poskytovatelia aplikačných programov podporujúcich klastre.



Podrobnejšie informácie o ukončovacích programoch skupín klastrových prostriedkov, vrátane toho, aké informácie sú ukončovaciemu programu odoslané pri každom z kódov akcií, nájdete v dokumentácii ku API klastrov v časti Ukončovací program skupiny klastrových prostriedkov.

## Doména obnovy

| Doména obnovy je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

Doména predstavuje tie uzly klastra, z ktorých je možný prístup na klastrový prostriedok. Táto podmnožina uzlov klastra, ktorá je priradená k určitej skupine prostriedkov klastra, podporuje primárny, sekundárny (záložný), replikovaný alebo partnerský prístupový bod.

Uzol môže mať v doméne obnovy tieto štyri typy rolí:

### Primárny

Klastrový uzol, ktorá je primárnym bodom prístupu pre obnoviteľný klastrový prostriedok.

- Pre CRG údajov obsahuje primárny uzol základnú kópiu prostriedku.
- Pre CRG aplikácie je primárnym uzlom systém, v ktorom sa aktuálne vykonáva aplikácia.
- Pre CRG zariadenia je primárny uzol aktuálnym vlastníkom prostriedku zariadenia.

| **Poznámka:** Pri použití geografického zrkadlenia, uzly v doméne obnovy CRG zariadenia vyžadujú názov lokality a adresy IP dátových portov. Detaily nájdete v téme Názov lokality a adresy IP údajových portov.

- Pre CRG s partnerskými uzlami nie je primárny uzol podporovaný.

| Ak primárny uzol pre CRG zlyhá alebo sa inicializuje manuálne prepnutie, primárny prístupový bod pre toto CRG sa presunie do prvého záložného uzla.

### Záložný

Klastrový uzol, ktorý prevezme úlohu primárneho prístupu v prípade zlyhania primárneho uzla, alebo inicializácie manuálneho prepnutia.

- Pre CRG údajov tento klastrový uzol obsahuje kópiu tých prostriedkov, ktoré sú udržiavané replikáciou.
- Pre CRG s partnerskými uzlami nie je záložný uzol podporovaný.

### Replikovať

| Klastrový uzol, ktorý má kópie klastrových prostriedkov, ale nie je schopný prevziať úlohu primárneho, alebo záložného uzla. Náprava zlyhania alebo prepnutie na replikačný uzol nie je možné. Ak chcete, aby sa z replikačného uzla stal primárny, musíte najprv zmeniť jeho úlohu z replikačného na záložný uzol.

- Pre skupiny CRG s partnerskými uzlami, reprezentujú uzly, definované ako replikované, neaktívne prístupové body k prostriedkom klastra.

### Partnerský

| Uzol klastra, ktorý nemá určené poradie a môže byť aktívnym prístupovým bodom k prostriedkom klastra. Keď je CRG spustené, všetky uzly definované ako partnerské budú aktívnymi prístupovými bodmi.

- V prípade CRG s partnerskými uzlami riadi prístupový bod riadiaca aplikácia, nie systém. Partnerská rola je podporovaná len v CRG s partnerskými uzlami.

## Model s primárnymi/záložnými uzlami

Pre uzly, ktoré sú súčasťou modelu s primárnymi/záložnými uzlami, má každý uzol v doméne obnovy priradenú rolu vzhľadom na aktuálne prevádzkové prostredie klastra. To sa nazýva *súčasná úloha* domény obnovy. Súčasná úloha sa mení v závislosti od operačných zmien, ktorými uzol prechádza, ako napr. spustenie a ukončenie práce uzla, zlyhanie uzla, atď. Každý uzol v doméne obnovy má ďalšie úlohy vzhľadom na preferované alebo ideálne prostredie klastra. To sa nazýva *preferovaná rola* domény obnovy. Preferovaná rola je statická definícia, ktorá je pôvodne nastavená pri vytvorení skupiny klastrových prostriedkov. Táto rola sa nemení pri zmenách prostredia klastra. Preferovaná rola je zmenená len ak sú uzly pridané, alebo odstránené z domény obnovy, alebo ak je uzol odstránená z klastra. Preferovanú rolu môžete tiež zmeniť ručne.

Prakticky môžete doménu obnovy v modeli s primárnymi/záložnými uzlami vnímať takto:

Tabuľka 1. Roly uzlov pre skupiny CRG s primárnymi/záložnými uzlami

Uzol	Aktuálna rola	Preferovaná rola
A	Záloha 1	Primárny
B	Záloha 2	Záloha 1
C	Primárny	Záloha 2
D	Replikovať	Replikovať

Uzly A, B, C a D tu poskytujú príklad skupiny CRG používajúcej model s primárnymi/záložnými uzlami. Uzol C slúži ako aktuálny primárny uzol. Jeho preferovaná rola je druhý záložný uzol, to znamená, že aktuálna rola Uzla C je dôsledkom dvoch akcií prepnutia alebo núdzového prepnutia. Pri prvom zotavení po zlyhaní alebo prepnutí bola úloha primárneho uzla presunutá z Uzla A na Uzol B, keďže tento bol určený ako prvá záloha. Druhé prepnutie alebo núdzové prepnutie spôsobilo, že Uzol C sa stal primárnym uzlom, pretože je definovaný ako druhý záložný uzol. Aktuálna a preferovaná rola uzla D je replikovaný uzol. Replikovaný uzol sa pri prepnutí ani núdzovom prepnutí nemôže stať prístupovým bodom, pokiaľ jeho rolu manuálne nezmeníte na primárny alebo záložný uzol.

**Poznámka:** Rola každého uzla v doméne obnovy sa dá zmeniť manuálne. Príklad ilustruje spôsob, akým sa roly v doméne obnovy menia pri vykonaní akcie prepnutia alebo núdzového prepnutia, pričom sa nezmení návrh rolí v doméne obnovy.

## Model s partnerskými uzlami

V prípade modelu s partnerskými uzlami môže mať uzol v skupine prostriedkov klastra jednu z dvoch rolí: partnerský uzol alebo replikovaný uzol.

Tabuľka 2. Roly uzlov pre skupiny CRG s partnerskými uzlami

Uzol	Aktuálna rola	Preferovaná rola
A	Partnerský	Partnerský
B	Partnerský	Partnerský
C	Partnerský	Partnerský
D	Replikovať	Replikovať

Uzly A, B a C sú v doméne obnovy definované ako partnerské uzly. Keď dôjde k zlyhaniu v Uzle A, oznámi sa to všetkým uzlom v doméne obnovy bez ohľadu na aktuálnu rolu. Tieto uzly budú pokračovať v operácii v bode, kde Uzol A zlyhal. Uzol D obsahuje údaje, ale nemôže pokračovať v operácii, pretože je definovaný ako replikovaný uzol.

Ako partnerské alebo replikované uzly môžete navrhnuť ľubovoľný počet uzlov. Partnerské uzly nemajú určené poradie a môžu sa stať aktívnym prístupovým bodom k prostriedkom klastra. Replikované uzly nemajú určené poradie a nemôžu sa stať aktívnym prístupovým bodom k prostriedkom klastra, pokiaľ sa pomocou API QcstChangeClusterResourceGroup (Change Cluster Resource Group) nezmení ich rola z replikovaného na partnerský uzol.

### Súvisiace úlohy

“Zmena domény obnovy pre skupinu klastrových prostriedkov” na strane 104

Môžete zmeniť roly uzlov v doméne obnovy pre skupinu prostriedkov klastra, ako aj pridať alebo odstrániť uzly z domény obnovy. Pre skupinu prostriedkov klastra zariadenia tiež môžete zmeniť názov lokality a adresy IP dátových portov pre uzol v doméne obnovy.

“Vykonanie prepnutia” na strane 104

Vykonanie manuálneho prepnutia spôsobí, že sa aktuálny primárny uzol prepne na záložný uzol, ako je to definované v doméne obnovy skupiny prostriedkov klastra.

## Verzia klastra

*Verzia klastra* reprezentuje úroveň funkcií dostupných v klastrí.

Nastavovanie verzii je technika, ktorá umožňuje, aby klaster obsahoval systémy s viacerými úrovňami vydania, pre ktoré určíte používanú úroveň protokolu, aby mohli navzájom správne spolupracovať. Ak používate klaster, ktorý bude obsahovať systémy s rôznymi úrovňami vydania, pozrite si tému *Klastre s viacerými verziami*.

Existujú dve verzie klastrov:

### Potenciálna verzia klastra

Predstavuje najvyvinutejšiu dostupnú úroveň klastrových funkcií pre daný uzol. Toto je verzia, s ktorou je uzol schopný komunikácie s inými uzlami klastra.

### Aktuálna verzia klastra

Predstavuje verziu, ktorá je aktuálne používaná na všetky operácie klastra. Toto je verzia komunikácií medzi uzlami v klastrí.

Potenciálna verzia klastra sa zvyšuje v každom vydaní operačného systému, ktoré obsahuje dôležité nové funkcie pre klastre, nedostupné v starších verziách. Ak je aktuálna verzia klastra menšia než potenciálna verzia klastra, túto funkciu nemôžete použiť, pretože niektoré uzly nemôžu rozoznať alebo spracovať danú požiadavku. Ak chcete využiť takúto novú funkciu, všetky systémy v klastrí musia mať rovnakú potenciálnu verziu klastra a aktuálna verzia klastra musí byť taktiež nastavená na túto úroveň.

Keď sa uzol pokúša pripojiť do klastra, jeho potenciálna verzia klastra sa porovná s aktuálnou verziou klastra. Ak hodnota potenciálnej verzie klastra nie je rovnaká ako aktuálna (N) alebo nie je rovná ďalšej úrovni verzie (N+1), potom tento uzol nemá dovolené pripojiť sa do klastra. Všimnite si, že aktuálna verzia klastra je pôvodne určená podľa prvého uzla zadaného v klastrí prevzatím hodnoty určenej príkazom, alebo API na vytvorenie klastra.

- | Ak napríklad chcete, aby uzly so systémom V5R3 spolupracovali s uzlami so systémom V5R4, môžete vykonať jedno z tohto:
- | • Vytvorte klaster v systéme V5R3 a pridajte doň uzol so systémom V5R4.
- | • Vytvorte klaster v systéme V5R4, pričom povolíte pridávanie starších uzlov do klastra, potom pridajte systémy V5R3.

V klastrí s viacerými vydaniaми, protokoly klastrov budú vždy používať najnižšiu úroveň vydania uzla, verziu aktuálneho klastra. Toto je definované, keď prvý krát vytvárate klaster. N môže byť nastavené na potenciálnu verziu uzla, bežiacu na uzle, z ktorého pochádza požiadavka na vytvorenie klastra, alebo na verziu klastra, ktorá je o jedno nižšia ako potenciálna verzia pôvodného uzla. Uzly v klastrí sa môžu líšiť najviac o jednu verziu klastra.

Keď rozšírite všetky systémy klastra na nové vydanie, môžete rozšíriť verziu klastra, aby boli dostupné nové funkcie. Toto môže byť vykonané upravením verzie klastra.

**Upozornenie:** Ak nová verzia operačného systému nie je rovná alebo o jednu verziu novšia než aktuálna verzia klastra, uzol klastra po reštartovaní zlyhá. Pri zotavovaní z tejto situácie musíte uzol vymazať a znova ho vytvoriť so správnou verziou.

- | **Upozornenie:** Keď vo vašom klastrí používate prepínateľné nezávislé diskové oblasti, pri vykonávaní prepnutia medzi vydaniaми platia isté obmedzenia. Nezávislú diskovú oblasť predošlého vydania musíte prepnúť na systém používajúci aktuálne vydanie i5/OS a sprístupniť ju. Po jej sprístupnení v systéme používajúcom aktuálne vydanie i5/OS sa jej interný obsah zmení a už ju nebudete môcť znova sprístupniť systému s predošlým vydaním.

Viac o verziách klastra sa dozviete v dokumentácii Rozhrania API pre klastre, vrátane informácií o obmedzeniach a o vzťahu medzi verziami klastra a vydaniaми i5/OS.

### Súvisiace koncepty

“Klastre s viacerými vydaniaми” na strane 84

Ak vytvárate klaster, ktorý bude obsahovať uzly s rôznymi verziami klastrovania, musíte vykonať určité kroky.

“Bežné problémy s klastrom” na strane 132

Táto téma uvádza niektoré bežné problémy, ku ktorým môže dôjsť v klastri, ako aj spôsoby, ako sa im vyhnúť alebo vykonať obnovu.

### Súvisiace úlohy

“Vytvorenie klastra” na strane 97

Ak chcete vytvoriť a nakonfigurovať klastre, musíte do klastra zahrnúť aspoň jeden uzol a musíte mať prístup aspoň k jednému z uzlov, ktoré budú v klastri.

“Prispôsobenie klastrovej verzie pre klastre” na strane 101

Verzia klastra definuje úroveň, na ktorej navzájom aktívne komunikujú všetky uzly klastra.

## Obnoviteľné prostriedky

*Obnoviteľné prostriedky* sú systémové prostriedky, napríklad údaje, zariadenia a aplikácie, ktoré sú vysoko dostupné, ak používate vo vašich systémoch klastrovanie.

Ak by sa mal na klastrovom uzle, ktorý je primárnym bodom prístupu pre konkrétnu skupinu obnoviteľných prostriedkov klastra, uskutočniť výpadok, stal by sa bodom prístupu ďalší klastrový uzol, ktorý je pre túto skupinu zariadení určený ako záložný.

Tieto typy zariadení môžu byť obnoviteľné:

1. Údaje replikované medzi uzlami.
2. Aplikácia používajúca IP adresu, ktorá môže byť prepnutá z jedného uzla na druhý.
3. Hardvérové zariadenia, ktoré môžu byť prepnuté z jedného uzla na druhý.
4. Partnerské prostriedky, ktoré podporuje administratívna doména klastra.

Definíciu vzťahu medzi uzlami, ktorým je priradená skupina obnoviteľných prostriedkov nájdete v objekte *skupina klastrových prostriedkov (CRG)*. Skupiny prostriedkov klastra sa medzi uzlami v klastri replikujú a koordinujú pomocou služieb klastrových prostriedkov.

### Súvisiace koncepty

“Skupina klastrových prostriedkov” na strane 7

*Skupina prostriedkov klastra (CRG)* je systémový objekt i5/OS predstavujúci množinu zoskupených prostriedkov klastra, ktorá sa používa na manažovanie udalostí v prostredí s klastrami. Skupina prostriedkov klastra opisuje doménu obnovy a poskytuje názov ukončovacieho programu pre skupinu prostriedkov klastra, ktorý sa zavolá pri výskyte určitých udalostí klastra.

“Administratívna doména klastra” na strane 8

*Administratívna doména klastra* sa používa na manažovanie prostriedkov, ktoré je nutné v uzloch prostredia s klastrami udržiavať konzistentné.

### Obnoviteľné aplikácie:

*Obnoviteľná aplikácia* je aplikácia, ktorú možno reštartovať na inom uzli klastra bez toho, aby bolo potrebné znova nastaviť konfiguráciu klientov.

Ak sa chcete dozvedieť, aké charakteristiky robia aplikáciu obnoviteľnou, pozrite si časť Zabezpečenie obnoviteľnosti aplikačných programov.

Obnoviteľná aplikácia musí mať schopnosť rozpoznať dočasnú stratu pripojenia internetovým protokolom (IP) medzi klientom a serverom. Klientská aplikácia musí vedieť, že bude pripojenie IP dočasne nedostupné a namiesto skončenia, alebo začatia zotavenia po zlyhaní, sa radšej musí znova pokúsiť o prístup. Podobne ak uskutočnite prepnutie, musí serverová aplikácia vedieť, že už nie je dostupné pripojenie IP. Prípadne môže byť chybné zadanie vrátené do aplikácie servera. Pokiaľ je prijaté takéto chybné zadanie, je najlepšie, keď ho aplikácia servera dokáže rozoznať a ukončiť normálnym spôsobom.

Prevzatie IP adresy je ľahko dostupná funkcia, ktorá sa používa na ochranu klientov pred výpadkom aplikácie.

**Aplikačná IP adresa na prevzatie** je nestála adresa priradená aplikácii. Koncept je použiť priradenie aliasov

adresám IP na určenie nestálej IP adresy, ktorá je pridelená viacerým aplikačným serverom, alebo hositeľom. Keď zlyhá jeden aplikačný server v klastru, prevezme na seba úlohu aplikačného servera ďalší uzol klastra bez toho, aby ste museli prekonfigurovať klientov.

Pri podpore prevzatia IP adresy je tiež predstavený koncept skupín klastrových prostriedkov (CRG). Skupiny CRG aplikácií sú skupiny prostriedkov klastra, ktoré obsahujú prostriedok adresy IP pre prevzatie aplikácie a doménu obnovy. Doména obnovy obsahuje zoznam aplikačných serverov v klastru, ktoré podporujú určitú aplikáciu. Ak zlyhá jednotlivý zdroj, služby klastrových prostriedkov iniciujú zotavenie po zlyhaní na skupine, do ktorej patrí prostriedok, ktorý zlyhal.

#### **Súvisiace koncepty**

“Zabezpečenie aplikačných programov proti zlyhaniu” na strane 29

Dozviete sa, ako spraviť aplikačné programy obnoviteľnými.

“Doména obnovy” na strane 11

*Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

#### **Súvisiace úlohy**

“Klastrové aplikácie” na strane 28

Obnoviteľnosť aplikácií je jedným z kľúčových prvkov v prostredí s klastrami. Ak máte v pláne písať pre váš klaster vysokodostupné aplikácie, mali by ste brať na vedomie, že tieto aplikácie majú zvláštne špecifikácie dostupnosti.

### **Obnoviteľné údaje:**

*Obnoviteľné údaje* sú údaje, ktoré sú replikované na viac ako jednom uzle klastra.

Každý uzol v doméne obnovy obsahuje kópiu obnoviteľných údajov, udržiavanú pomocou niektorého druhu mechanizmu replikácie. Uzly, ktoré sú v doméne obnovy definované ako záložné, môžu prevziať úlohu primárneho prístupového bodu pre obnoviteľné údaje. Uzly, ktoré sú definované ako replikačné, tiež obsahujú kópiu údajov, ale nemôžu prevziať úlohu primárneho. Pováčšine sa údaje, kopírované na replikačný uzol, používajú na prácu z primárneho uzla, ako napríklad zálohovanie alebo dotazy určené len na čítanie.

#### **Súvisiace koncepty**

“Replikácia” na strane 25

*Replikácia* vytvorí kópiu niečoho v reálnom čase. To znamená, že skopíruje objekty z jedného uzla v klastru na jeden alebo viac iných uzlov v klastru.

### **Obnoviteľné zariadenia:**

*Obnoviteľné zariadenia* sú fyzické zdroje predstavované konfiguračným objektom, akým je napríklad opis zariadenia, ktoré je prístupné z viac, než jedného uzla v klastru.

V prípade výpadku sa prístupový bod pre tento prostriedok prepne na prvý záložný uzol v skupine prostriedkov klastra domény obnovy. Nezávislé diskové oblasti, alebo nezávislé oblasti ASP, sú obnoviteľné zariadenia, ktoré sa môžu prepínať medzi stavom offline a online bez dopadu na zvyšný úložný priestor systému. Okrem toho môžete použiť geografické zrkadlenie, čo je podfunkcia zrkadlenia lokalít (XSM), ktoré je súčasťou voľby 41 systému i5/OS - Vysokodostupné prepínateľné prostriedky. Geografické zrkadlenie je funkcia, ktorá udržiava dve identické kópie nezávislej diskovej oblasti v dvoch lokalitách, čím zabezpečuje vysokú dostupnosť a obnovu po zlyhaní. Kópia, ktorú vlastní primárny uzol je produkčná kópia a kópia, ktorú vlastní záložný uzol v druhej lokalite je zrkadlová kópia. Užívateľské operácie a aplikácie prístupujú k nezávislej diskovej oblasti v primárnom uzle, teda v uzle, ktorý vlastní produkčnú kópiu.

*Skupina klastrových prostriedkov obnoviteľných zariadení* môže obsahovať zoznam prepínateľných zariadení. Každé zariadenie na zozname určuje prepínateľnú nezávislú diskovú oblasť. Keď príde k poruche, celá kolekcia zariadení sa prepne na záložný uzol. Zariadenia môžu byť tiež zapnuté ako súčasť procesu prepnutia/zotavenia po zlyhaní. Existujú

obmedzenia, týkajúce sa fyzickej konfigurácie spojenej so zoznamom prepínateľných zariadení. Viac informácií o tom, ako nastaviť vhodnú konfiguráciu pre nezávislú diskovú oblasť, definovanú ako obnoviteľnú, nájdete v téme Nezávislé diskové oblasti.

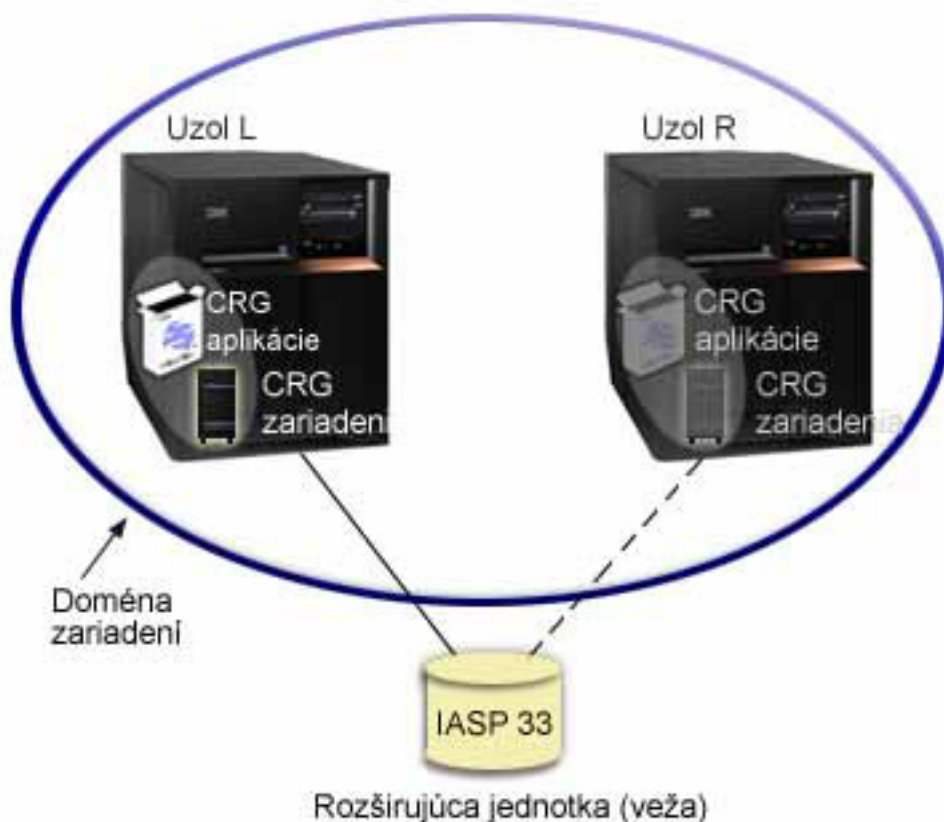
Obnoviteľné zariadenie CRG je veľmi podobné iným typom CRG. Jeden z rozdielov, zoznam prepínateľných zariadení, sme už spomenuli vyššie. Iný rozdiel je, že ukončovaci program je voliteľný pre zariadenie CRG. Ak je potrebné špecifické spracovanie kvôli prostrediu alebo údajom, ukončovaci program môže byť použitý pre CRG. Ďalšie informácie o tomto type CRG nájdete v informáciách o API QcstCreateClusterResourceGroup (Create Cluster Resource Group).

## Domény zariadení

*Doména zariadení* je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení. Konkrétne sa môžu uzly v doméne zariadení zúčastňovať akcie prepnutia pre určitú kolekciu obnoviteľných prostriedkov zariadení.

Domény zariadení sú určené a spravované skupinou rozhraní, ktoré vám do nej umožňujú uzol pridať, alebo odstrániť.

Doména zariadení je používaná na správu určitých globálnych informácií potrebných na prepnutie obnoviteľného zariadenia z jedného uzla na druhý. Aby sa predišlo konfliktom pri prepínaní zariadení, potrebujú mať tieto informácie všetky uzly v doméne zariadení. Ak ide napríklad o skupinu prepínateľných nezávislých diskových oblastí, musí byť v celej doméne zariadení jedinečná identifikácia nezávislej diskovej oblasti, zadání diskovej jednotky a zadání virtuálnej adresy.



Uzol klastra môže patriť najviac jednej doméne zariadení. Skôr, než je uzol pridaný do domény obnovy pre CRG zariadení, musí byť určený ako člen domény zariadení. Všetky uzly, ktoré budú v doméne obnovy pre CRG zariadení, musia byť v rovnakej doméne zariadení.

Ak chcete vytvoriť a manažovať domény zariadení, musíte mať vo vašom systéme nainštalovanú voľbu 41 (i5/OS - Vysokodostupné prepínateľné prostriedky) s platným licenčným kľúčom.

### Súvisiace koncepty



“Příklad: Klaster s prepínateľným diskom s použitím nezávislých diskových oblastí” na strane 118  
Klaster, ktorý používa technológiu prepínania diskov, poskytuje alternatívu k replikácii údajov. V klastrí s prepínateľnými diskami sú už údaje obsiahnuté v nezávislých diskových oblastiach (známych aj ako nezávislé ASP).

#### **Súvisiace úlohy**

“Pridanie uzla do domény pre zariadenie” na strane 105

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení.

“Odstránenie uzla z domény pre zariadenie” na strane 106

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení.

### **Voľba 41 (Vysokodostupné prepínateľné prostriedky):**

Ak chcete vytvoriť a manažovať domény zariadení, musíte mať vo vašom systéme nainštalovanú voľbu 41 (i5/OS - Vysokodostupné prepínateľné prostriedky) s platným licenčným kľúčom.

Túto vlastnosť musíte mať nainštalovanú, ak chcete vo vašom prostredí s klastrami vykonávať niektoré z týchto operácií:

- Používať rozhranie Navigátora iSeries na manažovanie klastrov.
- Prepínať nezávislé diskové oblasti (nezávislé oblasti ASP) medzi systémami.
- Používať zrkadlenie lokalít medzi geograficky vzdialenými systémami.

#### **Súvisiace úlohy**

“Pridanie uzla do domény pre zariadenie” na strane 105

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení.

“Odstránenie uzla z domény pre zariadenie” na strane 106

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení.

## **Udalosti klastra**

V klastrí môže dôjsť k niekoľkým typom udalostí, akcií a služieb.

### **Zlyhanie**

K *núdzovému prepnutiu* dochádza, keď server v klastrí vykoná pri zlyhaní systému automatické prepnutie na jeden alebo viac záložných serverov.

Naproti tomu k prepnutiu dochádza, keď manuálne prepnete prístup z jedného servera na iný. Prepnutie a zlyhanie fungujú po svojom spustení identicky. Jediným rozdielom je, ako bola udalosť spustená.

Keď sa objaví zlyhanie, je prístup prepnutý z klastrového uzla aktuálne fungujúceho v doméne obnovy CRG ako primárny uzol, na klastrový uzol určený ako prvý záložný uzol. Informácie o určovaní poradia prepnutia nájdete v téme Doména obnovy.

Keď sa vykonáva akcia núdzového prepnutia pre viac skupín prostriedkov klastra (CRG), systém najprv spracuje skupiny CRG zariadení (prepínateľné zariadenia), potom skupiny CRG údajov (prepínateľné skupiny údajov) a nakoniec skupiny CRG aplikácií (prepínateľné aplikácie).

Front správ o prevzatí služby dostáva správy ohľadne činností prevzatia služby. Môžete ho použiť pri riadení procesov zlyhania skupiny klastrových prostriedkov.

#### **Súvisiace koncepty**

“Doména obnovy” na strane 11

*Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

“Skupina klastrových prostriedkov” na strane 7

*Skupina prostriedkov klastra (CRG)* je systémový objekt i5/OS predstavujúci množinu zoskupených prostriedkov

l klastra, ktorá sa používa na manažovanie udalostí v prostredí s klastrami. Skupina prostriedkov klastra opisuje doménu obnovy a poskytuje názov ukončovacieho programu pre skupinu prostriedkov klastra, ktorý sa zavolá pri výskyte určitých udalostí klastra.

l “Front správ o prevzatí služby” na strane 113

l Front správ o prevzatí služby dostáva správy ohľadne činností prevzatia služby.

l “Hardvérové požiadavky klastrov” na strane 80

l S klastrovaním je kompatibilný každý model iSeries, v ktorom je možné použiť operačný systém i5/OS V4R4M0 alebo novší.

### l Súvisiace úlohy

l “Prepnutie” na strane 20

l K *prepnutiu* dochádza, keď manuálne prepnete prístup k prostriedku z jedného servera na iný.

### l Príklad: Zlyhanie:

l Zotavenie po zlyhaní zvyčajne vyplýva zo zlyhania uzla, ale sú aj iné príčiny, ktoré ho spôsobujú.

l Je možné, že problém ovplyvní len jedinú skupinu klastrových prostriedkov, ktorá spôsobí zotavenie po zlyhaní len pre túto skupinu (CRG).

l Nasledujúca tabuľka ukazuje rôzne typy zlyhaní a kategóriu, do ktorej patria:

Zlyhanie	Všeobecná kategória
Zlyhanie hardvéru CEC (napríklad CPU)	2
Zlyhanie komunikačného adaptéra, linky alebo smerovača alebo ENDTCPIFC, týkajúce sa všetkých IP adries rozhrania, definovaných pre uzol	4
Výpadok elektrického prúdu v CEC	1
Kontrola softvérového stroja operačného systému	2
Bol vydaný príkaz ENDTCP(*IMMED alebo *CNTRLD s časovým limitom)	1
Bol zadaný príkaz ENDSBS QSYSWRK(*IMMED alebo *CNTRLD)	1
Bol zadaný príkaz ENDSBS(*ALL, *IMMED alebo *CNTRLD)	1
Bol zadaný príkaz ENDSYS (*IMMED alebo *CNTRLD)	1
Bol zadaný príkaz PWRDWN SYS(*IMMED alebo *CNTRLD)	1
Tlačidlo Initial program load (IPL) bolo stlačené, kým boli služby klastrových prostriedkov aktívne v systéme	1
Bolo vykonané zrušenie úlohy QCSTCTL (*IMMED alebo *CNTRLD s časovým limitom)	1
Bolo vykonané zrušenie úlohy QCSTCRGM (*IMMED alebo *CNTRLD s časovým limitom)	1
Bolo vykonané zrušenie úlohy skupiny prostriedkov klastra (*IMMED alebo *CNTRLD s časovým limitom)	3
Je volané API End Cluster Node	1
Je volané API Remove Cluster Node	1
Úloha skupiny klastrových prostriedkov má softvérovú chybu, ktorá spôsobila, že skončila abnormálne	3
Zadaná funkcia 8 alebo funkcia 3 z kontrolného panelu na vypnutie systému	2



Zlyhanie	Všeobecná kategória
Zadajte funkciu 7 pre oneskorené vypnutie oddielu	1
Zlyhanie aplikačného programu pre skupinu aplikačných klastrových prostriedkov	3
Všeobecná kategória:	
<ol style="list-style-type: none"> <li>Všetky služby klastrových prostriedkov (CRS) zlyhali na uzle a zistilo sa zlyhanie uzla. Uzol môže stále fungovať alebo mohol zlyhať (napríklad kvôli strate napájania). Ak zlyhajú všetky služby klastrových prostriedkov, potom prostriedky, ktoré spravuje CRS prejdú procesom zotavenia po zlyhaní.</li> <li>Všetky CRS zlyhali na uzle, ale je to zistené ako rozklad klastra. Uzol môže i nemusí fungovať.</li> <li>Prišlo k zlyhaniu individuálnej skupiny klastrových prostriedkov. Tieto podmienky sú vždy považované za zlyhanie.</li> <li>Prišlo k zlyhaniu, ale uzol a služby klastrových prostriedkov sú stále funkčné a je to zistené ako rozklad klastra.</li> </ol>	

Keď príde k zlyhaniu, postup prijatý službami klastrových prostriedkov pre určitú skupinu klastrových prostriedkov závisí od typu zlyhania a stavu skupiny klastrových prostriedkov. Avšak vo všetkých prípadoch sa volá ukončovaci program. Núdzové prepnutie musí pracovať so zoznamom zlyhaných uzlov. Po zavolaní ukončovacieho programu, tento potrebuje určiť, či sa musí zaoberať so zlyhaním jedného uzla alebo so zoznamom zlyhávajúcich uzlov.

Ak je skupina klastrových prostriedkov *neaktívna*, stav člena zlyhaného uzla v doméne obnovy skupiny klastrových prostriedkov sa zmení buď na *Neaktívny* alebo *Oddiel*. Avšak úlohy uzla sa nezmenia a záložné uzly nebudú preradené. Pri zavolaní príkazu STRCRG (Start Cluster Resource Group) alebo API QcstStartClusterResourceGroup (Start Cluster Resource Group) sa zmení poradie záložných uzlov v neaktívnej skupine prostriedkov klastra. Ale toto API zlyhá, ak nie je aktívny primárny uzol. Musíte vykonať príkaz CHGCRG (Change Cluster Resource Group) alebo API QcstChangeClusterResourceGroup (Change Cluster Resource Group), aby ste nastavili aktívny uzol ako primárny uzol, potom znova zavolať API Start Cluster Resource Group.

Ak je skupina klastrových prostriedkov *aktívna* a zlyhávajúcí uzol *nie* je primárnym uzlom, zotavenie po zlyhaní zaktualizuje v doméne obnovy skupine klastrových prostriedkov stav člena domény obnovy, ktorý zlyhal. Ak je zlyhávajúcí uzol zálohový uzol, zoznam zálohových uzlov sa znovu usporiada tak, že aktívne uzly sú na začiatku zoznamu.

Ak je skupina prostriedkov klastra *aktívna* a člen domény obnovy je primárny uzol, podľa typu zlyhania sa vykonajú tieto akcie:

#### Zlyhanie kategórie 1

Objaví sa zotavenie po zlyhaní. Primárny uzol je označený ako *neaktívny* v každej skupine klastrových prostriedkov CRG a urobí sa z neho posledný zálohový uzol. Uzol, ktorý bol prvá záloha sa stane novým primárnym uzlom. Po zlyhaní sú najprv zotavené všetky CRG zariadení. Potom sú zotavené po zlyhaní všetky skupiny klastrových prostriedkov údajov. Nakoniec sú zotavené po zlyhaní všetky skupiny klastrových prostriedkov aplikácií. Ak proces zotavenia po zlyhaní zistí pre ktorúkoľvek CRG, že nie je aktívny ani jeden záložný uzol, je stav tejto CRG nastavený na *indoubt*.

#### Zlyhanie kategórie 2

Nastane zotavenie po zlyhaní, ale primárny uzol sa nezmení. Všetky uzly v oddieli klastrov, ktoré nemajú primárny uzol ako člena oddielu ukončia aktívnu skupinu klastrových prostriedkov. Stav uzlov v doméne obnovy v skupine klastrových prostriedkov sa nastaví na stav *oddiely* pre každý uzol, ktorý je v primárnom oddieli. Ak došlo naozaj k zlyhaniu uzla, ale to je označené len ako problém rozkladu a ak išlo o primárny uzol, stratíte všetky údaje a aplikačné služby na tomto uzle a nespustí sa žiadne automatické zotavenie po zlyhaní. Musíte buď prehlásiť uzol za zlyhaný alebo ho zapnúť a znova v ňom spustiť klastrovanie. Viac informácií nájdete v časti o zmene rozdelených uzlov pri zlyhaní.

#### Zlyhanie kategórie 3

Ak je ovplyvnená len jediná skupina klastrových prostriedkov, vyskytne sa zotavenie po zlyhaní samostatne, pretože skupiny klastrových prostriedkov sú na sebe navzájom nezávislé. Môže sa stať, že niekoľko skupín klastrových prostriedkov bude zasiahnutých v rovnakom čase kvôli tomu, že niekto zrušil niekoľko úloh klastrových prostriedkov. Tento typ zlyhania je však spracovaný len skupinou na základe CRG a nie je

vykonané žiadne koordinované zotavenie po zlyhaní medzi skupinami CRG. Primárny uzol je označený ako *neaktívny* v každej skupine klastrových prostriedkov a urobí sa z neho posledný zálohový uzol. Uzol, ktorý bol prvým zálohovým uzlom sa stane novým primárnym uzlom. Ak neexistuje žiadny aktívny zálohový uzol, stav skupiny klastrových prostriedkov je nastavený na *pochybný*.

#### Zlyhanie kategórie 4

Táto kategória je podobná kategórii 2. Kým sú však všetky uzly a služby klastrových prostriedkov na uzloch stále funkčné, nie všetky uzly môžu medzi sebou komunikovať. Z klastra je vytvorený oddiel, ale primárny uzol alebo uzly stále poskytujú služby. Vytvorenie oddielov môže spôsobiť rôzne problémy. Napríklad, ak je primárny uzol v jednom oddieli a zálohové uzly alebo replikované uzly sú v inom oddieli, potom už nereplikujete údaje a nemáte žiadnu ochranu, ak by primárny uzol zlyhal. V oddiele obsahujúcom primárny uzol proces núdzového prepnutia zaktualizuje stav uzlov v doméne obnovy skupiny prostriedkov klastra na hodnotu *rozklad* pre všetky uzly v druhom oddiele. V oddiele, ktorý neobsahuje primárny uzol, sa stav uzlov v doméne obnovy skupiny prostriedkov klastra nastavi pre všetky uzly v druhom oddiele na hodnotu *rozklad*.

#### Súvisiace koncepty

“Chyby oddielov” na strane 134

Určité klastrové stavy je možné jednoducho opraviť. Ak došlo k rozkladu klastra, dozviete sa tu, ako vykonať obnovu. Tiež sa dozviete, ako môžete zabrániť deleniu klastra a uvedie príklady, ako oddiely spojiť naspäť.

#### Prepnutie

K *prepnutiu* dochádza, keď manuálne prepnete prístup k prostriedku z jedného servera na iný.

Manuálne prepnutie iniciujete zvyčajne vtedy, keď chcete vykonať údržbu systému, napríklad aplikovanie dočasných opráv programu (PTF), inštaláciu nového vydania alebo rozšírenie vášho systému. Núdzové prepnutie sa na rozdiel od toho vykonáva automaticky, keď dôjde k výpadku primárneho uzla.

Keď sa objaví zlyhanie, je prístup prepnutý z klastrového uzla aktuálne fungujúceho ako primárny uzol v doméne obnovy skupiny klastrových prostriedkov, na klastrový uzol určený ako prvý záložný uzol. Informácie o určovaní poradia prepnutia nájdete v téme Doména obnovy.

Ak robíte administratívne prepnutie viacerých CRG, poradie, ktoré špecifikujete by malo zohľadňovať vzťahy medzi CRG. Napríklad, ak máte aplikáciu CRG, ktorá závisí od údajov spojených so zariadením CRG, kroky na nariadené prepnutie sú:

1. Zastavte aplikáciu na pôvodnom primárnom uzle (aby ste zabránili zmenám údajov).
2. Prepnite CRG zariadení na nový primárny uzol.
3. Prepnite CRG aplikácií na nový primárny uzol.
4. Reštartujte aplikáciu na novom primárnom uzle.

#### Súvisiace koncepty

“Zlyhanie” na strane 17

K *núdzovému prepnutiu* dochádza, keď server v klastru vykoná pri zlyhaní systému automatické prepnutie na jeden alebo viac záložných serverov.

“Doména obnovy” na strane 11

*Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

#### Súvisiace úlohy

“Vykonanie prepnutia” na strane 104

Vykonanie manuálneho prepnutia spôsobí, že sa aktuálny primárny uzol prepne na záložný uzol, ako je to definované v doméne obnovy skupiny prostriedkov klastra.

#### Opätovné pripojenie

*Opätovné pripojenie* znamená, že uzol sa stane aktívnym členom klastra po tom, čo bol nezúčastneným členom.

l Napríklad, keď je klastrovanie reštartované na uzle potom, čo bol uzol neaktívny a uzol klastra sa opäť pripojí do  
 l klastra. Služby klastrových prostriedkov na uzle spustíte tak, že ich naštartujete z iného uzla, ktorý už bol v klastri  
 l aktívny. Od verzie 3 sa uzol môže spustiť sám a bude schopný opätovne sa pripojiť k aktívnemu klastru, ak v ňom nájde  
 l aktívny uzol. Detaily nájdete v časti Spustenie uzla klastra.

l Predpokladajme, že uzly A, B a C vytvárajú klaster. Uzol A zlyhá. Aktívny klaster teraz pozostáva z uzlov B a C.  
 l Akonáhle je uzol, ktorý zlyhal, znova funkčný, môže sa opäť pripojiť ku klastru, ak je spustený z akéhokoľvek uzla  
 l klastra, vrátane jeho samého. Operácia opätovného pripojenia je vykonaná na základe skupiny klastrových  
 l prostriedkov, čo znamená, že sa každá skupina klastrových prostriedkov (CRG) pripája ku klastru nezávisle.

l Základná funkcia opätovného pripojenia zabezpečuje, že je objekt CRG replikovaný na všetky aktívne uzly domény  
 l obnovy. Uzol, ktorý sa opätovne pripája, ako aj všetky existujúce aktívne uzly klastrov, musia mať identickú kópiu  
 l objektu CRG. Navyše musia mať identickú kópiu niektorých interných údajov.

l Ak sa v administratívnej doméne klastra nachádzajú uzly, ktoré nie sú aktívne, všetky zmeny prostriedkov, vykonané v  
 l aktívnej doméne, sa do nich rozširujú, keď sa uzly znova pripoja do aktívnej domény.

l Keď uzol zlyhá, pokračovanie volania služieb klastrových prostriedkov vo zvyšných uzloch klastra môže zmeniť údaje  
 l v objekte CRG. Modifikácia sa musí vyskytnúť kvôli zavolaniu API alebo kvôli následnému zlyhaniu uzla. Pre  
 l jednoduché klastre je pripájaný uzol aktualizovaný s kópiou CRG z niektorého uzla, ktorý je v klastri aktuálne aktívny.  
 l Avšak nemusí to tak byť vo všetkých prípadoch.

#### l Súvisiace úlohy

l “Spustenie klastrového uzla” na strane 100

l Spustenie klastrového uzla spustí na tomto uzle služby klastrových prostriedkov. Od verzie 3 sa uzol môže spustiť  
 l sám a bude schopný opätovne sa pripojiť k aktívnemu klastru, ak v ňom nájde aktívny uzol.

l “Zmena oddielových uzlov na zlyhané” na strane 136

l V niektorých prípadoch je zobrazená podmienka rozdelenia pri skutočnom výpadku uzla. Táto situácia nastane  
 l vtedy, keď služby klastrových prostriedkov stratia komunikáciu v rámci jedného alebo medzi viacerými uzlami a  
 l nedá sa zistiť ich funkčnosť. V takomto prípade je tu pre vás jednoduchý mechanizmus zobrazenia, či uzol zlyhal.

#### l Príklad: Opätovné pripojenie:

l Táto téma opisuje akcie, ktoré sa môžu vykonať pri opätovnom pripojení uzla do klastra.

l Nasledujúci diagram popisuje, ktoré akcie sa vykonávajú, keď sa uzol znova pripája do klastra. Navyše bude v poli stav  
 l členstva v doméne obnovy CRG zmenený stav znovu sa pripájajúcich uzlov z *neaktívny* na *aktívny*. Na všetkých uzloch  
 l domény obnovy CRG je zavolaný ukončovací program a je použitý kód akcie Rejoin.

l *Tabuľka 3. Operácia opätovného pripojenia*

Operácia opätovného pripojenia			
Pripájaný uzol		Uzly klastra	
Obsahuje kópiu CRG	Neobsahuje kópiu CRG	Obsahuje kópiu CRG	Neobsahuje kópiu CRG
(1)	(2)	(3)	(4)

l S použitím vyššie uvedeného diagramu sú možné nasledujúce situácie:

l 1. 1 a 3

l 2. 1 a 4

l 3. 2 a 3

l 4. 2 a 4

l Ak má uzol v klastri kópiu CRG, všeobecné pravidlo pre opätovné pripojenie je, že CRG sa skopíruje z aktívneho uzla  
 l v klastri na pripájaný uzol.

### | **Situácia opätovného pripojenia 1**

| Kópia objektu CRG z uzla v klastru sa pošle pripájanému uzlu. Výsledok je:

- | • Objekt CRG je aktualizovaný na pripájanom uzle s údajmi, ktoré poslal klaster.
- | • Objekt CRG môže byť vymazaný z pripájaného uzla. Môže sa to stať, ak pripájaný uzol bol odstránený z domény obnovy CRG, kým bol pripájaný uzol mimo klastra.

### | **Situácia opätovného pripojenia 2**

| Kópia objektu CRG z pripájaného uzla sa pošle opätovne na všetky uzle klastra. Výsledok je:

- | • Žiadna zmena, ak žiadny z uzlov klastra nie je v doméne obnovy CRG.
- | • Objekt CRG mohol byť vytvorený na jednom alebo na viacerých uzloch klastra. Toto sa môže vyskytnúť v nasledujúcom scenári:
  - | – Uzly A, B, C a D vytvárajú klaster.
  - | – Všetky štyri uzly sú v doméne obnovy CRG.
  - | – Kým je uzol A mimo klastra, CRG sa modifikuje na odstránenie B z domény obnovy.
  - | – Uzly C a D zlyhajú.
  - | – Klaster je len uzol B, ktorý nemá kópiu CRG.
  - | – Uzol A sa opätovne pripojí do klastra.
  - | – Uzol A má CRG (aj keď má teraz nižšiu úroveň) a Uzol B nemá. CRG je vytvorené na uzle B. Keď sa uzly C a D znovu pripoja do klastra, kópia CRG v klastru aktualizuje uzol C a D a predošlá zmena odstránenia uzla B z domény obnovy je stratená.

### | **Situácia opätovného pripojenia 3**

| Kópia objektu CRG z uzla v klastru sa pošle pripájanému uzlu. Výsledok je:

- | • Žiadna zmena, ak pripájaný uzol nie je v doméne obnovy CRG.
- | • Objekt CRG možno vytvoriť na pripájanom uzle. Toto sa môže stať, ak CRG bola vymazaná na pripájanom uzle a služby klastrových prostriedkov neboli na uzle aktívne.

### | **Situácia opätovného pripojenia 4**

| Niektoré interné informácie z jedného z uzlov klastra sa môžu použiť na aktualizáciu informácií o pripájanom uzle, ale nevykoná sa nič viditeľné.

## | **Spájanie**

| Operácia *zlúčenia* je podobná operácii opakovaného pripojenia s tým rozdielom, že k nej dochádza, keď rozdelené uzly začnú znova komunikovať.

| Oddiel môže byť skutočným oddielom, ak sú služby klastrových prostriedkov stále aktívne na všetkých uzloch.

| Niektoré uzly však z dôvodu zlyhania komunikačnej linky nemôžu komunikovať s ostatnými uzlami. Problémom môže byť aj to, že uzol skutočne zlyhal, ale jeho zlyhanie nebolo zistené.

| V prvom prípade sú oddelenia automaticky spojené akonáhle je problém s komunikáciou odstránený. To sa stane, keď sa obe oddelenia periodicky snažia komunikovať s rozdelenými uzlami a napokon vzájomné spojenie obnovia. V druhom prípade musia byť na uzle, ktorý zlyhal, reštartované služby klastrových prostriedkov spustením uzla z iného uzla v klastru.

### | **Súvisiace koncepty**

| “Opätovné pripojenie” na strane 20

| *Opätovné pripojenie* znamená, že uzol sa stane aktívnym členom klastra po tom, čo bol nezúčastneným členom.

| “Chyby oddielov” na strane 134

| Určité klastrové stavy je možné jednoducho opraviť. Ak došlo k rozkladu klastra, dozviete sa tu, ako vykonať obnovu. Tiež sa dozviete, ako môžete zabrániť deleniu klastra a uvedie príklady, ako oddiely spojiť naspäť.

### | **Súvisiace úlohy**

| “Spustenie klastrového uzla” na strane 100

| Spustenie klastrového uzla spustí na tomto uzle služby klastrových prostriedkov. Od verzie 3 sa uzol môže spustiť sám a bude schopný opätovne sa pripojiť k aktívnemu klastru, ak v ňom nájde aktívny uzol.

“Zmena oddielových uzlov na zlyhané” na strane 136

V niektorých prípadoch je zobrazená podmienka rozdelenia pri skutočnom výpadku uzla. Táto situácia nastane vtedy, keď služby klastrových prostriedkov stratia komunikáciu v rámci jedného alebo medzi viacerými uzlami a nedá sa zistiť ich funkčnosť. V takomto prípade je tu pre vás jednoduchý mechanizmus zobrazenia, či uzol zlyhal.

#### **Příklad: Spojenie:**

K operáciám zlúčenia môže dôjsť v niekoľkých rôznych situáciách.

Operácia spojenia sa môže objaviť pri jednej z nasledujúcich konfigurácií:

*Tabuľka 4. Zlúčenie primárneho a sekundárneho oddielu*

zlúčenie	
primárny oddiel	sekundárny oddiel

*Tabuľka 5. Zlúčenie dvoch sekundárnych oddielov*

zlúčenie	
sekundárny oddiel	sekundárny oddiel

Primárne a sekundárne oddiely sú spojené so skupinami klastrových prostriedkov (CRG). Pre CRG s primárnymi/záložnými uzlami sa definuje primárny oddiel ako oddiel, ktorý obsahuje uzol označený ako primárny prístupový bod. Sekundárny oddiel je určený ako oddiel, ktorý neobsahuje uzol označený ako primárny bod prístupu.

Ak sa v CRG s partnerskými uzlami všetky uzly domény obnovy nachádzajú v jednom oddiele, ten bude primárnym oddielom. Ak uzly domény obnovy presahujú jeden oddiel, žiaden oddiel nebude primárny. Oba oddiely budú sekundárne.

V prípade administratívnej domény klastra, ak je doména rozdelená na dva alebo viac oddielov, každý oddiel bude naďalej fungovať ako samostatná skupina. Zmeny prostriedkov sa budú v rámci každého oddielu naďalej synchronizovať. Keď sa oddiely znova zlúčia, systém zosynchronizuje zmeny z každého oddielu. Konečným výsledkom bude, že monitorované prostriedky budú v doméne konzistentné. Keď je administratívna doména klastra rozdelená, nemôžete pridávať ani odstraňovať položky MRE.

*Tabuľka 6. Zlúčenie primárneho a sekundárneho oddielu*

operácia zlúčenia			
primárny oddiel		sekundárny oddiel	
obsahuje kópiu CRG	NEOBSAHUJE kópiu CRG	obsahuje kópiu CRG	NEOBSAHUJE kópiu CRG
(1)	(2)	(3)	(4)

Počas primárno-sekundárneho spájania, ktoré je zobrazené na hornej schéme, sú možné nasledujúce situácie:

- 1 a 3
- 1 a 4
- 2 a 3 (Nie je možné, pretože väčšinový oddiel má aktívny primárny uzol a musí mať kópiu CRG.)
- 2 a 4 (Nie je možné, pretože väčšinový oddiel má aktívny primárny uzol a musí mať kópiu CRG.)

#### **Situácie zlúčenia primárneho a sekundárneho oddielu**

Kópia objektu CRG je odoslaná na všetky uzly sekundárneho oddielu. Na uzloch sekundárneho oddielu môže dôjsť k nasledujúcim akciám:

- Žiadna akcia, ak pripájaný uzol nie je v doméne obnovy CRG.

- Kópia CRG na sekundárnom uzle je aktualizovaná údajmi z primárneho oddielu.
- Objekt CRG je vymazaný zo sekundárneho uzla, ak už viac sekundárny uzol nie je v doméne obnovy CRG.
- Na sekundárnom uzle je vytvorený objekt CRG, ak takýto objekt neexistuje. Uzol je však v doméne obnovy kópie CRG, ktorá je odoslaná z primárneho oddielu.

Tabuľka 7. Scenár zlúčenia dvoch sekundárnych oddielov

operácia zlúčenia			
sekundárny oddiel		sekundárny oddiel	
obsahuje kópiu CRG	NEOBSAHUJE kópiu CRG	obsahuje kópiu CRG	NEOBSAHUJE kópiu CRG
(1)	(2)	(3)	(4)

Počas sekundárno-sekundárneho spájania, ktoré je zobrazené na hornej schéme, sú možné nasledujúce situácie:

- 1 a 3
- 1 a 4
- 2 a 3
- 2 a 4

### Situácia zlúčenia dvoch sekundárnych oddielov 1

Pre CRG s primárnymi/záložnými uzlami sa vyberie uzol s poslednou zmenou CRG, aby odoslal kópiu objektu CRG do všetkých uzlov v druhom oddiele. Ak je vybraných viac uzlov, pretože sa javí, že všetky obsahujú najaktuálnejšiu zmenu, je pri výbere uzla použitá doména obnovy.

Pri zlučovaní dvoch sekundárnych oddielov pre CRG s partnerskými oddielmi sa skopíruje verzia CRG s partnerskými uzlami s aktívnym stavom do ostatných uzlov v druhom oddiele. Ak majú oba oddiely rovnaký stav CRG s partnerskými uzlami, do uzlov v druhom oddiele sa skopíruje oddiel, ktorý obsahuje prvý uzol uvedený v doméne obnovy skupiny prostriedkov klastra.

V uzloch cieľového oddielu v CRG s primárnymi/záložnými uzlami alebo v CRG s partnerskými uzlami sa môžu vykonať tieto akcie:

- Žiadna akcia, ak pripájaný uzol nie je doménou obnovy CRG.
- CRG je vytvorená na uzle, ak je uzol v doméne obnovy kópie CRG objektu, ktorý prijíma.
- CRG je vymazaná z uzla, ak uzol nie je v doméne obnovy kópie CRG objektu, ktorý prijíma.

### Situácie zlúčenia dvoch sekundárnych oddielov 2 a 3

Je určený uzol, ktorý obsahuje kópiu CRG objektu, aby odoslal údaje objektu na všetky uzly v inom oddieli. CRG objektu môže byť vytvorená na uzloch prijímajúcich údaje, ak je uzol v doméne obnovy CRG.

### Situácia zlúčenia dvoch sekundárnych oddielov 4

Aby sa zabezpečila konzistentnosť vo všetkých častiach klastra, sú vymieňané interné údaje.

Primárny oddiel môže byť následne rozdelený na primárny a sekundárny oddiel. Ak primárny uzol zlyhá, služby klastrových prostriedkov (CRS) to zistia ako zlyhanie uzla. Z primárneho oddielu sa stane sekundárny. Rovnaký výsledok nastane aj v prípade, že ste ukončili primárny uzol, ktorý používa API na ukončenie uzla klastra. Sekundárny oddiel sa môže stať primárnym, ak sa v oddieli aktivuje primárny uzol, či už opätovným pripojením, alebo operáciou spojenia.

V prípade operácie spojenia je ukončovaci program zavolaný na všetkých uzloch v doméne obnovy CRG nezávisle na tom, v ktorom oddieli sa nachádzajú. Je použitý rovnaký kód akcie ako pri opätovnom pripojení. Výsledkom spojenia nie je žiadna zmena úloh, ale stav členstva uzlov v doméne obnovy CRG je zmenený z *oddiel* na *aktívny*. Keď sa spoja



| všetky oddiely, sú zmazané podmienky rozkladu a môžu byť používané všetky CRG rozhraní API.

## | Replikácia

| *Replikácia* vytvorí kópiu niečoho v reálnom čase. To znamená, že skopíruje objekty z jedného uzla v klastrí na jeden alebo viac iných uzlov v klastrí.

| Replikácia urobí a udržuje objekty na vašom systéme identickými. Ak vykonáte zmeny objektu na jednom uzle klastra, replikujú sa tieto zmeny aj na ostatné uzly.

### | Súvisiace koncepty

| “Obnoviteľné údaje” na strane 15

| *Obnoviteľné údaje* sú údaje, ktoré sú replikované na viac ako jednom uzli klastra.

| “Plán pre logickú replikáciu” na strane 87

| Pri logickej replikácii sa uchováva viacero kópií údajov. Údaje sú replikované, alebo kopírované z primárneho uzla v klastrí na uzly, ktoré sú v doméne obnovy určené ako záložné. Keď dôjde na primárnom uzle k výpadku, ostávajú údaje dostupné, keď záložný uzol prevezme primárny bod prístupu.

## | Sledovanie aktivity

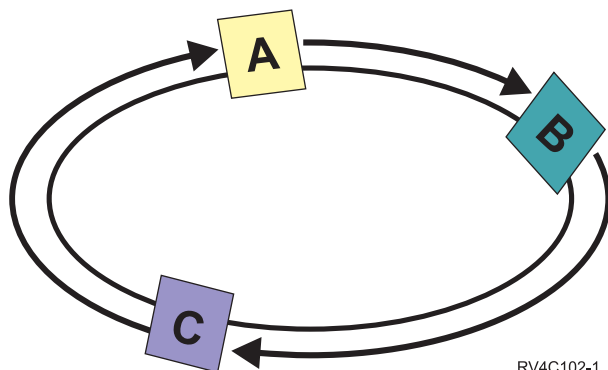
| *Monitorovanie kontrolného signálu* je funkcia služieb klastrových prostriedkov, ktorá zabezpečuje, že každý uzol je aktívny. Funkcia odosiela z každého uzla v klastrí signál do všetkých ostatných uzlov v klastrí na potvrdenie, že sú uzly stále aktívne.

| Keď kontrolný signál pre uzol zlyhá, služby klastrových prostriedkov vykonajú vhodnú akciu.

| Pomocou nasledujúcich príkladov pochopíte, ako monitorovanie aktivity funguje:

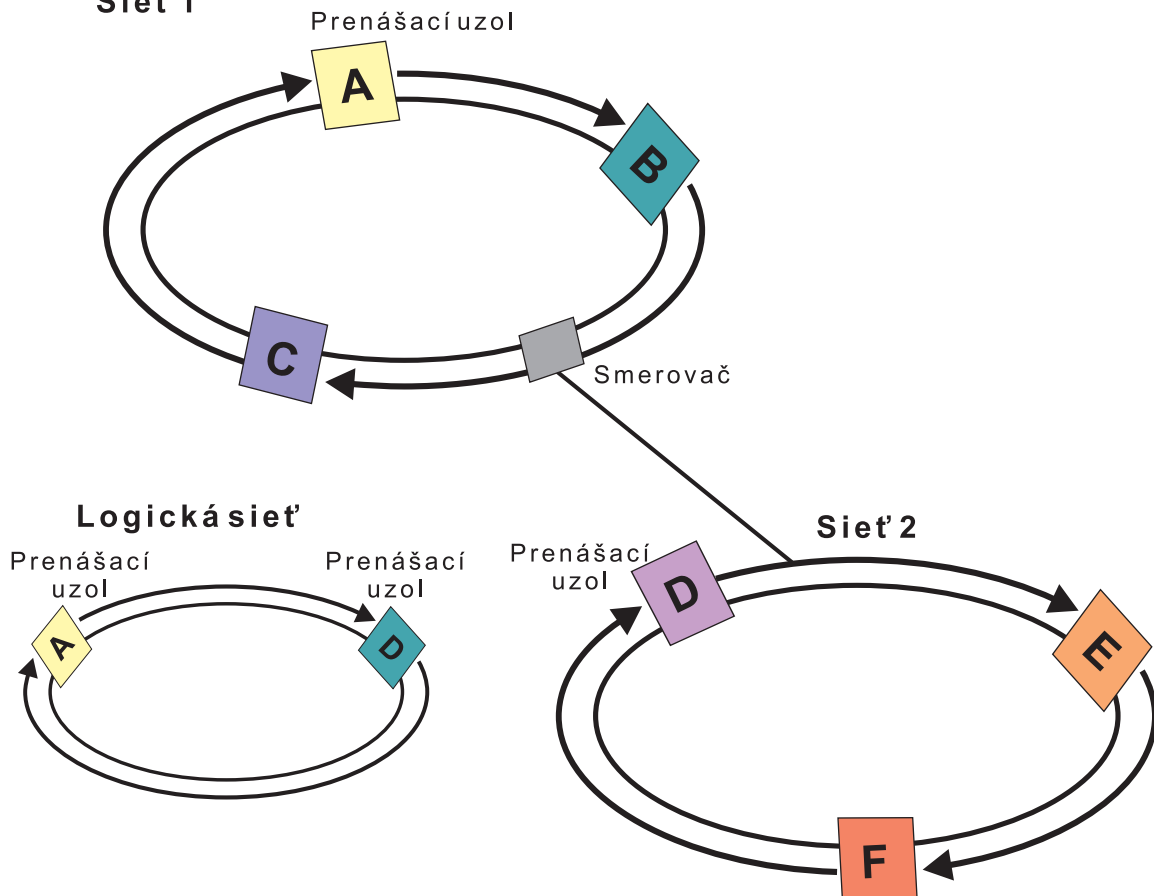
### | Príklad 1

#### | Sieť 1



| Pri štandardných (alebo normálnych) nastaveniach je z každého uzla klastra každé 3 sekundy odosielaná správa na monitorovanie aktivity k nasledujúcemu susedovi v sieti. Ak napríklad nakonfigurujete Uzol A, Uzol B a Uzol C v Sieti 1, Uzol A bude odosielať správu Uzlu B, Uzol B bude odosielať správu Uzlu C a ten bude odosielať správu Uzlu A. Uzol A očakáva potvrdenie kontrolného signálu od Uzla B, ako aj prichádzajúci kontrolný signál od Uzla C. V praxi sa tak kontrolný signál prenáša v kruhu oboma smermi. Ak Uzol A neprijme kontrolný signál od Uzla C, Uzol A a Uzol B bude pokračovať v odosielaní signálu každé 3 sekundy. Ak Uzol C vynechá štyri po sebe nasledujúce kontrolné signály, ohlásí sa zlyhanie kontrolného signálu.

## Príklad 2 Sieť 1



RV4C101-1

V tomto príklade si pridáme ešte jednu sieť, aby sme predviedli ako budú použité smerovače a uzly. Nakonfigurujete Uzol D, Uzol E a Uzol F v Sieť 2. Sieť 2 je pripojená k Sieť 1 pomocou smerovača. Smerovačom môže byť iný server iSeries alebo zariadenie smerovača, ktoré smeruje komunikáciu do iného smerovača v inom umiestnení. Každý lokálnej sieťi je priradený prenosový uzol. Tento prenosový uzol je priradený uzlu, ktorý má v sieťi najnižšie ID uzla. Uzol A je priradený ako prenášací uzol v Sieť 1 a Uzol D je priradený ako prenášací uzol v Sieť 2. Potom sa vytvorí logická sieť, obsahujúca Uzol A a Uzol D. Pomocou smerovačov a prenosových uzlov sa môžu uzly v týchto sieťach navzájom sledovať a oznamovať akékoľvek zlyhania.

### Súvisiace koncepty

“Riadenie klastrov” na strane 98

V tejto téme nájdete informácie o niektorých úlohách, ktoré sa týkajú správy vašich klastrov.

“Výkon klastra” na strane 110

Keď sú vykonané zmeny na klastri, môže to ovplyvniť réžiu spravovania klastra.

### Súvisiace úlohy

“Monitorovanie stavu klastra” na strane 109

Služby klastrových prostriedkov vykonávajú základné monitorovanie klastra a jeho komponentov pomocou funkcie garantovaných správ a monitorovania kontrolného signálu a v prípade potreby vykonávajú vhodné akcie.

## Funkcia spoľahlivých správ

Funkcia *spoľahlivých odkazov* služby klastrových prostriedkov sleduje každý uzol v klastri a zabezpečuje, že všetky uzly majú nepretržité informácie o stave klastrových prostriedkov.

Spoľahlivé odkazy používajú hodnoty opakovania pokusov a uplynutia vyhradeného času, ktoré sú pre klastrovanie jedinečné. Tieto hodnoty sú prednastavené tak, aby vyhovovali väčšine prostredí. Môžete ich však zmeniť pomocou



| rozhrania Zmena nastavení služieb klastrových prostriedkov. Pomocou hodnôt opakovania pokusov a uplynutia vyhradeného času sa určuje, koľko krát je uzlu odoslaná správa, kým bude signalizovaná situácia zlyhania oddielu. Pre lokálnu sieť (LAN), kým je signalizovaný stav oddielu, uplynie pri opakovaných pokusoch s použitím štandardných počtov opakovaní a vyhradeného času uplynutia približne 45 sekúnd. V prípade vzdialenej siete je na určenie prípadného stavu zlyhania oddielu povolený dlhší čas. Pri vzdialenej sieti môžete rátať približne so 4 minútami a 15 sekundami.

#### | **Súvisiace koncepty**

| “Zmena nastavení služieb klastrových prostriedkov”

| Predvolené hodnoty ovplyvňujúce vyhradený čas a opakovanie správ sú nastavené tak, aby vyhovovali najtypickejším inštaláciám. Tieto hodnoty však môžete zmeniť, aby lepšie zodpovedali vášmu komunikačnému prostrediu.

| “Riadenie klastrov” na strane 98

| V tejto téme nájdete informácie o niektorých úlohách, ktoré sa týkajú správy vašich klastrov.

#### | **Súvisiace úlohy**

| “Monitorovanie stavu klastra” na strane 109

| Služby klastrových prostriedkov vykonávajú základné monitorovanie klastra a jeho komponentov pomocou funkcie garantovaných správ a monitorovania kontrolného signálu a v prípade potreby vykonávajú vhodné akcie.

### | **Zmena nastavení služieb klastrových prostriedkov**

| Predvolené hodnoty ovplyvňujúce vyhradený čas a opakovanie správ sú nastavené tak, aby vyhovovali najtypickejším inštaláciám. Tieto hodnoty však môžete zmeniť, aby lepšie zodpovedali vášmu komunikačnému prostrediu.

| Hodnoty môžu byť zmenené jedným z týchto spôsobov:

- | • Nastavte všeobecnú úroveň výkonu, ktorá sa zhoduje s vaším prostredím
- | • Nastavte hodnoty špecifických ladiacich parametrov na presnejšie nastavenie.

| Prvým spôsobom je premávka správ nastavená na jednu z troch komunikačných úrovní. Normálna úroveň je predvolená a je detailne opísaná v téme Monitorovanie kontrolného signálu.

| Druhý spôsob by mal byť vykonaný len na radu odborníka.

| API QcstChgClusterResourceServices (Change Cluster Resource Services) opisuje detaily oboch metód.

#### | **Súvisiace koncepty**

| “Sledovanie aktivity” na strane 25

| *Monitorovanie kontrolného signálu* je funkcia služieb klastrových prostriedkov, ktorá zabezpečuje, že každý uzol je aktívny. Funkcia odosiela z každého uzla v klastrí signál do všetkých ostatných uzlov v klastrí na potvrdenie, že sú uzly stále aktívne.

### | **Oddiel klastra**

| *Oddiel klastra* je podskupina aktívnych uzlov klastra, ktorá vzniká následkom zlyhania spojenia. Členovia podskupiny si medzi sebou udržiavajú vzájomnú pripojiteľnosť.

| K zlyhaniu klastra dochádza vždy, keď sa stratí spojenie medzi dvoma, alebo viacerými uzlami v klastrí a keď zlyhanie stratených uzlov nemožno potvrdiť. Pri zistení stavu rozkladu klastra obmedzia služby klastrových prostriedkov typy akcií, ktoré môžete vykonávať s uzlami v oddieli klastrov. Obmedzenie funkcie počas rozkladu sa vykonáva preto, aby mohli služby klastrových prostriedkov zlúčiť oddiely, keď sa odstráni problém.

| Keď je klaster rozložený, určité operácie CRG sú obmedzené. Detaily o tom, ktoré operácie sú obmedzené pre jednotlivé typy oddielov, nájdete v časti Rozhrania API pre skupiny prostriedkov klastra.

| Ak je administratívna doména klastra rozložená, zmeny sa budú naďalej synchronizovať medzi aktívnymi uzlami v každom oddiele. Keď sa uzly znova zlúčia, administratívna doména klastra rozšíri všetky vykonané zmeny v každom klastrí, aby boli v aktívnej doméne prostriedky konzistentné.

#### | **Súvisiace koncepty**

- | “Zabránenie rozdeleniu klastra” na strane 83
- | Typickému rozkladu klastra súvisiacemu so sieťou môžete najlepšie predísť tak, že medzi všetkými uzlami klastra nakonfigurujete redundantné komunikačné cesty.
- | “Chyby oddielov” na strane 134
- | Určité klastrové stavy je možné jednoducho opraviť. Ak došlo k rozkladu klastra, dozviete sa tu, ako vykonať obnovu. Tiež sa dozviete, ako môžete zabrániť deleniu klastra a uvedie príklady, ako oddiely spojiť naspäť.
- | “Hardvérové požiadavky klastrov” na strane 80
- | S klastrovaním je kompatibilný každý model iSeries, v ktorom je možné použiť operačný systém i5/OS V4R4M0 alebo novší.

## Klastrové aplikácie

- | Obnoviteľnosť aplikácií je jedným z kľúčových prvkov v prostredí s klastrami. Ak máte v pláne písať pre váš cluster vysokodostupné aplikácie, mali by ste brať na vedomie, že tieto aplikácie majú zvláštne špecifikácie dostupnosti.

Ak budete vo vašom klastri používať obnoviteľné aplikácie, budete môcť aplikáciu opakovane spustiť v inom uzle klastra bez nutnosti zmeny konfigurácie klientov. Navyše budú údaje spojené s aplikáciou dostupné aj po zlyhaní, alebo prepnutí po zlyhaní. To znamená, že užívateľ aplikácie môže zaznamenať pri práci len minimálne alebo žiadne prerušenie, kým sa aplikácia a jej údaje prepnú z primárneho uzla do záložného uzla. Užívateľ nepotrebuje vedieť, že sa aplikácia a jej údaje presunuli na koncové zariadenie.

- | Ak chcete vo vašom klastri dosiahnuť obnoviteľnosť aplikácií, musia byť použité aplikácie, ktoré spĺňajú určité špecifikácie dostupnosti. Aby bola aplikácia v klastri prepínateľná, teda vždy dostupná pre užívateľov, musí spĺňať určité charakteristiky. Detaily o týchto črtách aplikácií nájdete v dokumente High Availability and Clusters. Keďže je nutné splniť uvedené požiadavky, máte tieto možnosti pre používanie prepínateľnej aplikácie v klastri:

### 1. Zakúpiť softvérovú aplikáciu podporujúcu klastre

Niektoré softvérové produkty podporujúce klastre vyhovujú určitým požiadavkám na vysokú dostupnosť.

### 2. Napísať alebo zmeniť vlastnú aplikáciu, aby bola vysokodostupná

Nezávislí dodávatelia softvéru môžu prispôsobiť aplikácie, aby umožnili ich prepínanie v prostredí s klastrami iSeries.

Keď už máte obnoviteľnú aplikáciu, musíte ju spravovať v rámci vášho klastra.

#### Súvisiace koncepty

“Obnoviteľné aplikácie” na strane 14

*Obnoviteľná aplikácia* je aplikácia, ktorú možno reštartovať na inom uzli klastra bez toho, aby bolo potrebné znova nastaviť konfiguráciu klientov.

## Architektúra i5/OS pre aplikácie s podporou klastrov

Ďalšie zhodnotenie pre koncového užívateľa predstavuje aplikácia, ktorá je vysokodostupná a rozpoznáva aplikácie, ktoré zostanú dostupné aj v prípade výpadku, plánovaného alebo neplánovaného.

i5/OS poskytuje architektúru pre obnoviteľnosť aplikácií, ktorá podporuje rôzne stupne vysokodostupných aplikácií. Navrchu tohto spektra budú aplikácie vylepšené pomocou integrovaných funkcií, ktoré demonštrujú charakteristiky vysokej dostupnosti a automatizáciu vysokodostupného prostredia, riadeného pomocou nástrojov na správu klastra.

Tieto aplikácie môžu mať takéto charakteristiky:

- Ak prestane byť primárny uzol dostupný, dokáže sa aplikácia prepnúť na záložný uzol klastra.
- V časti Definícia obnoviteľnosti a údaje o stave aplikácie definuje obnoviteľné prostredie, čím povolí, aby bola táto aplikácia automaticky konfigurovaná a aktivovaná aplikáciou spravujúcou klastre.
- Aplikácia poskytuje obnoviteľnosť pomocou ukončovacieho programu pre CRG, ktorý spracúva udalosti súvisiace s klastrom a využíva schopnosti služieb klastrových prostriedkov i5/OS.
- Aplikácia poskytuje funkciu reštartu aplikácie, ktorá premiestni užívateľa na obrazovku ponuky aplikácie, alebo za ňu.

Aplikácie, ktoré demonštrujú prísnu dostupnosť a charakteristiky reštartu majú tieto charakteristiky:

- Aplikácia musí svojim ukončovacím programom CRG poskytovať dokonalejšiu obnoviteľnosť náročnejším zvládnutím klastrových udalostí (kódy akcií).
- Aplikácia musí poskytovať vyšší stupeň podpory reštartu aplikácie. Pri aplikáciách zameraných na hostiteľa, bude užívateľ umiestnený na okraj operácie pomocou kontroly odovzdania alebo funkciami kontrolného bodu. Pri aplikáciách zameraných na klienta nastane zotavenie po zlyhaní s minimálnym prerušením služby.

#### **Súvisiace koncepty**

Vysoká dostupnosť a klastre iSeries

## **Tvorba vysoko dostupnej klastrovej aplikácie**

Vysoko dostupná aplikácia je tá, ktorá môže byť obnoviteľná pri výpadku systému v klastrovom prostredí.

Existuje niekoľko úrovní dostupnosti aplikácie:

1. Ak sa objaví chyba aplikácie, reštartuje sa aplikácia na rovnakom uzle a opraví akékoľvek potenciálne príčiny chyby (ako napríklad poškodené riadiace údaje). Na aplikáciu sa môžete pozerať, akoby sa spustila po prvý raz.
2. Aplikácia vykonáva pri reštarte určité množstvo kontrolných procesov. Na aplikáciu sa môžete pozerať, akoby bola blízko bodu zlyhania.
3. Ak nastane výpadok systému, reštartuje sa aplikácia na záložnom serveri. Na aplikáciu sa môžete pozerať, akoby sa spustila po prvý raz.
4. Ak nastane výpadok systému, reštartuje sa aplikácia na záložnom serveri a pri reštarte vykoná na serveroch určité množstvo kontrolných procesov. Na aplikáciu sa môžete pozerať, akoby bola blízko bodu zlyhania.
5. Ak dôjde k výpadku systému, vykoná sa koordinované núdzové prepnutie aplikácie aj jej priradených údajov do iného uzla alebo uzlov v klastri. Na aplikáciu sa môžete pozerať, akoby sa spustila po prvý raz.
6. Ak dôjde k výpadku systému, vykoná sa koordinované núdzové prepnutie aplikácie aj jej priradených údajov do iného uzla alebo uzlov v klastri. Aplikácia vykonáva pri reštarte určité množstvo kontrolných procesov na serveroch. Na aplikáciu sa môžete pozerať, akoby bola blízko bodu zlyhania.

**Poznámka:** V prípadoch 1 až 4 hore ste zodpovedný za obnovu údajov.

### **Zabezpečenie aplikačných programov proti zlyhaniu:**

Dozviete sa, ako spraviť aplikačné programy obnoviteľnými.

Obnoviteľná aplikácia by mala mať nasledovné atribúty:

- Aplikáciu je možné reštartovať na hociktorom uzle
- Aplikácia je prístupná klientovi prostredníctvom IP adresy
- Aplikácia buď nemá svoju štruktúru, alebo ju má a poznáme ju
- Údaje spojené s aplikáciou sú dostupné aj po prepnutí po zlyhaní

| Existujú tri základné prvky, ktoré vytvárajú aplikáciu odolnú voči systémovým výpadkom v prostredí klastrovania:

#### **Samotná aplikácia**

Aká je úroveň tolerancie aplikácie voči chybám a systémovým výpadkom a ako prehľadne sa aplikácia dokáže reštartovať?

Aplikácia to môže vykonať pomocou schopností klastra.

#### **Priradené údaje**

Ak dôjde k výpadku je postihnutá dostupnosť pripojených údajov?

Toto môže zabezpečovať produkt strednej úrovne na replikáciu údajov od obchodného partnera IBM, ktorý využíva schopnosti klastrovania. Prípadne môžu byť údaje uložené v prepínateľnej nezávislej diskovej oblasti (prepínateľné nezávislé ASP).

## Riadenie schopností a správa

Aká je náročnosť naefinovania prostredia, ktoré podporuje dostupnosť údajov a aplikácie?

Toto môže zabezpečiť riešenie manažovania klastra od tretej strany, ktoré používa rozhrania API klastrovania a tiež kombinuje obnoviteľné aplikácie s obnoviteľnými údajmi.

## Reštartovanie vysoko dostupných aplikácií klastra:

Ak chceme reštartovať aplikáciu, aplikácia musí poznať svoj stav v čase prevzatia alebo prepnutia služby.

Stavové informácie sú aplikačne špecifické; teda aplikácia musí určovať, ktoré informácie budú potrebné. Ak nemáte informácie o stave, potom na vašom PC môžete reštartovať aplikáciu, Budete však musieť znova nastaviť vašu pozíciu v aplikácii.

Na uloženie stavu aplikácie pre zálohový systém máte k dispozícii niekoľko metód. Každá aplikácia musí určiť, ktorá metóda je pre ňu najlepšia.

- Aplikácia môže preniesť všetky stavové informácie na požadovaný klientsky systém. Keď nastane prepnutie alebo prevzatie služby, aplikácia použije uložený stav na klientovi a obnoví stav na novom servere. Toto môžete vykonať pomocou API na distribuovanie informácií alebo rozhraní API pre klastrované hašovacie tabuľky.
- Stavové informácie (napr. informácie o úlohe a ďalšie kontrolné štruktúry, ktoré sú priradené k aplikácii) môže aplikácia replikovať v reálnom čase. Pri každej zmene štruktúry preniesie aplikácia zmenu cez zálohový systém.
- Aplikácia môže uložiť vhodné stavové informácie, ktoré sú k nej priradené, v údajovej časti ukončovacieho programu pre skupinu klastrových prostriedkov pre túto aplikáciu. Táto metóda očakáva, že je požadované malé množstvo stavových informácií. Môžete to vykonať pomocou API QcstChangeClusterResourceGroup (Change Cluster Resource Group).
- Aplikácia môže uložiť stavové informácie v údajovom objekte, ktorý je replikovaný v zálohovom systéme s údajmi aplikácie.
- Aplikácia môže uložiť stavové informácie v údajovom objekte, ktorý je spolu s údajmi aplikácie v prepínateľnom IASP.
- Aplikácia môže uchovávať stavové informácie o klientovi.
- Nie sú uložené žiadne stavové informácie a vy potrebujete urobiť obnovu.

**Poznámka:** Množstvo informácií, ktoré je potrebné uložiť sa zníži, ak aplikácia používa niektorú formu spracovania reštartu kontrolných bodov. Stavové informácie sa uložia iba vo vopred určenom kontrolnom bode aplikácie. Pri reštarte sa vrátite do posledného uloženého kontrolného bodu, čo pripomína spôsob, akým pracuje kontrola potvrdenia v databázach procesu.

## Volanie ukončovacieho programu skupiny klastrových prostriedkov:

Ukončovací program skupiny klastrových prostriedkov je volaný v rozličných fázach klastrového prostredia.

Tento program vytvára potrebné prostredie pre obnoviteľnosť prostriedkov v klastri. Ukončovací program je pre obnoviteľné zariadenie CRG voliteľný, ale je povinný pre ostatné typy CRG. Keď sa používa ukončovací program pre skupinu prostriedkov klastra, volá sa pri výskyte celoklastrových udalostí, vrátane týchto:

- Uzol opustí klaster neočakávane.
- Uzol opustí klaster ako dôsledok API QcstEndClusterNode (End Cluster Node) alebo API QcstRemoveClusterNodeEntry (Remove Cluster Node Entry).
- Klaster sa vymaže ako dôsledok API QcstDeleteCluster (Delete Cluster).
- Uzol sa aktivuje pomocou API QcstStartClusterNode (Start Cluster Node).
- Je obnovená komunikácia s uzlom delenia.

Tento ukončovací program:

- Spúšťa sa v pomenovanej aktivačnej skupine, alebo v aktivačnej skupine volajúceho (\*CALLER).

- Ignoruje parameter opakovaného spustenia, ak má ukončovaci program nespracovanú výnimku alebo sa zruší.
- Poskytuje rutinu ošetrojúcu zrušenie.

Keď sa vykoná API pre skupinu prostriedkov klastra, ukončovaci program sa zavola zo samostatnej úlohy s užívateľským profilom, ktorý je zadaný v API QcstCreateClusterResourceGroup (Create Cluster Resource Group). Osobitnú úlohu automaticky vytvorí API, keď je zavolaný ukončovaci program. Ak je ukončovaci program pre údaje CRG neúspešný, alebo skončí neobvykle, je na všetkých aktívnych uzloch v doméne obnovy zavolaný ukončovaci program skupiny klastrových prostriedkov s kódom akcie Undo. Tento kód akcie umožňuje, aby boli všetky neukončené aktivity odvolané a aby bol obnovený pôvodný stav skupiny klastrových prostriedkov.

Ak je ukončovaci program pre aplikáciu CRG neúspešný, alebo skončí neobvykle, služby klastrových prostriedkov sa pokúsia, ak je stav CRG aktívny, reštartovať aplikáciu. Je zavolaný ukončovaci program skupiny klastrových prostriedkov s kódom akcie Restart. Ak sa aplikáciu nepodarí reštartovať pri zadanom maximálnom počte pokusov, je ukončovaci program skupiny klastrových prostriedkov zavolaný s kódom akcie zotavenia po zlyhaní. Počet reštartov môže byť vynulovaný len ak je ukončovaci program zavolaný s kódom akcie štartu, ktorý je následkom štartu CRG, zlyhania, prepnutia po zlyhaní.

Keď je spustená skupina klastrových prostriedkov, ukončovaci program CRG aplikácie zavolaný primárnym uzlom nevráti riadenie službám klastrových prostriedkov, kým sa aplikácia sama neukončí, alebo kým sa neobjaví chyba. Ak musia služby klastrových prostriedkov, po tom, čo je aplikácia CRG aktívna, upozorniť ukončovaci program aplikácie CRG na nejakú udalosť, je v inej úlohe spustená iná inštancia ukončovacieho programu. Je očakávaný návrat akéhokoľvek iného kódu akcie, než Štart alebo Reštart.

Keď je zavolaný ukončovaci program skupiny klastrových prostriedkov, je odoslaná skupina parametrov, ktoré určujú spracovávanú klastrovú udalosť, aktuálny stav klastrových prostriedkov a očakávaný stav klastrových prostriedkov.

Úplné informácie o ukončovacích programoch pre skupinu prostriedkov klastra, vrátane detailov o tom, ktoré informácie sa odovzdávajú ukončovaciemu programu pre každý kód akcie, nájdete v téme Ukončovaci program pre skupinu prostriedkov klastra v dokumentácii Rozhrania API pre klastre. V knižnici QUSRTOOL je poskytnutý príklad zdrojového kódu, ktorý môže byť použitý ako základ pre napísanie ukončovacieho programu. Pozrite si člena TCSTAPPEXT v súbore QATTSYSC.

## Úvahy o klastrových aplikáciách

Skupina klastrových prostriedkov (CRG) aplikácie spravuje jej obnoviteľnosť.

### Manažovanie adres IP pre prevzatie pre CRG aplikácie:

- | Adresy IP pre prevzatie CRG aplikácie môžete manažovať pomocou služieb klastrových prostriedkov. Môžete ich manažovať aj manuálne.

Existujú dva spôsoby, ako spravovať prevzatie IP adresy aplikácie spojené s CRG aplikácie. Najjednoduchší spôsob, ktorý je zároveň predvolený, je ponechať manažovanie adresy IP pre prevzatie službám klastrových prostriedkov. Táto metóda nariadi službám klastrových prostriedkov, aby vytvorili adresu IP pre prevzatie vo všetkých uzloch domény obnovy, vrátane uzlov, ktoré sa do domény obnovy pridajú neskôr. Keď vyberiete túto metódu, v žiadnom uzle domény obnovy nemôže byť aktuálne definovaná adresa IP pre prevzatie.

Alternatívnym spôsobom je manuálne manažovanie adres IP pre prevzatie. Pri tejto metóde služby klastrových prostriedkov nevykonajú žiadne kroky na nakonfigurovanie adresy IP pre prevzatie; za konfiguráciu je zodpovedný užívateľ. Pred spustením skupiny prostriedkov klastra musíte pridať adresu IP pre prevzatie do všetkých uzlov domény obnovy (okrem replikovaných uzlov). Každý uzol, ktorý chcete pridať do domény obnovy aktívneho CRG, musí mať pred pridaním nakonfigurovanú adresu IP pre prevzatie.

### Viacnásobné podsiete

Hoci je štandardné, aby všetky uzly domény obnovy boli v jednej sieti, je možné aby IP adresa prevzatia

aplikácie fungovala aj v rámci viacnásobných podsietí. Kroky konfigurácie adresy IP pre prevzatie aplikácie pre prípad, keď sú uzly domény obnovy vo viacerých podsieťach, nájdete v téme Povolenie prepínania aplikácií.

### Súvisiace koncepty

“Príklad: Akcie pri zlyhaní skupiny klastrových prostriedkov aplikácií” na strane 33

Pozrite si, ako funguje vzorový scenár núdzového prepnutia.

“Vytvorenie CRG aplikácie s aktívnou adresou IP pre prevzatie” na strane 103

Pri vytváraní CRG aplikácie môžete nastaviť, aby sa povolila aktívna adresa IP pre prevzatie. Toto je možné, len ak užívateľ nakonfiguruje adresu IP pre prevzatie.

### Umožnenie prepínania aplikácií v podsieťach:

Klastrovanie vo všeobecnosti vyžaduje, aby sa všetky uzly v doméne obnovy skupiny prostriedkov klastra aplikácie nachádzali v jednej lokálnej sieti LAN (používali rovnaké adresovanie podsiete).

Základný sieťový protokol, používaný na prepnutie prevzatia IP adresy konfigurovanej aplikácie v doméne obnovy z jedného uzla na druhý je Address Resolution Protocol (ARP). Je však možné rozšíriť doménu obnovy tak, aby obsahovala aj klastrové uzly, umiestnené v iných sieťach LAN oddelených smerovačmi.

Toto rozšírenie je možné pomocou podpory Virtuálnych IP adries a s využitím protokolu Routing Information Protocol (RIP) na uzloch klastra a na smerovačoch v sieti. Viac detailov nájdete v téme “Povolenie prepínania aplikácií”.

### Povolenie prepínania aplikácií:

- | Služby klastrových prostriedkov podporujú pri konfigurovaní skupín CRG užívateľom nakonfigurovanú adresu IP pre prevzatie.

Ak chcete povoliť prepínateľné prostredie, potrebujete vykonať konfiguračné kroky podľa nasledovného návodu. **Táto množina inštrukcií sa musí vykonať na všetkých uzloch v doméne obnovy a musí sa zopakovať na všetkých uzloch klastra, ktoré sa stanú uzlami v doméne obnovy danej aplikačnej CRG.**

1. Vyberte prevzatú IP adresu, ktorá bude použitá aplikačnou CRG.
  - Aby nevznikol zmätok, táto adresa by sa nemala prekrývať s inými existujúcimi adresami, použitými v klastrových uzloch alebo smerovačoch. Napríklad ak vyberiete adresu 19.19.19.19, zabezpečte, aby adresa 19.0.0.0 (19.19.0.0 alebo ...) nebola trasou, ktorá sa nachádza v tabuľkách systémového smerovania.
  - Pridajte rozhranie pre prevzatie (napríklad 19.19.19.19); vytvorte ho s opisom linky \*VIRTUALIP, maskou podsiete 255.255.255.255 (trasa hostiteľa), maximálnou prenosovou jednotkou 1500 (ľubovoľné číslo z rozsahu 576-16388) a hodnotou automatického spustenia \*NO. Táto adresa pre prevzatie (napríklad 19.19.19.19) musí existovať ako adresa \*VIRTUALIP pred tým, ako ju v ďalšom kroku identifikujete ako Priradené lokálne rozhranie. Nemusi byť však aktívna.
2. K IP adrese prevzatia priradte jednu alebo obe IP adresy, určené na komunikáciu s klastrom, keď vytvárate klaster alebo keď pridávate uzol do klastra.
  - To znamená, že napríklad vytvoríte adresu prevzatia 19.19.19.19 v priradenom lokálnom rozhraní na IP adrese klastrového uzla na Ethernetovej zbernici, aby bola lokálne použitá na klastrovanie. Túto akciu musíte vykonať pre každú adresu klastra v každom klastrovom uzle.

**Poznámka:** Ak chcete vykonať túto zmenu pod CFGTCP, adresy klastra musia byť ukončené.

3. Vytvorte klaster a prípadne skupiny CRG. Pre CRG aplikácie zadajte do poľa 'Konfigurácia adresy IP pre prevzatie' hodnotu QcstUserCfgsTakeoverIpAddr. Nespúšťajte žiadnu aplikačnú CRG.
4. Skontrolujte, či je hodnota ponuky Supply nastavená na \*YES, keď používate voľby Konfigurovanie TCP/IP aplikácií (voľba 20) pod CFGTCP, ďalej Konfigurovanie RouteD (voľba 2), Zmena atribútov RouteD (voľba 1). Ak nie je, nastavte \*YES a spustíte alebo reštartujete ROUTED (RIP alebo RIP-2) v každom klastrovom uzle.



- NETSTAT, voľba 3, zobrazí príkaz ROUTED použitím lokálneho portu, ak je spustený. Príkaz ROUTED musí byť spustený a zverejňovať trasy (Supply = \*YES) v každom klastrovom uzle domény obnovy CRG.
5. Zabezpečte, aby všetky komerčné smerovače v sieti spájajúcej domény obnovy LAN akceptovali a zverejňovali trasy hostiteľa na RIP.
    - Toto nemusí byť štandardné nastavenie smerovačov. Jazyk sa bude meniť v závislosti od výrobcu smerovača a v rozhraniach RIP budú odosielať trasy hostiteľov a budú prijímaní dynamickí hostitelia.
    - Toto sa týka rozhraní smerovača pripojených k serverom iSeries, ako aj rozhraní medzi dvoma smerovačmi
- Poznámka:** V tomto scenári nepoužívajte ako smerovač server iSeries. Použite komerčný smerovač (IBM alebo iný), ktorý je navrhnutý na smerovacie účely. Smerovanie iSeries nie je možné nakonfigurovať na túto funkciu.
6. Teraz môžete manuálne aktivovať adresu pre prevzatie v jednom z uzlov klastra, poskytnúť protokolu RIP 5 minút na rozšírenie trás a pomocou príkazu ping otestovať adresu pre prevzatie zo všetkých uzlov v doméne obnovy CRG a z vybraných klientov v lokálnych sieťach, ktorí budú používať túto adresu.
    - Po tomto overovaní teste zabezpečte, aby bola adresa prevzatia služby znova ukončená.
    - Keď sa spustia CRG, klastrovanie spustí adresu na zadanom primárnom uzle.
  7. Spustenie aplikačnej CRG.
    - Adresa prevzatia služby sa spustí klastrovaním na zadanom preferovanom uzle a RIP zverejní trasy v celej doméne obnovy. RIP môže do 5 minút aktualizovať trasy v celej doméne. Funkcia RIP je nezávislá od funkcie spustenia CRG.

#### Dôležité poznámky:

- Ak uvedená procedúra nenasleduje na všetkých uzloch klastra v doméne obnovy aplikačnej CRG, klastr sa počas procesu prepnutia zablokuje.
- Aj keď pri zlyhaní neprepíname na replikované uzly, je dobré vykonať túto procedúru na replikovaných uzloch, aby ich bolo možné meniť aj po tom, ako sa stanú zálohami.
- Ak chcete použiť viacero virtuálnych adries IP, každá z nich bude vyžadovať samostatné CRG aplikácie a samostatnú adresu IP, ku ktorej bude priradená. Touto adresou môže byť iná logická IP adresa buď na tom istom adaptéri, alebo na inom adaptéri. Musíte dávať pozor, aby ste predišli nejednoznačnostiam v tabuľkách trás. Najlepšie to dosiahnete týmto spôsobom:
  - V tabuľke trás každej virtuálnej IP adrese pridajte \*DFTRROUTE.
  - Môžete to urobiť pod CFGTCP (voľba 2).
  - Všetky parametre (vrátane ďalšieho skoku) na dosiahnutie vybraného smerovača nastavte narovnako, ale rozhranie preferovaného vytvárania väzieb by malo byť nastavené na lokálnu systémovú IP adresu priradenú k virtuálnej IP adrese, ktorá bude reprezentovaná touto trasou.

#### Príklad: Akcie pri zlyhaní skupiny klastrových prostriedkov aplikácií:

Pozrite si, ako funguje vzorový scenár núdzového prepnutia.

Toto sa stane, keď skupina prostriedkov klastra pre obnoviteľnú aplikáciu vykoná núdzové prepnutie, pretože sa prekročil limit opakovaní alebo sa zrušila úloha:

- Vo všetkých aktívnych uzloch v doméne obnovy CRG sa zavolá ukončovaci program pre skupinu prostriedkov klastra s kódom akcie núdzového prepnutia. Toto naznačuje, že služby klastrových prostriedkov pripravujú nápravu zlyhania prístupového bodu aplikácie.
- Služby klastrových prostriedkov ukončia prevzatie pripojenia IP (Internet Protocol) na primárnom uzle. Viac informácií o adrese IP pre prevzatie nájdete v téme Manažovanie adries IP pre CRG aplikácie.
- Služby klastrových prostriedkov spustia preberanie IP adries na prvom záložnom (novom primárnom) uzle.
- Služby klastrových prostriedkov odošlú úlohu, ktorá zavolá ukončovaci program skupiny klastrových prostriedkov len na novom primárnom uzle s kódom akcie Start. Táto akcia reštartuje aplikáciu.

Vyššie uvedený príklad ukazuje, ako pracuje jeden scenár na odstránenie zlyhania. Iné scenáre pri zlyhaní môžu pracovať odlišne.

### Príklad: Ukončovaci program aplikácie:

Tento príklad obsahuje kód pre vzorový ukončovaci program pre skupinu prostriedkov klastra aplikácie.

Tento vzorový kód nájdete v knižnici QUSRTOOL.

Použitím príkladov kódu súhlasíte s podmienkami v časti Informácie o licencií na kód a právne vyhlásenia.

```
/******  
/*  
/* Knižnica: QUSRTOOL  
/* Súbor: QATTSYSC  
/* Člen: TCSTAPPEXT  
/* Typ: ILE C  
/*  
/* Opis:  
/* Toto je vzorový ukončovaci program CRG aplikácie, ktorý sa zvyčajne  
/* volá pre rôzne udalosti klastrov alebo ich API. Musí sa doplniť veľa  
/* logiky, pretože táto logika je naozaj závislá od jedinečných  
/* vecí, ktoré sa pre príslušnú aplikáciu musia vykonať.  
/*  
/* Cieľom tohto príkladu je poskytnúť prostredie obsahujúce základňu pre  
/* zostavenie ukončovacieho programu CRG. Komentáre nachádzajúce sa  
/* v príklade zvyrazňujú druhy problémov, ktorým musí čeliť skutočná  
/* implementácia ukončovacieho programu.  
/*  
/* V tomto príklade sú obsiahnuté všetky kódy akcie, ktoré sa týkajú  
/* CRG aplikácie.  
/*  
/* V tcstdtaara.h sa dodáva aj s knižnicou QUSRTOOL. V súbore  
/* QATTSYSC si pozrite člen TCSTDTAARA.  
/*  
/* Protokol zmien:  
/* Príz.Príčina Verzia Dátum ID užív. Popis  
/*  
/* -----  
/* ... D98332 v5r1m0 000509 ROCH Úvodné vytvorenie.  
/* $A1 P9950070 v5r2m0 010710 ROCH Opravy údajovej oblasti  
/* $A2 D99055 v5r2m0 010913 ROCH Pridanie kódu akcie CancelFailover  
/* $A3 D98854 v5r2m0 010913 ROCH Pridanie kódu akcie VerificationPhase  
/* $A4 P9A10488 v5r3m0 020524 ROCH Pridanie príkladu kódu na čakanie na údaje  
/* CRG údajov v kóde akcie prepínania  
/*  
/*  
/******  
  
/*-----*/  
/*  
/* Súbory hlavičky  
/*  
/*-----*/  
#include /* Užitočné pri ladení */  
#include /* makro offsetof */  
#include /* systémové funkcie */  
#include /* reťazcové funkcie */  
#include /* Konštanty/štruktúry pre spracovanie výnimiek */  
#include /* Rôzne konštanty pre klaster */  
#include /* Štruktúra informácií o CRG */  
#include "qusrtool/qattsysc/tcstdtaara" /* Údajové oblasti QCSTHAAPPI/QCSTHAAPPO */  
#include /* API na získanie obsahu údajovej oblasti */  
#include /* Definícia typu kódu chyby API */  
#include /* vstavené mitime */  
#include /* vstavené waittime */
```



```

/*-----*/
/*
/* Konštanty
/*
/*-----*/
#define UnknownRole -999
#define DependCrgDataArea "QCSTHAAPPO"
#define ApplCrgDataArea "QCSTHAAPPI"
#define Nulls 0x00000000000000000000

/*-----*/
/*
/* Vo funkcii checkDependCrgDataArea() sú použité nasledujúce konštanty.
/* Prvá definuje čas spánku pred skontrolovaním údajovej oblasti.
/* Druhá definuje maximálny čas čakania, kým bude údajová oblasť
/* pripravená pred zlyhaním spustenia aplikácie, keď sa vykonáva funkcia
/* Start CRG. Tretia definuje maximálny čas čakania na funkciu
/* Initiate Switchover or failover.
/*
/*-----*/
#define WaitSecondsIncrement 30
#define MaxStartCrgWaitSeconds 0
#define MaxWaitSeconds 900

/*-----*/
/*
/* Keďže je tento ukončovací program aktualizovaný tak, aby spracúval
/* nové kódy akcie, zmeňte nižšie uvedenú definíciu na hodnotu najvyššieho
/* očíslovaného kódu akcie, ktorý sa spracúva.
/*
/*-----*/
#define MaxAc 21

/*-----*/
/*
/* Ak majú údaje ukončovacieho programu v CRG určitú štruktúru, zahrňte
/* súbor hlavičky do tejto definície štruktúry a zmeňte nižšie uvedenú
/* definíciu tak, aby radšej používala názov tejto štruktúry, ako znak.
/*
/*-----*/
#define EpData char

/*-----*/
/*
/* Zmeňte nasledujúcu definíciu na knižnicu, v ktorej sa nachádza
/* aplikácia, ako aj údajové oblasti QCSTHAAPPO a QCSTHAAPPI.
/*
/*-----*/
#define ApplLib "QGPL"

/*-----*/
/*
/* Prototypy interných funkcií.
/*
/*-----*/
static int getMyRole(Qcst_EXTP0100_t *, int, int);
#pragma argopt(getMyRole)
static int doAction(int, int, int, Qcst_EXTP0100_t *, EpData *);
#pragma argopt(doAction)
static int createCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int startCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int restartCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int endCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int verifyPhase(int, int, Qcst_EXTP0100_t *, EpData *);

```

```

static int deleteCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int memberIsJoining(int, int, Qcst_EXTP0100_t *, EpData *);
static int memberIsLeaving(int, int, Qcst_EXTP0100_t *, EpData *);
static int switchPrimary(int, int, Qcst_EXTP0100_t *, EpData *);
static int addNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int rmvNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int chgCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int deleteCrgWithCmd(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoPriorAction(int, int, Qcst_EXTP0100_t *, EpData *);
static int endNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int chgNodeStatus(int, int, Qcst_EXTP0100_t *, EpData *);
static int cancelFailover(int, int, Qcst_EXTP0100_t *, EpData *);
static int newActionCode(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoCreateCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoStartCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoEndCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoMemberIsJoining(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoMemberIsLeaving(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoSwitchPrimary(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoAddNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoRmvNode(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoChgCrg(int, int, Qcst_EXTP0100_t *, EpData *);
static int undoCancelFailover(int, int, Qcst_EXTP0100_t *, EpData *);
static void bldDataAreaName(char *, char *, char *);
#pragma argopt(bldDataAreaName)
static int checkDependCrgDataArea(unsigned int);
#pragma argopt(checkDependCrgDataArea)
static void setApp1CrgDataArea(char *);
#pragma argopt(setApp1CrgDataArea)
static void cancelHandler(_CNL_Hndlr_Parms_T *);
static void unexpectedExceptionHandler(_INTRPT_Hndlr_Parms_T *);
static void endApplication(unsigned int, int, int, Qcst_EXTP0100_t *, EpData *);
#pragma argopt(endApplication)

/*-----*/
/*                                          */
/* Niektoré rutiny ladenia                */
/*                                          */
/*-----*/
static void printParms(int, int, int, Qcst_EXTP0100_t *, EpData *);
static void printActionCode(unsigned int);
static void printCrgStatus(int);
static void printRcvyDomain(char *,
                                unsigned int,
                                Qcst_Rcvy_Domain_Array1_t *);

static void printStr(char *, char *, unsigned int);

/*-----*/
/*                                          */
/* Definície typov                        */
/*                                          */
/*-----*/

/*-----*/
/*                                          */
/* Táto štruktúra definuje údaje, ktoré budú odovzdané do obslužnej rutiny */
/* výnimky a zrušenia. Rozšírte ju informáciami jedinečnými pre aplikáciu. */
/*                                          */
/*-----*/
typedef struct {
    int *retCode;           /* Smerník na návratový kód           */
    EpData *epData;        /* Údaje ukončovacieho programu z CRG */
    Qcst_EXTP0100_t *crgData; /* Údaje CRG                           */
    unsigned int actionCode; /* Kód akcie                           */
    int role;              /* Rola domény obnovy tohto uzla       */
    int priorRole;        /* Rola tohto uzla v doméne pred obnovou */
}

```

```

} volatile HandlerDataT;

/*-----*/
/*
/* Pole smerníka funkcie pre spracovanie kódov akcie. Po aktualizovaní
/* ukončovacieho programu na spracovanie nových kódov akcie pridajte názvy
/* nových funkcií do tohto poľa smerníka funkcie.
/*
/*-----*/
static int (*fcn[MaxAc+1]) (int role,
                           int priorRole,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) = {
    newActionCode,      /* 0 - aktuálne obsadený */
    createCrg,          /* 1 */
    startCrg,           /* 2 */
    restartCrg,         /* 3 */
    endCrg,             /* 4 */
    verifyPhase,        /* 5 - aktuálne obsadený */
    newActionCode,      /* 6 - aktuálne obsadený */
    deleteCrg,          /* 7 */
    memberIsJoining,    /* 8 */
    memberIsLeaving,    /* 9 */
    switchPrimary,      /* 10 */
    addNode,            /* 11 */
    rmvNode,            /* 12 */
    chgCrg,             /* 13 */
    deleteCrgWithCmd, /* 14 */
    undoPriorAction,    /* 15 */
    endNode,            /* 16 */
    newActionCode,      /* 17 - vzťahuje sa len na CRG zariadenia */
    newActionCode,      /* 18 - vzťahuje sa len na CRG zariadenia */
    newActionCode,      /* 19 - vzťahuje sa len na CRG zariadenia */
    chgNodeStatus,      /* 20 */
    cancelFailover      /* 21 */
};

/*-----*/
/*
/* Pole smerníka funkcie pre spracovanie pred kódmi akcie, keď boli
/* zavolané kódom akcie Späť. Po aktualizovaní ukončovacieho programu na
/* spracovanie akcie Späť pre nové kódy akcie pridajte do tohto poľa
/* smerníka funkcie názvy nových funkcií.
/*
/*-----*/
static int (*undoFcn[MaxAc+1]) (int role,
                                int priorRole,
                                Qcst_EXTP0100_t *crgData,
                                EpData *epData) = {
    newActionCode,      /* 0 - aktuálne obsadený */
    undoCreateCrg,      /* 1 */
    undoStartCrg,       /* 2 */
    newActionCode,      /* 3 */
    undoEndCrg,         /* 4 */
    newActionCode,      /* 5 - pre tento kód akcie nie je akcia späť */
    newActionCode,      /* 6 - aktuálne obsadený */
    newActionCode,      /* 7 */
    undoMemberIsJoining, /* 8 */
    undoMemberIsLeaving, /* 9 */
    undoSwitchPrimary,  /* 10 */
    undoAddNode,        /* 11 */
    undoRmvNode,        /* 12 */
    undoChgCrg,         /* 13 */
    newActionCode,      /* 14 */
    newActionCode,      /* 15 */
};

```

```

    newActionCode,      /* 16 */
    newActionCode,      /* 17 - vzťahuje sa len na CRG zariadenia */
    newActionCode,      /* 18 - vzťahuje sa len na CRG zariadenia */
    newActionCode,      /* 19 - vzťahuje sa len na CRG zariadenia */
    newActionCode,      /* 20 */
    undoCancelFailover /* 21 */
};

```

```

/*****
/*
/* Toto je vstupný bod ukončovacieho programu.
/*
/*
/*****
void main(int argc, char *argv[]) {

```

```

    HandlerDataT hdldata;

```

```

/*-----*/
/*
/* Pretypovanie každého argumentu odovzdaného v poli argv
/* na správny typ údajov.
/*
/*

```

```

/*-----*/
    int *retCode      = (int *)argv[1];
    unsigned int *actionCode = (unsigned int *)argv[2];
    EpData *epData     = (EpData *)argv[3];
    Qcst_EXTP0100_t *crgData = (Qcst_EXTP0100_t *)argv[4];
    char *formatName    = (char *)argv[5];

```

```

/*-----*/
/*
/* Kontrola, či sú odovzdávané údaje v očakávanom formáte.
/* Ak nie, vykonala sa zmena a tento ukončovací program sa musí
/* zaktualizovať, sa zmena prejavila. Pridajte vhodné protokolovanie chýb pre
/* váš návrh aplikácie.
/*

```

```

/*-----*/
    if (0 != memcmp(formatName, "EXTP0100", 8))
        abort();

```

```

/*-----*/
/*
/* Nastavenie údajov, ktoré sa odovzdajú obslužným rutinám pre výnimku
/* a zrušenie.
/*

```

```

/*-----*/
    hdldata.retCode      = retCode;
    hdldata.epData       = epData;
    hdldata.crgData      = crgData;
    hdldata.actionCode   = *actionCode;
    hdldata.role         = UnknownRole;
    hdldata.priorRole    = UnknownRole;
    _VBDY(); /* vnúťte zmenené premenné do domovského úložného priestoru */

```

```

/*-----*/
/*
/* Pre všetky výnimky umožnite obslužnú rutinu výnimky.
/*

```

```

/*-----*/
#pragma exception_handler(unexpectedExceptionHandler, hdldata, \
                          _C1_ALL, _C2_ALL, _CTLA_INVOKE )

/*-----*/
/*
/* Povolenie obnovy obslužnej rutiny pre zrušenie, ak sa táto úloha zruší.
/*
/*
/*-----*/
#pragma cancel_handler(cancelHandler, hdldata)

/*-----*/
/*
/* Rozbaľte rolu a pôvodnú rolu uzla, v ktorom sa vykonáva tento
/* ukončovací program. Ak API klastra alebo udalosť zmení doménu obnovy
/* (rolu uzla alebo stav členstva), do poľa Offset_Rcvy_Domain_Array sa
/* odovzdá posunutie novej domény obnovy a posunutie domény obnovy v
/* pôvodnom stave pred udalosťou API alebo klastra sa zadá do poľa
/* Offset_Prior_Rcvy_Domain_Array. Ak sa doména obnovy nezmení,
/* na adresovanie domény obnovy sa môže použiť len pole
/* Offset_Rcvy_Domain_Array.
/*
/*
/*-----*/
hdldata.role = getMyRole(crgData,
                        crgData->Offset_Rcvy_Domain_Array,
                        crgData->Number_Nodes_Rcvy_Domain);
if (crgData->Offset_Prior_Rcvy_Domain_Array)
    hdldata.priorRole =
        getMyRole(crgData,
                  crgData->Offset_Prior_Rcvy_Domain_Array,
                  crgData->Number_Nodes_Prior_Rcvy_Domain);
else
    hdldata.priorRole = hdldata.role;
_VBDY(); /* vnúťte zmenené premenné do domovského úložného priestoru */

/*-----*/
/*
/* Ak chcete vytlačiť informácie ladenia, umožnite nasledujúce.
/*
/*
/*-----*/
/*
printParms(*actionCode, hdldata.role, hdldata.priorRole, crgData,
           epData);
*/

/*-----*/
/*
/* Vykonajte správny zásah v závislosti od kódu akcie. Návratový kód
/* je nastavený na výsledok funkcie doAction().
/*
/*
/*-----*/
*retCode = doAction(*actionCode,
                    hdldata.role,
                    hdldata.priorRole,
                    crgData,
                    epData);

```

```

/*-----*/
/*
/* Úloha ukončovacieho programu sa ukončí, keď sa riadenie vráti do
/* operačného systému v tomto bode.
/*
/*-----*/
return;

#pragma disable_handler /* unexpectedExceptionHandler */
#pragma disable_handler /* cancelHandler */
} /* end main()

/*****
/*
/* Získajte rolu tohto príslušného uzla z niektorého zobrazenia
/* domény obnovy.
/*
/* API a udalosti klastra, ktoré do ukončovacieho programu zadajú
/* aktualizovanú a pôvodnú doménu pred obnovou, sú nasledujúce:
/* QcstAddNodeToRcvyDomain
/* QcstChangeClusterNodeEntry
/* QcstChangeClusterResourceGroup
/* QcstEndClusterNode (ukončovací uzol nezíska pôvodnú doménu)
/* QcstInitiateSwitchOver
/* QcstRemoveClusterNodeEntry (odstránený uzol nezíska pôvodnú doménu)
/* QcstRemoveNodeFromRcvyDomain
/* QcstStartClusterResourceGroup (len ak sú znova zoradené neaktívne
/* zložné uzly)
/* núdzové prepnutie zapríčiňujúce zlyhanie
/* uzol, ktorý sa znovu pripája do klastra
/* zlúčenie oddielov klastra
/*
/* Všetky ostatné API zadajú len aktualizovanú doménu obnovy.
/*
/*****
static int getMyRole(Qcst_EXTP0100_t *crgData, int offset, int count) {
    Qcst_Rcvy_Domain_Array1_t *nodeData;
    unsigned int iter = 0;

/*-----*/
/*
/* Za niektorých okolností operačný systém nemusí vedieť určiť ID tohto
/* uzla a zadá hodnotu *NONE. Takáto okolnosť môže nastať napríklad
/* vtedy, keď služby prostriedkov klastra nie sú v uzle aktívne
/* a používa sa príkaz príkazového riadka DLTCRG.
/*
/*-----*/
    if (0 == memcmp(crgData->This_Nodes_ID, QcstNone,
        sizeof(Qcst_Node_Id_t)))
        return UnknownRole;

/*-----*/
/*
/* Posuňte smerník na prvý prvok v poli domény obnovy.
/*
/*-----*/
    nodeData = (Qcst_Rcvy_Domain_Array1_t *)((char *)crgData + offset);
/*-----*/
/*

```

```

    /* Nájdi tento uzol v poli domény obnovy. Ak bol uzol pridaný pomocou */
    /* funkcie Pridať uzol do API domény obnovy, nebude sa nachádzať */
    /* pôvodnej domény pred obnovou. */
/*
*/

/*-----*/
while ( 0 != memcmp(crgData->This_Nodes_ID,
                  nodeData->Node_ID,
                  sizeof(Qcst_Node_Id_t)

                &&
                iter &lt; count
            ) {
    nodeData++;
    iter++;
}

if (iter &lt; count)
    return nodeData->Node_Role;
else
    return UnknownRole;
} /* end getMyRole() */

/*****
*/
/* Zavolajte správnu funkciu v závislosti od kódu akcie klastra. Funkcia */
/* doAction() bola oddelená od funkcie main() za účelom objasnenia */
/* príkladu. Pozrite si vysvetlivky pre každú zavolanú funkciu, kde */
/* nájdete informácie o príslušných akciách klastra. */
/* */
/* Každý kód akcie je rozdelený na osobitné funkcie len kvôli lepšiemu */
/* vysvetleniu tohto príkladu. Pre konkrétny ukončovací program môžu */
/* niektoré kódy vykonať rovnakú funkciu. V tom prípade môže tá istá */
/* funkcia spracovať viac kódov akcie. */
/* */
/*****
static int doAction(int actionCode,
                  int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/
/*
*/
/* Pre kódy akcie, známe tomuto ukončovaciemu programu, zavolajte */
/* funkciu, ktorá pre ne vykoná danú prácu. */
/*
*/

/*-----*/

if (actionCode &lt;= MaxAc )
    return (*fcn[actionCode]) (role, priorRole, crgData, epData);
else

/*-----*/
/*
*/
/* Spoločnosť IBM v novom vydaní operačného systému definovala nový */
/* kód akcie a tento ukončovací program ešte nebol aktualizovaný na */
/* jeho spracovanie. Zatiaľ použite predvolenú akciu. */
/*
*/

/*-----*/
return newActionCode(role, priorRole, crgData, epData);
} /* end doAction() */

```



```

/*****/
/*
/* Kód akcie = QcstCrgAcInitialize
/*
/* Bolo zavolané API QcstCreateClusterResourceGroup. Vytvára sa nový
/* objekt skupiny prostriedkov klastra.
/*
/* Nezabudnite vykonať:
/* - Skontrolujte, či sa aplikačný program a všetky priradené objekty
/* nachádzajú v primárnom a záložnom uzle. Ak sa tu objekty
/* nenachádzajú, mali by ste odoslať chybovú alebo upozorňujúcu správu
/*
/* - Skontrolujte, či sa požadované údaje alebo CRG zariadenia
/* domény obnovy.
/* - Vykonajte potrebné nastavenie, nevyhnutné na
/* spustenie aplikácie v primárnom alebo záložnom uzle.
/* - Ak má toto CRG povolené používať API QcstDistributeInformation,
/* teraz je možné vytvoriť užívateľský front potrebný pre toto API.
/*
/*****/
static int createCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

    return QcstSuccessful;
} /* end createCrg()

/*****/
/*
/* Kód akcie = QcstCrgAcStart
/*
/* Bolo zavolané API QcstStartClusterResourceGroup. Spúšťa sa skupina
/* prostriedkov klastra.
/* Bolo zavolané API QcstInitiateSwitchOver a toto je druhý kód akcie
/* zadaný pre ukončovací program.
/* Vyskytla sa udalosť núdzového prepnutia a toto je druhý kód akcie,
/* zadaný pre ukončovací program.
/*
/* Počas kontroly, či sú aktívne všetky závislé CRG, sa použije maximálny
/* čas čakania. Tento čas je krátky v prípade, ak bolo CRG spustené kvôli
/* API QcstStartClusterResourceGroup. Tento čas je dlhší v prípade, ak ide
/* o núdzové prepnutie alebo prepnutie. Ak sa vykonáva núdzové prepnutie
/* alebo prepnutie, môže trvať dlhšie, kým budú údaje alebo CRG zariadenia
/* pripravené, preto je čas čakania dlhší. Ak sa používa API pre spustenie
/* CRG, závislé CRG by sa už mali spustiť. V opačnom prípade sa mohla
/* vyskytnúť chyba, CRG boli nesprávne spustené a podobne. Vtedy už nebude
/* potrebné čakať dlho.
/*
/* Nezabudnite vykonať:
/* - Ak je táto rola uzla primárna, aplikácia by sa mala spustiť.
/* Tento ukončovací program by mal zavolať aplikáciu, aby sa
/* vykonávala v tej istej úlohe alebo by mal monitorovať všetky úlohy,
/* ktoré spustil, aby vedel, kedy bude úloha aplikácie ukončená.
/* Najjednoduchší spôsob, ako spustiť aplikáciu v tejto úlohe, je ten,
/* že ju zavoláte.
/* Služby prostriedkov klastra neočakávajú návrat tohto ukončovacieho
/* programu, kým sa aplikácia neskončí.
/* - Podľa potreby spustíte priradené podsystémy, úlohy servera, atď.
/* - Presvedčite sa, či CRG požadovaných údajov majú aktívny stav
/* vo všetkých uzloch domény obnovy.
/*
/*****/
static int startCrg(int role,
                   int doesNotApply,
                   Qcst_EXTP0100_t *crgData,

```

```

        EpData *epData) {

    unsigned int maxWaitTime;

    /* Ak je tento uzol primárny, spustite aplikáciu */
    if (role == QcstPrimaryNodeRole) {

/*-----*/
        /*
            /* Zistite, či sú pripravené všetky CRG, na ktorých je toto CRG */
            /* aplikácie závislé. Ak kontrola zlyhá, vráťte sa z kódu akcie */
            /* pre spustenie. Služby prostriedkov klastra zmenia stav CRG na */
            /* Neaktívny. */
        */

/*-----*/
        if (crgData->Cluster_Resource_Group_Status == QcstCrgStartCrgPending)
            maxWaitTime = MaxStartCrgWaitSeconds;
        else
            maxWaitTime = MaxWaitSeconds;
        if (QcstSuccessful != checkDependCrgDataArea(maxWaitTime))
            return QcstSuccessful;

/*-----*/
        /*
            /* Tesne pred spustením aplikácie zaktualizujte údajovú oblasť, */
            /* aby sa zobrazilo, že aplikácia je spustená. */
        */

/*-----*/
        setApp1CrgDataArea(App1_Running);

/*-----*/
        /*
            /* Zadajte logiku, ktorá zavolá aplikáciu. Očakáva sa, že */
            /* riadenie sa nevráti, kým niečo nezapríčiní ukončenie aplikácie. */
            /* Normálne vrátenie z ukončovacieho programu, zrušenie úlohy alebo */
            /* výskyt nespracovanej výnimky. V opise funkcie cancelHandler() */
            /* nájdete niektoré bežné spôsoby zrušenia tejto úlohy. */
        */

/*-----*/

/*-----*/
        /*
            /* Po normálnom ukončení aplikácie zaktualizujte údajovú oblasť, */
            /* aby sa zobrazilo, že aplikácia sa už nevykonáva. */
        */

/*-----*/
        setApp1CrgDataArea(App1_Ended);
    }
    else

/*-----*/
        /*
            /* V záložných alebo replikovaných uzloch označte stav aplikácie */
            /* v údajovej oblasti ako zastavený. */
        */

/*-----*/

```

```

        setApplCrgDataArea(Appl_Ended);

        return QcstSuccessful;
    } /* end startCrg() */
    /**
     *
     * Kód akcie = QcstCrgAcRestart
     *
     * Predošlé zavolanie ukončovacieho programu zlyhalo a nastavilo
     * návratový kód na QcstFailWithRestart, alebo zlyhalo kvôli výnimke a
     * výnimka sa mohla preniesť navrch zásobníka volaní. V ani jednom
     * prípade nebol dosiahnutý maximálny počet reštartovaní ukončovacieho
     * programu.
     *
     * Tento kód akcie sa zadáva len pre ukončovacie programy CRG aplikácie,
     * ktoré boli zavolané kódom akcie spustenia.
     */
    /**
     *
     * static int restartCrg(int role,
     *                       int doesNotApply,
     *                       Qcst_EXTP0100_t *crgData,
     *                       EpData *epData) {

    /*-----*/
    /*
     * Vykonajte ľubovoľnú logiku, ktorá môže byť potrebná počas
     * reštartovania aplikácie po zlyhaní. Potom zavolajte funkciu startCrg()
     * pre vykonanie funkcií spustenia.
     */
    /*-----*/

        return startCrg(role, doesNotApply, crgData, epData);
    } /* end restartCrg() */

    /**
     *
     * Kód akcie = QcstCrgAcEnd
     *
     * Kód akcie ukončenia bol použitý z niektorej z nasledujúcich príčin:
     * - Bolo zavolané API QcstEndClusterResourceGroup.
     * - Klastor bol rozdelený na oddiely a tento uzol je v sekundárnom
     * oddiele. Kód akcie ukončenia sa použije bez ohľadu na to, či bolo
     * CRG aktívne alebo neaktívne. Budú zadané aj závislé údaje kódu
     * akcie QcstPartitionFailure.
     * - Aplikácia bola ukončená. Budú zadané aj závislé údaje kódu
     * akcie QcstResourceEnd. Pre všetky uzly v doméne obnovy (vrátane
     * primárnych) bude vykonaný rovnaký kód akcie.
     * - Úloha CRG bola zrušená. Ukončovací program v tomto uzle bude
     * zavolaný kódom akcie ukončenia. QcstMemberFailure bude zadaný
     * ako závislý údaj kódu akcie.
     *
     *
     *
     *
     * Nezapadnite vykonať:
     * - Ak je CRG aktívne, úloha vykonávajúca aplikáciu sa zruší
     * a adresa IP na prebratie sa ukončí PO zrušení ukončovacieho
     * programu.
     * - Ak boli spustené podsystémy alebo úlohy servera, ako výsledok kódu
     * akcie QcstCrgAcStart, ukončíte ich tu. Alebo skonsolidujte všetky
     * logiky na ukončenie aplikácie vo funkcii cancelHandler(), pretože
     * bude vyvolaná pre všetky rozhrania API služieb prostriedkov
     * klastra, ktoré musia ukončiť aplikáciu v aktuálnom primárnom uzle.
     */

```

```

/*                                                                 */
/*****                                                                 */
static int endCrg(int role,
                 int priorRole,
                 Qcst_EXTP0100_t *crgData,
                 EpData *epData) {

/*-----*/
/*                                                                 */
/*   /* Ak sa aplikácia vykonáva v tomto uzle, ukončíte ju.          */
/*                                                                 */
/*-----*/
    endApplication(QcstCrgAcRemoveNode, role, priorRole, crgData,
                  epData);

    return QcstSuccessful;
} /* end endCrg()                                                                 */

/*****                                                                 */
/*                                                                 */
/* Kód akcie = QcstCrgAcVerificationPhase                            */
/*                                                                 */
/* Použil sa kód akcie overenia fázy, ktorý umožní ukončovaciemu programu */
/* vykonať overovanie pred spracovaním požadovanej funkcie identifikovanej */
/* závislými údajmi kódu akcie. Ak ukončovací program zistí, že požadovaná */
/* funkcia nemôže byť vykonaná, mal by vrátiť                          */
/* QcstFailWithOutRestart.                                           */
/*                                                                 */
/*                                                                 */
/* POZNÁMKA: Ukončovací program NEBUDE zavolaný kódom akcie Späť.   */
/*                                                                 */
/*****                                                                 */
static int verifyPhase(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

/*-----*/
/*                                                                 */
/*   /* Vykonajte overenie                                          */
/*                                                                 */
/*-----*/
    if (crgData->Action_Code_Dependent_Data == QcstDltCrg) {
        /* do verification */
        /* if ( fail ) */
        /* return QcstFailWithOutRestart */
    }

    return QcstSuccessful;
} /* end verifyPhase()                                                                 */

/*****                                                                 */
/*                                                                 */
/* Kód akcie = QcstCrgAcDelete                                       */
/*                                                                 */
/* Bolo zavolané API QcstDeleteClusterResourceGroup alebo           */
/* QcstDeleteCluster. Kým budú služby prostriedkov klastra aktívne, vymaže */
/* sa skupina prostriedkov klastra.                                  */
/* Ak bolo použité API QcstDeleteCluster, zadajú sa závislé údaje kódu */
/* akcie QcstDltCluster.                                            */
/* Ak bolo použité API QcstDeleteCluster a CRG je aktívne, po spracovaní */

```

```

/* kódu akcie vymazania sa zruší úloha ukončovacieho programu, ktorá je */
/* stále aktívna pre kód akcie spustenia. */
/* */
/* Nezabudnite vykonať: */
/* - Vymažte aplikačné programy a objekty z uzlov, v ktorých už nie sú */
/* potrebné, napríklad zo záložných uzlov. Dajte si pozor na potrebné */
/* zásahy počas vymazávania objektov aplikácie, pretože ak určité */
/* scenáre vyžadujú ponechanie objektov aplikácií vo všetkých uzloch, */
/* vymaže sa CRG. */
/* */
/*****/
static int deleteCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

    return QcstSuccessful;
} /* end deleteCrg() */
/*****/
/* */
/* Kód akcie = QcstCrgAcReJoin */
/* */
/* Vyskytla sa jedna z nasledujúcich situácií - */
/* 1. Problém, ktorý zapríčinil rozdelenie klastra do oddielov bol */
/* opravený a dva oddiely sa zlučujú späť, aby vytvorili jeden */
/* klaster. Budú zadané údaje závislé od kódu akcie QcstMerge. */
/* */
/* 2. Uzol, ktorý predtým zlyhal alebo bol ukončený, znova spustil služby */
/* prostriedkov klastra a pripája sa do klastra. Budú zadané údaje */
/* závislé od kódu akcie QcstJoin. */
/* 3. Úloha CRG v určitom uzle, ktorá mohla byť zrušená alebo */
/* ukončená, sa opakovane spustila. Budú zadané údaje závislé od kódu */
/* akcie QcstJoin. */
/* */
/* Nezabudnite vykonať: */
/* - Ak aplikácia replikuje informácie o stave aplikácie do iných uzlov */
/* počas jej vykonávania, tieto informácie o stave sa budú musieť */
/* resynchronizovať s pripájanými uzlami, ak je CRG aktívne. */
/* prepnuté, a či sú aktívne. */
/* - V pripájaných uzloch skontrolujte chýbajúce objekty aplikácie. */
/* - Ubezpečte sa, či sú požadované CRG údajov v pripájaných uzloch. */
/* - Ak je CRG aplikácie aktívny, presvedčite sa, či sú aktívne aj */
/* prepnuté, a či sú aktívne. */
/* */
/*****/
static int memberIsJoining(int role,
                           int priorRole,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) {

/*-----*/
/* */
/* Presvedčite sa, že stav údajovej oblasti v tomto uzle začal */
/* označovač, že aplikácia nie je spustená, ak uzol nie je primárny. */
/* */
/*-----*/

    if (role != QcstPrimaryNodeRole) {
        setApplCrgDataArea(Appl_Ended);
    }

/*-----*/
/* */
/* Ak sa ku klastru znova pripája jeden uzol, môžete vykonať postupnosť */
/* akcií. V prípade, že sa uzly v klastru, ktorý bol rozdelený */

```

```

/* do oddielov, zlučujú späť, môžete vykonať inú postupnosť akcií. */
/* */

/*-----*/
if (crgData->Action_Code_Dependent_Data == QcstJoin) {
    /* Vykonajte akcie pre pripájanie uzlov. */
}
else {
    /* Vykonajte akcie pre zlúčenie oddielov. */
}

return QcstSuccessful;
} /* end memberIsJoining() */

/*****/
/* */
/* Kód akcie = QcstCrgAcFailover */
/* */
/* Služby prostriedkov klastra v určitých uzloch zlyhali alebo boli */
/* ukončené pre túto skupinu prostriedkov klastra. Zadal sa kód akcie */
/* núdzového prepnutia bez ohľadu na to, či je CRG aktívne alebo */
/* neaktívne. Núdzové prepnutie je možné vykonať z niekoľkých príčin: */
/* */
/* - operátor v uzle zrušil úlohu CRG. Odovzdá sa kód akcie závislý od */
/* kódu akcie QcstMemberFailure. */
/* - v uzle boli ukončené služby prostriedkov klastra (napríklad */
/* podsystem QSYSWRK bol ukončený, pričom CRS boli stále aktívne). */
/* Budú zadané údaje závislé od kódu akcie QcstNodeFailure. */
/* - aplikácia pre CRG aplikácie zlyhala v primárnom uzle a nebolo možné */
/* ju tu reštartovať. CRG je aktívne. */
/* Budú zadané údaje závislé od kódu akcie QcstApplFailure. */
/* - uzol zlyhal (napríklad výpadok elektriny). Budú zadané údaje */
/* závislé od kódu akcie QcstNodeFailure. */
/* - Klaster bol rozdelený z dôvodu zlyhania komunikácie, napríklad */
/* zlyhanie komunikačnej linky alebo siete LAN. Zadá sa akcia */
/* núdzového prepnutia pre uzly domény obnovy v majoritnom oddiele. */
/* Pre uzly v minoritnom oddiele pozrite kód akcie ukončenia. Budú */
/* zadané údaje závislé od kódu akcie QcstPartitionFailure. */
/* - Uzol v doméne obnovy CRG sa ukončuje s API */
/* QcstEndClusterNode. Pre uzol, ktorý sa ukončuje bude vykonaný */
/* kód akcie ukončenia uzla. Pre všetky ostatné uzly v doméne obnovy */
/* sa vykoná kód akcie núdzového prepnutia. Pre tento kód akcie budú */
/* zadané údaje závislé od kódu akcie QcstEndNode. */
/* - Odstraňuje sa aktívny uzol domény obnovy pre aktívne CRG z klastra */
/* s API QcstRemoveClusterNodeEntry. Budú zadané údaje závislé od */
/* kódu akcie QcstRemoveNode. Ak sa odstráni neaktívny uzol */
/* pre aktívne CRG, alebo ak je CRG neaktívne, zadá sa kód akcie */
/* pre odstránenie uzla. */
/* */
/* Zavolá sa ukončovací program bez ohľadu na to, či je CRG aktívne */
/* alebo nie. Ukončovací program možno nevykonať žiadnu činnosť, ak CRG */
/* prepnuté, a či sú aktívne. */
/* */
/* Ak je CRG aktívne a opúšťajúci člen bol primárnym uzlom, vykonajte */
/* funkcie potrebné pre núdzové prepnutie na nový primárny uzol. */
/* */
/* Pole Action_Code_Dependent_Data môžete použiť, ak chcete zistiť, či: */
/* - k zlyhaniu došlo kvôli problému, ktorý zapríčinil rozdelenie */
/* klastra (týka sa to všetkých CRG, ktoré mali rozdelené uzly v */
/* doméne obnovy) */
/* - uzol zlyhal alebo ukončil služby prostriedkov klastra v uzle (týka */
/* sa to všetkých CRG, ktoré mali zlyhaný alebo ukončený uzol v doméne */
/* obnovy) */
/* - bolo ovplyvnené len jedno CRG (napríklad v uzle bola zrušená jedna */
/* úloha CRG alebo zlyhala jedna aplikácia) */
/* */

```

```

/* */
/* Nezapadnite vykonať: */
/* - Pripravte nový primárny uzol, aby bolo možné spustiť aplikáciu. */
/* - Aplikácia by sa NEMALA spustiť teraz. Ukončovací program bude */
/* zavolaný znova s kódom akcie QcstCrgAcStart v prípade, ak bolo CRG */
/* aktívne, keď sa vyskytla chyba. */
/* - Ak je CRG aplikácie aktívny, presvedčite sa, či sú aktívne aj */
/* prepnuté, a či sú aktívne. */
/* */
/*****/
static int memberIsLeaving(int role,
                           int priorRole,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) {

/*-----*/
/* */
/* Ak je CRG aktívne, vykonajte núdzové prepnutie. Inak nevykonajte nič. */
/* */
/*-----*/
    if (crgData->Original_Cluster_Res_Grp_Stat == QcstCrgActive) {

/*-----*/
/* */
/* CRG je aktívne. Zistite, či sa zmenila rola tohto uzla, a tento */
/* bude odteraz novým primárnym uzlom. */
/* */
/*-----*/

        if (priorRole != role && role == QcstPrimaryNodeRole) {

/*-----*/
/* */
/* Tento uzol nebol primárny, teraz však už je. Vykonajte akcie */
/* núdzového prepnutia, ale nespustite ešte aplikáciu, pretože tento */
/* ukončovací program bude znova zavolaný s kódom akcie spustenia. */
/* */
/*-----*/

/*-----*/
/* */
/* Presvedčite sa, že stav údajovej oblasti v tomto uzle začal */
/* zobrazovať, že aplikácia nie je spustená. */
/* */
/*-----*/

            setApp1CrgDataArea(App1_Ended);

/*-----*/
/* */
/* Ak aplikácia nemá čo vykonať v kóde akcie spustenia a stane sa */
/* aktívnou hneď po aktivácii adresy IP na preberatie, vtedy by mal */
/* tento kód zostať bez poznámok. Tento kód určí, či sú pripravené */
/* všetky CRG, na ktorých je CRG tejto aplikácie závislé. */
/* Ak táto kontrola zlyhá, vráťte zlyhanie z kódu akcie. */
/* */
/* */
/*-----*/
}
}
}

```



```

/*      if (QcstSuccessful != checkDependCrgDataArea(MaxWaitSeconds))      */
/*      return QcstFailWithOutRestart;                                     */
}
}

return QcstSuccessful;
} /* end memberIsLeaving() */

/*****
/*
/* Kód akcie = QcstCrgAcSwitchover
/*
/* Bolo zavolané API QcstInitiateSwitchOver. Prvý záložný uzol v doméne
/* obnovy skupiny prostriedkov klastra sa preberá ako primárny uzol.
/* Aktuálny primárny uzol urobil poslednú zálohu.
/*
/* Nezabudnite vykonať:
/* - Pripravte nový primárny uzol, aby bolo možné spustiť aplikáciu.
/* - Aplikácia by sa NEMALA spustiť teraz. Ukončovací program bude
/* zavolaný znova s kódom akcie QcstCrgAcStart.
/* - Úloha vykonávajúca aplikáciu sa zrušila a adresa IP pre prevzatie
/* sa ukončila pred zavolaním ukončovacieho programu v aktuálnom
/* primárnom uzle.
/* - Presvedčite sa, či boli CRG požadovaných údajov alebo zariadenia
/* prepnuté, a či sú aktívne.
/*
/*
/*****
static int switchPrimary(int role,
                        int priorRole,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

/*-----*/
/*
/* Zistite, či je tento uzol starým primárnym uzlom.
/*
/*-----*/
if (priorRole == QcstPrimaryNodeRole) {

/*-----*/
/*
/* Pred prepnutím vykonajte všetky potrebné akcie na vyčistenie
/* starého primárneho uzla. Nezabudnite, že úloha, ktorá spustila
/* ukončovací program, ktorý spustil aplikáciu, sa už zrušila.
/*
/* Ako príklad môže poslúžiť vyčistenie všetkých procesov so zámkami
/* ne databáze. Toto sa mohlo vykonať obslužným programom na zrušenie
/* aplikácie, ak bol vyvolaný.
/*-----*/
}

/*-----*/
/*
/* Toto nie je starý primárny uzol. Zistite, či je nový primárny uzol.
/*
/*-----*/
else if (role == QcstPrimaryNodeRole) {

/*-----*/
/*
/* Vykonajte všetky potrebné akcie v novom primárnom uzle predtým, ako */

```

```

/* sa aplikácia spustí kódom akcie QcstCrgAcStart. */
/*
/*-----*/

/*-----*/
/*
/* Presvedčite sa, či stav údajovej oblasti v tomto uzle začal */
/* zobrazovať, že aplikácia nie je spustená. */
/*
/*-----*/
        setApplCrgDataArea(Appl_Ended);

/*-----*/
/*
/* Ak aplikácia nemá čo vykonať v kóde akcie spustenia a stane sa */
/* aktívnou hneď po aktivácii adresy IP na preberatie, vtedy by mal */
/* tento kód zostať bez poznámok. Tento kód určí, či sú pripravené */
/* všetky CRG, na ktorých je CRG tejto aplikácie závislé. */
/* Ak táto kontrola zlyhá, vráťte zlyhanie z kódu akcie. */
/*
/*
/*-----*/
/*
/* if (QcstSuccessful != checkDependCrgDataArea(MaxWaitSeconds)) */
/* return QcstFailWithOutRestart; */
}
else {

/*-----*/
/*
/* Tento uzol je jedným z ďalších záložných uzlov, alebo je to */
/* replikovaný uzol. Ak tieto uzly musia niečo vykonať, urobte to */
/* tu. V opačnom prípade odstráňte tento ďalší blok. */
/*
/*-----*/

/*-----*/
/*
/* Presvedčite sa, či stav údajovej oblasti v tomto uzle začal */
/* zobrazovať, že aplikácia nie je spustená. */
/*
/*-----*/
        setApplCrgDataArea(Appl_Ended);
}

return QcstSuccessful;
} /* end switchPrimary() */

/*****
/*
/* Kód akcie = QcstCrgAcAddNode
/*
/* Bolo zavolané API QcstAddNodeToRcvyDomain. Do domény obnovy skupiny
/* prostriedkov klastra sa pridáva nový uzol.
/*
/* Nezapudnite vykonať:
/* - Do domény obnovy sa pridáva nový uzol. Pozrite si poznámky
/* akcií pre funkciu createCrg().
*/

```

```

/* - Ak má toto CRG povolené používať API QcstDistributeInformation, */
/* teraz je možné vytvoriť užívateľský front potrebný pre toto API. */
/* */
/******
static int addNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/

/* */
/* Zistite, či toto je uzol, ktorý sa pridáva. */
/* */

/*-----*/

    if (0 == memcmp(&crgData->This_Nodes_ID,
                  &crgData->Changing_Node_ID,
                  sizeof(Qcst_Node_Id_t)))
    {

/*-----*/

        /* */
        /* Nastavte stav údajovej oblasti v tomto novom uzle. */
        /* */

/*-----*/

        setApplCrgDataArea(Appl_Ended);

/*-----*/

        /* */
        /* Vytvorte front potrebný pre API distribúcie informácií. */
        /* */

/*-----*/

        if (0 == memcmp(&crgData->DI_Queue_Name,
                      Nulls,
                      sizeof(crgData->DI_Queue_Name)))
        {
        }

        return QcstSuccessful;
    } /* end addNode() */
/******
/* */
/* Kód akcie = QcstCrgAcRemoveNode */
/* */
/* Bolo zavolané API QcstRemoveNodeFromRcvyDomain alebo */
/* QcstRemoveClusterNodeEntry. Odstraňuje sa uzol z domény obnovy skupiny */
/* klastra alebo sa odstraňuje z klastra úplne. */
/* */
/* */
/* Tento kód akcie sa vykoná pre nasledujúce situácie: */
/* Pre API QcstRemoveClusterNodeEntry: */
/* - Ak je odstránený uzol aktívny a CRG je neaktívne, pre všetky uzly */
/* v doméne obnovy, vrátane odstraňovaného uzla, sa vykoná tento */
/* kód akcie. Pre uzly, ktoré NIE SÚ odstránené sa zadajú údaje */
/* závislé od kódu akcie QcstNodeFailure. */
/* - Ak je odstránený uzol aktívny a CRG je aktívne, pre odstraňovaný */
/* uzol sa vykoná kód akcie odstránenia uzla. Pre všetky uzly v */

```

```

/* doméne obnovy sa vykoná kód akcie núdzového prepnutia a zadajú sa */
/* údaje závislé od kódu akcie QcstNodeFailure. */
/* - Ak odstraňovaný uzol nie je v klastri aktívny, pre všetky uzly */
/* v doméne obnovy sa vykoná tento kód akcie. */
/* Pre API QcstRemoveNodeFromRcvyDomain: */
/* - Pre všetky uzly sa vykoná kód akcie odstránenia uzla, bez ohľadu */
/* na to, či je CRG aktívne. Budú zadané aj údaje závislé od kódu */
/* akcie QcstRmvRcvyDmnNode. */
/*
/* Nezabudnite vykonať:
/* - Možno budete chcieť vyčistiť odstránený uzol vymazaním objektov,
/* ktoré tu už viac nepotrebuje.
/* - Úloha vykonávajúca aplikáciu sa zrušila a adresa IP pre prevzatie
/* bola ukončená po zavolaní ukončovacieho programu, ak toto je
/* primárny uzol a CRG je aktívne.
/* - Ak boli spustené podsystémy alebo úlohy servera, ako výsledok kódu
/* akcie QcstCrgAcStart, ukončíte ich tu. Alebo skonsolidujte všetky
/* logiky na ukončenie aplikácie vo funkcii cancelHandler(), pretože
/* bude vyvolaná pre všetky rozhrania API služieb prostriedkov
/* klastra, ktoré musia ukončiť aplikáciu v aktuálnom primárnom uzle.
/*
/*****/
static int rmvNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/
/*
/* Zistite, či toto je uzol, ktorý sa odstraňuje.
/*
/*-----*/

    if (0 == memcmp(&crgData->This_Nodes_ID,
                   &crgData->Changing_Node_ID,
                   sizeof(Qcst_Node_Id_t)))
    {
/*-----*/
/*
/* Ak sa aplikácia vykonáva v tomto uzle, ukončíte ju.
/*
/*-----*/
        endApplication(QcstCrgAcRemoveNode, role, priorRole, crgData,
                       epData);
    }
    return QcstSuccessful;
} /* end rmvNode */

/*****/
/*
/* Kód akcie = QcstCrgAcChange
/*
/* Bolo zavolané API QcstChangeClusterResourceGroup. Zmenia sa niektoré
/* atribúty alebo informácie uložené v objekte CRG.
/* Zapamätajte si, že nie všetky zmeny objektu CRG zapríčinia zavolanie
/* ukončovacieho programu. Vo verzii V5R1M0 len nasledujúce zmeny spôsobia
/* zavolanie ukončovacieho programu:
/* - zmena aktuálnej domény obnovy
/* - zmena preferovanej domény obnovy
/*

```

```

/* Ak sa vykoná niektorá z vyššie uvedených zmien a následne sa hodnota */
/* ukončovacieho programu zmení na *NONE, tento program nebude zavolaný. */
/* */
/* Nezabudnite vykonať: */
/* - Žiadne akcie, pokiaľ zmeny domény obnovy neovplyvnia informácie */
/* alebo procesy pre túto skupinu prostriedkov klastra. Zapamätajte */
/* si, že primárny uzol sa nedá zmeniť s API */
/* QcstChangeClusterResourceGroup, ak je CRG aktívne. */
/* */
/*****/
static int chgCrg(int role,
                 int priorRole,
                 Qcst_EXTP0100_t *crgData,
                 EpData *epData) {

    return QcstSuccessful;
} /* end chgCrg() */

/*****/
/* */
/* Kód akcie = QcstCrgAcDeleteCommand */
/* */
/* Z príkazového riadka bol zavolaný príkaz DLTCRG (Delete Cluster */
/* Resource Group) na vymazanie objektu skupiny prostriedkov klastra, bolo */
/* zavolané API QcstDeleteCluster alebo API QcstRemoveClusterNodeEntry. */
/* V každom prípade, služby prostriedkov klastra nie sú aktívne v uzle */
/* klastra, z ktorého bol zavolaný príkaz alebo API. Táto funkcia preto */
/* nie je distribuovaná do celého klastra, ale vyskytne sa len v tom uzle, */
/* z ktorého bol zavolaný príkaz príkazového riadka alebo API. */
/* */
/* Ak bolo použité API QcstDeleteCluster, zadajú sa závislé údaje kódu */
/* akcie QcstDltCluster. */
/* */
/* Pozrite si poznámky k funkcii deleteCrg() */
/* */
/*****/
static int deleteCrgWithCmd(int role,
                            int doesNotApply,
                            Qcst_EXTP0100_t *crgData,
                            EpData *epData) {

    return QcstSuccessful;
} /* end deleteCrgWithCmd() */

/*****/
/* */
/* Kód akcie = QcstCrgEndNode */
/* */
/* Bolo zavolané API QcstEndClusterNode alebo bola zrušená úloha CRG. */
/* */
/* Kód akcie QcstCrgEndNode sa zadá do ukončovacieho programu len */
/* v uzle, ktorý sa ukončuje alebo kde sa zrušila úloha CRG. V uzle, kde */
/* sa zrušila úloha služieb prostriedkov klastra sa odovzdajú údaje závislé od */
/* od kódu akcie QcstMemberFailure. */
/* Keď sa služby prostriedkov klastra ukončia v tomto uzle alebo sa ukončí */
/* úloha CRG, zapríčiní to, že sa pre všetky ostatné uzly v klastri vykoná */
/* proces núdzového prepnutia. Pre všetky ostatné uzly sa zadá kód akcie */
/* QcstCrgAcFailover. Pre tieto uzly sa zadajú údaje závislé od kódu akcie */
/* QcstMemberFailure, ak sa zruší úloha CRG, alebo QcstNodeFailure, ak */
/* sa uzol ukončí. */
/* */
/* Nezabudnite vykonať: */
/* - Úloha vykonávajúca aplikáciu sa zrušila a adresa IP pre prevzatie */
/* bola ukončená po zavolaní ukončovacieho programu, ak toto je */
/* primárny uzol a CRG je aktívne. */

```

```

/* - Ak boli spustené podsystémy alebo úlohy servera, ako výsledok kódu */
/* kódu akcie QcstCrgAcStart, ukončíte ich v tomto bode. */
/* */
/*****/
static int endNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/
/*
/* Ak sa aplikácia vykonáva v tomto uzle, ukončíte ju. */
/* */
/*-----*/
    endApplication(QcstCrgEndNode, role, priorRole, crgData, epData);

    return QcstSuccessful;
} /* end endNode() */

/*****/
/*
/* Kód akcie = QcstCrgAcChgNodeStatus */
/* */
/* Bolo zavolané API QcstChangeClusterNodeEntry. Stav uzla sa zmení */
/* na zlyhaný. Toto API sa používa na informovanie služieb prostriedkov */
/* klastra, že uzol sa nerozdelil na oddiely, ale skutočne zlyhal. */
/* */
/* Nezabudnite vykonať: */
/* - Ukončovací program bol predtým zavolaný s kódom akcie */
/* QcstCrgAcEnd, ak bolo CRG aktívne, alebo s kódom akcie */
/* QcstCrgAcFailover, ak bolo CRG neaktívne, pretože sa pre služby */
/* prostriedkov klastra javilo, že bol klaster rozdelený. Užívateľ */
/* teraz zadáva pre služby prostriedkov klastra informáciu, že uzol */
/* skutočne zlyhal a nebol rozdelený. Ukončovací program musí vykonať */
/* akciu len v prípade, ak predtým vykonal akciu, ktorú teraz musí */
/* zmeniť, aby mohol potvrdiť zlyhanie uzla. */
/* */
/*****/
static int chgNodeStatus(int role,
                        int priorRole,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

    return QcstSuccessful;
} /* end chgNodeStatus() */

/*****/
/*
/* Kód akcie = QcstCrgAcCancelFailover */
/* */
/* Služby prostriedkov klastra v primárnom uzle zlyhali alebo boli pre */
/* túto skupinu prostriedkov klastra ukončené. Bola odoslaná správa do */
/* frontu správ pre núdzové prepnutie definované pre CRG, a táto správa */
/* zapríčinila zrušenie núdzového prepnutia. Toto zmení stav CRG na */
/* neaktívny a ponechá primárny uzol ako primárny. */
/* */
/* Nezabudnite vykonať: */
/* - Primárny uzol sa už nepodieľa na činnostiach klastra. */
/* Problém, ktorý zapríčinil zlyhanie primárneho uzla by sa mal */
/* opraviť, aby bolo možné znova spustiť CRG. */
/* */
/*****/
static int cancelFailover(int role,

```

```

        int priorRole,
        Qcst_EXTP0100_t *crgData,
        EpData *epData) {

    return QcstSuccessful;
} /* end cancelFailover() */

/*****/
/*
/* Kód akcie = ukončovací program ho ešte nepozná
/*
/* Pre tento ukončovací program bol zadaný nový kód akcie. Môže sa to stať
/* po inštalácii nového vydania i5/OS a zavolaní niektorého nového API
/* klastra alebo pri výskyte novej udalosti klastra. Logika v tomto
/* ukonč. programe ešte nebola aktualizovaná, aby poznala nový kód akcie.
/*
/* Pre nový kód akcie je možné použiť dve rôzne stratégie. Správna
/* stratégia závisí od druhov činností, ktoré tento príslušný ukončovací
/* program vykonáva pre aplikáciu.
/*
/* Jednou stratégiou je nevykonať nič a vrátiť úspešný návratový kód.
/* Toto umožní kompletizáciu nového API alebo udalosti klastra. Umožní to
/* vykonanie funkcie aj napriek tomu, že tento ukončovací program
/* nerozumie novému kódu akcie. Je tu však riziko, že ukončovací program
/* nevykoná činnosť, ktorú mal vykonať. Prínajmenšom budete možno chcieť
/* zaprotokolovať určitý druh chybového hlásenia o tom, čo sa stalo.
/* Potom môžete programovaním prešetriť a aktualizovať ukončovací program.
/*
/* Opačnou stratégiou je vrátiť chybový návratový kód, napríklad
/* QcstFailWithRestart. Samozrejme to znamená, že nové API alebo udalosť
/* klastra nebude možné použiť, kým nebude ukonč. program zaktualizovaný
/* pre nový kód akcie. Aj v tomto prípade sa oplatí zaprotokolovať určitý
/* druh chybovej správy, ako aj programovaním prešetriť program.
/*
/* Len ten, kto navrhol ukončovací program, môže skutočne rozhodnúť, ktorú
/* z akcií je najlepšie použiť.
/*
/*****/
static int newActionCode(int role,
                        int doesNotApply,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

/*-----*/
/*
/* Pridajte logiku na zaznamenanie chyby - napríklad do frontu správ
/* operátora, protokolu úloh, protokolu chýb špecifického pre aplikáciu,
/* aby ukončovací program vedel správne spracovať nový kód akcie.
/*
/* Zapamätajte si, že ak je kódovanie ponechané v pôvodnom stave,
/* znamená to, aby ste použili vyššie uvedenú stratégiu nič nevykonať.
/*
/*-----*/

    return QcstSuccessful;
} /* end newActionCode() */

/*****/
/*
/* Kód akcie = QcstCrgAcUndo
/*
/* Poznámka: Ukončovací program nikdy nie je zavolaný s kódom akcie späť
/* pre žiadnu z týchto predchádzajúcich kódov akcie:
/*

```



```

/* QcstCrgAcChgNodeStatus */
/* QcstCrgAcDelete */
/* QcstCrgAcDeleteCommand */
/* QcstCrgEndNode */
/* QstCrgAcRemoveNode (Ak je odstraňovaný uzol aktívny v klastri
/* a API je Odstrániť uzol klastra.
/* Akcia Odstrániť uzol z domény obnovy bude volaná
/* s akciou Späť a Odstrániť API uzla klastra bude
/* volaná s akciou Späť, ak je odstraňovaný uzol
/* neaktívny.
/* QcstCrgAcRestart */
/* QcstCrgAcUndo */
/*
/* API, ktoré volajú ukončovací program vykonávajú tieto tri kroky:
/* 1. Logika, vykonávaná pred zavolaním ukončovacieho programu.
/* 2. Zavolanie ukončovacieho programu.
/* 3. Logika, vykonávaná po zavolaní ukončovacieho programu.
/*
/* Akékoľvek chyby, ktoré sa vyskytnú počas kroku 2 alebo 3 zapríčinia
/* opätovné volanie ukončovacieho programu s kódom akcie späť. To poskytne
/* ukončovaciemu programu príležitosť odstúpiť od vykonávanej práce pri
/* prvom volaní API. API tiež odstúpi od vykonávanej práce, pričom sa
/* pokúsi vrátiť stav klastra a jeho objektov na pôvodný stav pred
/* zavolaním API.
/*
/* Odporúča sa vrátiť nasledujúce návratové kódy pre zadaný kód akcie,
/* keďže tieto návratové kódy vyústia do použitia najvhodnejšej
/* akcie.
/*
/* QcstCrgAcInitialize: QcstSuccessful; CRG sa nevytvorilo.
/* QcstCrgAcStart: QcstSuccessful; CRG sa nespustilo.
/* QcstCrgAcEnd: QcstFailWithOutRestart; CRG je teraz Indoubt
/* Príčina zlyhania sa musí
/* prešetriť.
/* QcstCrgAcReJoin: QcstFailWithOutRestart; CRG je teraz Indoubt
/* Príčina zlyhania sa musí
/* prešetriť.
/* QcstCrgAcFailover: QcstFailWithOutRestart; CRG je teraz Indoubt
/* Príčina zlyhania sa musí
/* prešetriť.
/* QcstCrgAcSwitchover: QcstFailWithOutRestart; CRG je teraz Indoubt
/* Príčina zlyhania sa musí
/* prešetriť.
/* QcstCrgAcAddNode: QcstSuccessful; Uzol sa nepridal.
/* QcstCrgAcRemoveNode: QcstFailWithOutRestart; CRG je teraz Indoubt
/* Príčina zlyhania sa musí
/* prešetriť.
/* QcstCrgAcChange: QcstSuccessful; Doména obnovy nie je
/* zmenená.
/*
/*****/
static int undoPriorAction(int role,
int priorRole,
Qcst_EXTP0100_t *crgData,
EpData *epData) {

/*-----*/
/*
/* Predchádzajúci kód akcie definuje, čo ukončovací program vykonával
/* v čase zlyhania, zrušenia alebo pri vrátení návratového kódu neúspechu.
/*
/*-----*/
if (crgData->Prior_Action_Code <= MaxAc )
return (*undoFcn[crgData-<Prior_Action_Code])
(role, priorRole, crgData,

```

```

        epData);
    else

/*-----*/
    /*
    /* Spoločnosť IBM v novom vydaní operačného systému definovala nový */
    /* kód akcie a tento ukončovací program ešte nebol aktualizovaný na */
    /* jeho spracovanie. Zatiaľ použite predvolenú akciu. */
    /*
    /*-----*/
    return newActionCode(role, priorRole, crgData, epData);
} /* end undoPriorAction() */

/*****/
/*
/* Kód akcie = QcstCrgAcUndo */
/*
/* Predchádzajúci kód akcie = QcstCrgAcInitialize */
/*
/* Nezabudnite vykonať: */
/* CRG nebude vytvorené. Objekty, ktoré mohli byť vytvorené v uzloch */
/* domény obnovy, by sa mali vymazať, pretože nasledujúce vytváranie */
/* by mohlo zlyhať, ak tieto objekty už existujú. */
/*
/*****/
static int undoCreateCrg(int role,
                        int doesNotApply,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

    return QcstSuccessful;
} /* end undoCreateCrg() */

/*****/
/*
/* Kód akcie = QcstCrgAcUndo */
/*
/* Predchádzajúci kód akcie = QcstCrgAcStart */
/*
/* Nezabudnite vykonať: */
/* Služby prostriedkov klastra zlyhali počas ukončovania API na */
/* spustenie CRG po tom, ako zavolali ukončovací program pomocou kódu */
/* akcie na spustenie. */
/*
/* Úloha ukončovacieho programu v primárnom uzle, ktorá spustila */
/* aplikáciu, sa zruší. Ukončovací program sa potom zavolá */
/* pomocou kódu akcie Späť. */
/*
/* Všetky ostatné uzly v doméne obnovy budú zavolané pomocou kódu akcie */
/* Späť. */
/*
/*****/
static int undoStartCrg(int role,
                       int doesNotApply,
                       Qcst_EXTP0100_t *crgData,
                       EpData *epData) {

    return QcstSuccessful;
} /* end undoStartCrg() */

/*****/
/*
/* Kód akcie = QcstCrgAcUndo */

```

```

/* */
/* Predchádzajúci kód akcie = QcstCrgAcEnd */
/* */
/* Nezabudnite vykonať: */
/* CRG nebude ukončené. Ak ukončovací program vykonal čokoľvek na */
/* ukončenie aplikácie, môže reštartovať túto aplikáciu alebo sa môže */
/* rozhodnúť, že ju nebude reštartovať. Ak aplikácia nebude */
/* reštartovaná, návratový kód by sa mal nastaviť na */
/* QcstFailWithOutRestart a stav CRG bude nastavené na Indoubt. */
/* */
/*****/
static int undoEndCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoEndCrg() */

/*****/
/* */
/* Kód akcie = QcstCrgAcUndo */
/* */
/* Predchádzajúci kód akcie = QcstCrgAcReJoin */
/* */
/* Nezabudnite vykonať: */
/* Vyskytla sa chyba, ktorá nepovolí členovi pripojiť sa do tohto CRG. */
/* Akýkoľvek zásah vykonaný pre kód akcie pripojenia a sa musí */
/* skontrolovať, aby sa zistilo, či sa niečo musí vrátiť späť, ak tento */
/* člen nie je aktívnym členom skupiny CRG. */
/* */
/*****/
static int undoMemberIsJoining(int role,
                              int doesNotApply,
                              Qcst_EXTP0100_t *crgData,
                              EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoMemberIsJoining() */

/*****/
/* */
/* Kód akcie = QcstCrgAcUndo */
/* */
/* Predchádzajúci kód akcie = QcstCrgAcFailover */
/* */
/* Nezabudnite vykonať: */
/* To neznamená, že proces zlyhania uzla alebo zlyhaný člen sa vráti */
/* späť. Toto zlyhanie je nezvratiteľné. Znamená to, že ukončovací */
/* program vrátil chybu z kódu akcie núdzového prepnutia, alebo služby */
/* prostriedkov klastra sa dostali do problémov po tom, ako zavolali */
/* ukončovací program. Ak bolo CRG aktívne počas pokusu o núdzové */
/* prepnutie, teraz už nie je. Ukončite zotaviteľný prostriedok */
/* a očakávajte kontrolu zlyhania servisným pracovníkom. Po oprave */
/* zlyhania bude nutné spustiť CRG pomocou API na spustenie CRG. */
/* */
/* */
/*****/
static int undoMemberIsLeaving(int role,
                              int doesNotApply,
                              Qcst_EXTP0100_t *crgData,
                              EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoMemberIsLeaving() */

```

```

/*****/
/*
/* Kód akcie = QcstCrgAcUndo
/*
/* Predchádzajúci kód akcie = QcstCrgAcSwitchover
/*
/* Nezabudnite vykonať:
/* Vyskytla sa chyba po presunutí prístupového bodu z pôvodného
/* primárneho uzla a pred jeho prijatím v novom primárnom uzle.
/* adresa IP bola ukončená v pôvodnom primárnom uzle pred presunutím
/* prístupového bodu, ale znova sa spustila v pôvodnom primárnom uzle.
/* Služby prostriedkov klastra sa teraz pokúsia presunúť prístupový bod
/* späť do pôvodného primárneho uzla. Ukončovací program aplikácie
/* a adresa IP na prebratie budú spustené v pôvodnom primárnom uzle.
/*
/*
/*****/
static int undoSwitchPrimary(int role,
                           int doesNotApply,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) {

    return QcstFailWithOutRestart;
} /* end undoSwitchPrimary()

/*****/
/*
/* Kód akcie = QcstCrgAcUndo
/*
/* Predchádzajúci kód akcie = QcstCrgAcAddNode
/*
/* Nezabudnite vykonať:
/* Ak boli v novom uzle vytvorené objekty, mali by sa odstrániť, aby
/* následná akcia pridania uzla do domény obnovy nezlyhala, ak sa znova
/* pokúsi vytvoriť objekty.
/*
/*
/*****/
static int undoAddNode(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

    return QcstSuccessful;
} /* end undoAddNode()

/*****/
/*
/* Kód akcie = QcstCrgAcUndo
/*
/* Predchádzajúci kód akcie = QcstCrgAcRemoveNode
/*
/* Nezabudnite vykonať:
/* Uzol sa je stále v doméne obnovy. Ak boli z tohto uzla odstránené
/* objekty, mali by sa pridať naspäť.
/*
/*
/*****/
static int undoRmvNode(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

    return QcstFailWithOutRestart;
}

```

```

} /* end undoRmvNode() */

/*****
/*
/* Kód akcie = QcstCrgAcUndo
/*
/* Predchádzajúci kód akcie = QcstCrgAcChange
/*
/* Nezabudnite vykonať:
/* Zmeny v CRG budú vrátené späť. CRG a jeho doména obnovy budú
/* v takom stave, v akom boli pred vykonanými zmenami.
/* Všetky zmeny vykonané ukončovacím programom by sa mali vrátiť tiež.
/*
*****/
static int undoChgCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

    return QcstSuccessful;
} /* end undoChgCrg() */

/*****
/*
/* Kód akcie = QcstCrgAcUndo
/*
/* Predchádzajúci kód akcie = QcstCrgAcCancelFailover
/*
/* Nezabudnite vykonať:
/* To neznamená, že proces zlyhania uzla alebo zlyhaný člen sa vráti
/* späť. Toto zlyhanie je nezvratiteľné. Znamená to, že služby
/* prostriedkov klastra sa dostali do problémov po tom, ako zavolali
/* ukončovací program. CRG bude nastavené na InDoubt bez ohľadu na to,
/* čo sa vrátilo z volania ukončovacieho programu. Bude potrebná
/* manuálna kontrola zlyhania. Po oprave zlyhania bude
/* potrebné spustiť CRG pomocou API na spustenie CRG.
/*
*****/
static int undoCancelFailover(int role,
                             int doesNotApply,
                             Qcst_EXTP0100_t *crgData,
                             EpData *epData) {

    return QcstSuccessful;
} /* end undoCancelFailover() */

/*****
/*
/* Jednoduchá rutina na prevzatie názvu objektu ukončeného znakom null,
/* názvu knižnice ukončeného znakom null a na vytvorenie 20-znakového
/* kvalifikovaného názvu bez ukončenia znakom null.
/*
*****/
static void bldDataAreaName(char *objName, char* libName, char *qualName) {

    memset(qualName, 0x40, 20);
    memcpy(qualName, objName, strlen(objName));
    qualName += 10;
    memcpy(qualName, libName, strlen(libName));
    return;
} /* end bldDataAreaName */

```

```

/*****
/*
/* Kontroluje sa údajová oblasť, či sú pripravené všetky CRG, na ktorých
/* je táto aplikácia závislá. Ak nie sú pripravené, je potrebné počkať
/* určitý čas a údajová oblasť sa skontroluje znova. Tento
/* proces kontroly a čakania pokračuje dovtedy, kým nebudú pripravené
/* všetky závislé CRG, alebo kým sa nedosiahne maximálny čas čakania.
/* Dĺžku času čakania je možné zmeniť na inú hodnotu, ak by určitá
/* situácia bola lepšia s kratším alebo dlhším časom čakania.
/*
/*
/*
/*****
static int checkDependCrgDataArea(unsigned int maxWaitTime) {

    Qus_EC_t errCode = { sizeof(Qus_EC_t), 0 };
    char dataAreaName[20];
    struct {
        Qwc_Rdtaa_Data_Returned_t stuff;
        char ready;
    } data;

/*-----*/
/*
/* Toto je akumulovaná dĺžka času čakania, kým budú závislé CRG
/* pripravené.
/*
/*-----*/
    unsigned int timeWaited = 0;

/*-----*/
/*
/* Zadajte definíciu dĺžky času čakania.
/*
/*-----*/
    _MI_Time    timeToWait;
    int hours   = 0;
    int minutes = 0;
    int seconds = WaitSecondsIncrement;
    int hundreths = 0;
    short int options = _WAIT_NORMAL;
    mitime( &timeToWait, hours, minutes, seconds, hundreths );

/*-----*/
/*
/* Zadajte kvalifikovaný názov údajovej oblasti.
/*
/*-----*/
    bldDataAreaName(DependCrgDataArea, ApplLib, dataAreaName);

/*-----*/
/*
/* Získajte údaje z údajovej oblasti, ktoré označujú, či sú CRG
/* pripravené alebo nie. Táto údajová oblasť je aktualizovaná pomocou
/* High Availability Business Partners, ak je aplikácia pripravená na
/* spracovanie.
/*
/*-----*/
    QWCRDTAA(&data,
            sizeof(data),

```

```

        dataAreaName,
        offsetof(Qcst_HAAPP0_t,Data_Status)+1, /* API wants a 1 origin */
        sizeof(data.ready),
        &errCode);

/*-----*/
/*
/* Ak sú závislé CRG nepripravené, počkajte chvíľu a skontrolujte znova. */
/*
/*-----*/
    while (data.ready != Data_Available) {

/*-----*/
/*
/* Ak závislé CRG nebudú pripravené do skončenia maximálneho času
/* čakania, vráťte chybu. Zvážte zaprotokolovanie niektorej správy,
/* ktorá opisuje, prečo aplikácia nebola spustená. Potom bude možné
/* skontrolovať problém.
/*
/*-----*/
    if (timeWaited >= maxWaitTime)
        return QcstFailWithoutRestart;

/*-----*/
/*
/* Počkajte, aby sa sprístupnili skupiny CRG.
/*
/*-----*/
    waittime(&timeToWait, options);
    timeWaited += WaitSecondsIncrement;

/*-----*/
/*
/* Získajte znova informácie z údajovej oblasti, aby ste zistili, či
/* sú CRG údajov pripravené.
/*
/*-----*/
        QWCRDTAA(&data,
                sizeof(data),
                dataAreaName,
                offsetof(Qcst_HAAPP0_t,Data_Status)+1, /* API wants a 1 origin */
                sizeof(data.ready),
                &errCode);
    }

    return QcstSuccessful;
} /* end checkDependCrgDataArea */

/*****
/*
/* Údajová oblasť CRG aplikácie je aktualizovaná, aby označovala, či je
/* aplikácia spustená alebo nie. Tieto informácie údajovej oblasti
/* sa používajú pre High Availability Business Partners na koordináciu
/* aktivít prepínania medzi CRG, ktoré sú navzájom závislé.
/*
/*
/*
/*****
static void setApp1CrgDataArea(char status) {

```



```

char cmd[54];
char cmdEnd[3] = {0x00, ' ', 0x00};

/*-----*/
/*
/* Nastavte reťazec príkazového riadka s názvom knižnice údajovej
/* oblasti, názvom údajovej oblasti a znakom, ktorý bude vložený do
/* údajovej oblasti. Potom spustite príkaz.
*/
/*
/*-----*/
    memcpy(cmd, "CHGDTAARA DTAARA(", strlen("CHGDTAARA DTAARA")+1);
    strcat(cmd, ApplLib);
    strcat(cmd, "/");
    strcat(cmd, ApplCrgDataArea);
    strcat(cmd, " (425 1) VALUE(");
    cmdEnd[0] = status;
    strcat(cmd, cmdEnd);

    system(cmd);

    return;
} /* end setApplCrgDataArea */

/*****
/*
/* Táto funkcia sa volá vždy, keď ukončovací program prijme výnimku, ktorá
/* nie je špecificky monitorovaná inou obslužnou rutinou výnimky. Pridajte
/* príslušnú logiku, ktorá vykoná prípadné požadované funkcie vyčistenia.
/* Vtedy sa nastaví návratový kód zlyhania a do operačného syst. sa vráti
/* riadenie. Úloha, v ktorej sa vykonáva tento ukonč. program sa ukončí.
/*
/* Po zavolaní tejto funkcie môže rola myData->role stále obsahovať
/* hodnotu UnknownRole, ak sa vyskytla výnimka pred nastavením hodnoty
/* roly tohto uzla. Aby ste zabezpečili úplnú správnosť, mala by sa
/* otestovať hodnota roly UnknownRole pred vykonaním akýchkoľvek
/* rozhodnutí na základe hodnoty roly.
/*
/*****
static void unexpectedExceptionHandler(_INTRPT_Hndlr_Parms_T *exData) {
/*-----*/
/*
/* Nastavte smerník na štruktúru obsahujúcu údaje, ktoré boli zadané
/* pre obslužnú rutinu výnimky.
*/
/*
/*-----*/
    HandlerDataT *myData = (HandlerDataT *)exData->Com_Area;

/*-----*/
/*
/* Vykonajte potrebný počet funkcií vyčistenia. Mohli by sa ponechať
/* niektoré globálne stavové informácie, aby obslužná rutina výnimky
/* poznala, ktoré kroky boli vykonané pred výskytom zlyhania a následne
/* bude vedieť, ktoré kroky vyčistenia sa musia vykonať. Informácie
/* o stave sa môžu ponechať v štruktúre HandlerDataT alebo na inom
/* mieste, na ktoré bude možné adresovať túto funkciu.
*/
/*
/*-----*/

/*****

```

```

/* */
/* Ak toto je primárny uzol a aplikácia bola spustená, ukončíte ju. */
/* Aplikácia sa ukončí, pretože bude znova zavolaný ukončovací program */
/* pomocou kódu akcie Restart a chceme, aby funkcia restartCrg() vždy */
/* pracovala rovnako. Okrem toho, ukončenie aplikácie môže objasniť */
/* podmienku, ktorá zapríčinila výnimku, ktorá nás sem dostala. */
/* Ak je to možné, upozornite užívateľov, aby prestali používať */
/* aplikáciu, potom bude možné vykonať operácie správnym spôsobom. */
/* */

/*-----*/
endApplication(myData->actionCode,
               myData->role,
               myData->priorRole,
myData->crgData,
               myData->epData);

/*-----*/
/* */
/* Nastavte návratový kód ukončovacieho programu. */
/* */

/*-----*/
*myData->retCode = QcstFailWithRestart;

/*-----*/
/* */
/* Povolenie postupného prenosu výnimky až navrch zásobníka volaní. */
/* */
/*-----*/
return;
} /* end unexpectedExceptionHandler */

/*****/
/* */
/* Táto funkcia sa volá vždy, keď bola zrušená úloha, v ktorej sa vykonáva */
/* tento ukončovací program. Úloha sa mohla zrušiť z ľubovoľnej z týchto */
/* príčin (tento zoznam nemusí zahŕňať všetky prípady): */
/* - API zruší aktívne CRG aplikácie. API End CRG, Initiate Switchover, */
/* End Cluster Node, Remove Cluster Node alebo Delete Cluster zruší */
/* úlohu, ktorá bola odovzdaná po zavolaní ukončovacieho programu */
/* pomocou kódu akcie Start. */
/* - operátor zruší úlohu z niektorej obrazovky operačného systému, */
/* napríklad Work with Active Jobs */
/* - ukončí sa podsystém, v ktorom sa vykonáva táto úloha */
/* - ukončia sa všetky podsystémy */
/* - systém sa vypne */
/* - vyskytla sa počítačová kontrola operačného systému */
/* */
/* Po zavolaní tejto funkcie môže rola myData->role stále obsahovať */
/* hodnotu UnknownRole, ak sa pred nastavením hodnoty roly tohto uzla */
/* roly tohto uzla. Aby ste zabezpečili úplnú správnosť, mala by sa */
/* otestovať hodnota roly UnknownRole pred vykonaním akýchkoľvek */
/* rozhodnutí na základe hodnoty roly. */
/* */
/*****/
static void cancelHandler(_CNL_Hndlr_Parms_T *cnlData) {

/*-----*/
/* */
/* Nastavte smerník na štruktúru obsahujúcu údaje, ktoré boli zadané */
/* pre obslužnú rutinu zrušenia. */

```

```

/*                                                                 */
/*-----*/
HandlerDataT *myData = (HandlerDataT *)cnldata->Com_Area;

/*-----*/
/*                                                                 */
/*  /* Vykonajte potrebný počet funkcií vyčistenia. Mohli by sa ponechať */
/*  /* niektoré globálne stavové informácie, aby obslužná rutina zrušenia */
/*  /* vedela, ktoré kroky sa vykonali pred zrušením úlohy a teda vedela, */
/*  /* či sa funkcia naozaj dokončila úspešne alebo sa dokončila len */
/*  /* čiastočne a či je potrebné vykonať vyčistenie. Tieto informácie */
/*  /* o stave sa môžu ponechať v štruktúre HandlerDataT alebo na inom */
/*  /* mieste, na ktoré bude možné adresovať túto funkciu. */
/*                                                                 */
/*-----*/

/*-----*/

/*                                                                 */
/*  /* Táto úloha sa ruší. Ak vykonávala aplikáciu v dôsledku kódu */
/*  /* akcie Start alebo Restart, ihneď ukončíte aplikáciu. */
/*  /* Táto úloha sa ruší, pretože bolo použité API na prepnutie alebo API */
/*  /* pre iné služby prostriedkov klastra, ktoré ovplyvnilo primárny uzol, */
/*  /* alebo niekto vykonal úlohu zrušenia príkazom príkazového riadka na */
/*  /* systémovej obrazovke a podobne. */
/*-----*/
endApplication(myData->actionCode,
               myData->role,
               myData->priorRole,
               myData->crgData,
               myData->epData);

/*-----*/
/*                                                                 */
/*  /* Nastavte návratový kód ukončovacieho programu. */
/*                                                                 */
/*-----*/
myData->retCode = QcstSuccessful;

/*-----*/
/*                                                                 */
/*  /* Vráťte sa do operačného systému kvôli úplnému ukončeniu úlohy. */
/*                                                                 */
/*-----*/
return;
} /* end cancelHandler */

/*****
/*
/* Bežná rutina na ukončenie aplikácie rôznymi funkciami kódu akcie,
/* obslužnou rutinou výnimky a obslužnou rutinou zrušenia.
/*
/*****
static void endApplication(unsigned int actionCode,
                          int role,
                          int priorRole,
                          Qcst_EXTP0100_t *crgData,
                          EpData *epData) {

```

```

    if ( role == QcstPrimaryNodeRole
        &&
        crgData->Original_Cluster_Res_Grp_Stat == QcstCrgActive) {
/*-----*/
/*
/* Ihneď ukončíte aplikáciu pridaním logiky. Možno budete musieť */
/* pridať logiku, aby ste zistili, či sa aplikácia stále vykonáva, */
/* pretože táto funkcia mohla byť zavolaná pre kód akcie a potom znova */
/* z obslužnej rutiny zrušenia (napríklad End CRG). */
/*-----*/

/*-----*/
/*
/* Po ukončení aplikácie zaktualizujte údajovú oblasť, */
/* aby sa zobrazilo, že aplikácia sa už nevykonáva. */
/*-----*/
        setApplCrgDataArea(Appl_Ended);
    }

    return;
} /* end endApplication */

/*****
/*
/* Vytlačte údaje zadané pre tento program. */
/*-----*/
static void printParms(int actionCode,
                      int role,
                      int priorRole,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

    unsigned int i;
    char *str;

    /* Vytlačte kód akcie. */
    printf("%s", "Action_Code = ");
    printActionCode(actionCode);

    /* Vytlačte údaje závislé od kódu akcie. */
    printf("%s", " Action_Code_Dependent_Data = ");
    switch (crgData->Action_Code_Dependent_Data) {
        case QcstNoDependentData: str = "QcstNoDependentData";
                                break;
        case QcstMerge:          str = "QcstMerge";
                                break;
        case QcstJoin:           str = "QcstJoin";
                                break;
        case QcstPartitionFailure: str = "QcstPartitionFailure";
                                break;
        case QcstNodeFailure:    str = "QcstNodeFailure";
                                break;
        case QcstMemberFailure:  str = "QcstMemberFailure";
                                break;
        case QcstEndNode:        str = "QcstEndNode";
                                break;
        case QcstRemoveNode:     str = "QcstRemoveNode";
    }
}

```

```

        break;
    case QcstApplFailure:    str = "QcstApplFailure";
                            break;
    case QcstResourceEnd:   str = "QcstResourceEnd";
                            break;
    case QcstDltCluster:    str = "QcstDltCluster";
                            break;
    case QcstRmvRcvyDmnNode: str = "QcstRmvRcvyDmnNode";
                            break;
    case QcstDltCrg:        str = "QcstDltCrg";
                            break;
    default: str = "unknown action code dependent data";
}
printf("%s \n", str);

/* Vytlačte predchádzajúci kód akcie. */
printf("%s", " Prior_Action_Code = ");
if (crgData->Prior_Action_Code)
    printActionCode(crgData->Prior_Action_Code);
printf("\n");

/* Vytlačte názov klastra. */
printStr(" Cluster_Name = ",
        crgData->Cluster_Name, sizeof(Qcst_Cluster_Name_t));

/* Vytlačte názov CRG. */
printStr(" Cluster_Resource_Group_Name = ",
        crgData->Cluster_Resource_Group_Name, sizeof(Qcst_Crg_Name_t));
/* Vytlačte typ CRG. */
printf("%s \n", " Cluster_Resource_Group_Type = QcstCrgApplResiliency");
/* Vytlačte stav CRG. */
printf("%s", " Cluster_Resource_Group_Status = ");
printCrgStatus(crgData->Cluster_Resource_Group_Status);

/* Vytlačte pôvodný stav CRG. */
printf("%s", " Original_Cluster_Res_Grp_Stat = ");
printCrgStatus(crgData->Original_Cluster_Res_Grp_Stat);

/* Vytlačte názov frontu Distribute Information. */
printStr(" DI_Queue_Name = ",
        crgData->DI_Queue_Name, sizeof(crgData->DI_Queue_Name));
printStr(" DI_Queue_Library_Name = ",
        crgData->DI_Queue_Library_Name,
        sizeof(crgData->DI_Queue_Library_Name));

/* Vytlačte atribúty CRG. */
printf("%s", " Cluster_Resource_Group_Attr = ");
if (crgData->Cluster_Resource_Group_Attr & QcstTcpConfigByUsr)
    printf("%s", "User Configures IP Takeover Address");
printf("\n");

/* Vytlačte ID tohto uzla. */
printStr(" This_Nodes_ID = ",
        crgData->This_Nodes_ID, sizeof(Qcst_Node_Id_t));

/* Vytlačte rolu tohto uzla. */
printf("%s %d \n", " this node's role = ", role);

/* Vytlačte predchádzajúcu rolu tohto uzla. */
printf("%s %d \n", " this node's prior role = ", priorRole);

/* Vytlačte doménu obnovy, z ktorej pochádza táto rola. */
printf("%s", " Node_Role_Type = ");
if (crgData->Node_Role_Type == QcstCurrentRcvyDmn)
    printf("%s \n", "QcstCurrentRcvyDmn");
else

```

```

printf("%s \n", "QcstPreferredRcvyDmn");

/* Vytlačte ID zmeneného uzla (ak existuje). */
printStr(" Changing_Node_ID = ",
        crgData->Changing_Node_ID, sizeof(Qcst_Node_Id_t));

/* Vytlačte rolu zmeneného uzla (ak existuje). */
printf("%s", " Changing_Node_Role = ");
if (crgData->Changing_Node_Role == -3)
    printf("%s \n", "*LIST");
else if (crgData->Changing_Node_Role == -2)
    printf("%s \n", "does not apply");
else
    printf("%d \n", crgData->Changing_Node_Role);

/* Vytlačte adresu IP na prebratie. */
printStr(" Takeover_IP_Address = ",
        crgData->Takeover_IP_Address, sizeof(Qcst_TakeOver_IP_Address_t));
/* Vytlačte názov úlohy. */
printStr(" Job_Name = ", crgData->Job_Name, 10);

/* Vytlačte zmeny CRG. */
printf("%s \n", " Cluster_Resource_Group_Changes = ");
if (crgData->Cluster_Resource_Group_Changes & QcstRcvyDomainChange)
    printf(" %s \n", "Recovery domain changed");
if (crgData->Cluster_Resource_Group_Changes & QcstTakeOverIpAddrChange)
    printf(" %s \n", "Takeover IP address changed");

/* Vytlačte čas čakania na núdzové prepnutie. */
printf("%s", "Failover_Wait_Time = ");
if (crgData->Failover_Wait_Time == QcstFailoverWaitForever)
    printf("%d %s \n", crgData->Failover_Wait_Time, "Wait forever");
else if (crgData->Failover_Wait_Time == QcstFailoverNoWait)
    printf("%d %s \n", crgData->Failover_Wait_Time, "No wait");
else
    printf("%d %s \n", crgData->Failover_Wait_Time, "minutes");

/* Vytlačte predvolenú akciu núdzového prepnutia. */
printf("%s", "Failover_Default_Action = ");
if (crgData->Failover_Default_Action == QcstFailoverProceed)
    printf("%d %s \n", crgData->Failover_Default_Action, "Proceed");
else
    printf("%d %s \n", crgData->Failover_Default_Action, "Cancel");
/* Vytlačte názov frontu správ núdzového prepnutia. */
printStr(" Failover_Msg_Queue = ",
        crgData->Failover_Msg_Queue, sizeof(crgData->Failover_Msg_Queue));
printStr(" Failover_Msg_Queue_Lib = ",
        crgData->Failover_Msg_Queue_Lib,
        sizeof(crgData->Failover_Msg_Queue_Lib));

/* Vytlačte verziu klastra. */
printf("%s %d \n",
        " Cluster_Version = ", crgData->Cluster_Version);

/* Vytlačte úroveň zmeny verzie klastra */
printf("%s %d \n",
        " Cluster_Version_Mod_Level = ",
        crgData->Cluster_Version_Mod_Level);

/* Vytlačte profil užívateľa s požiadavkou. */
printStr(" Req_User_Profile = ",
        crgData->Req_User_Profile, sizeof(crgData->Req_User_Profile));
/* Vytlačte dĺžku údajov v štruktúre. */
printf("%s %d \n",
        " Length_Info_Returned = ", crgData->Length_Info_Returned);
/* Vytlačte posunutie do poľa domény obnovy. */
printf("%s %d \n",

```

```

        " Offset_Rcvy_Domain_Array = ", crgData->Offset_Rcvy_Domain_Array);
/* Vytlačte počet uzlov v poli domény obnovy. */
printf("%s %d \n",
        " Number_Nodes_Rcvy_Domain = ",
crgData->Number_Nodes_Rcvy_Domain);

/* Vytlačte aktuálnu alebo novú doménu obnovy. */
printRcvyDomain(" The recovery domain:",
        crgData->Number_Nodes_Rcvy_Domain,
        (Qcst_Rcvy_Domain_Array1_t *)
        ((char *)crgData + crgData->Offset_Rcvy_Domain_Array));
/* Vytlačte posunutie poľa predchádzajúcej domény obnovy. */
printf("%s %d \n",
        " Offset_Prior_Rcvy_Domain_Array = ",
        crgData->Offset_Prior_Rcvy_Domain_Array);

/* Vytlačte počet uzlov v poli predchádzajúcej domény obnovy. */
printf("%s %d \n",
        " Number_Nodes_Prior_Rcvy_Domain = ",
        crgData->Number_Nodes_Prior_Rcvy_Domain);

/* Vytlačte predchádzajúcu doménu obnovy, ak bola zadaná. */
if (crgData->Offset_Prior_Rcvy_Domain_Array) {
    printRcvyDomain(" The prior recovery domain:",
        crgData->Number_Nodes_Prior_Rcvy_Domain,
        (Qcst_Rcvy_Domain_Array1_t *)
        ((char *)crgData + crgData->Offset_Prior_Rcvy_Domain_Array));
}

return;
} /* end printParms */

/*****
/*
/* Vytlačte reťazec kódu akcie.
/*
/*
*****/
static void printActionCode(unsigned int ac) {

    char *code;
    switch (ac) {
        case QcstCrgAcInitialize: code = "QcstCrgAcInitialize";
            break;
        case QcstCrgAcStart: code = "QcstCrgAcStart";
            break;
        case QcstCrgAcRestart: code = "QcstCrgAcRestart";
            break;
        case QcstCrgAcEnd: code = "QcstCrgAcEnd";
            break;
        case QcstCrgAcDelete: code = "QcstCrgAcDelete";
            break;
        case QcstCrgAcReJoin: code = "QcstCrgAcReJoin";
            break;
        case QcstCrgAcFailover: code = "QcstCrgAcFailover";
            break;
        case QcstCrgAcSwitchover: code = "QcstCrgAcSwitchover";
            break;
        case QcstCrgAcAddNode: code = "QcstCrgAcAddNode";
            break;
        case QcstCrgAcRemoveNode: code = "QcstCrgAcRemoveNode";
            break;
        case QcstCrgAcChange: code = "QcstCrgAcChange";
            break;
        case QcstCrgAcDeleteCommand: code = "QcstCrgAcDeleteCommand";
            break;
        case QcstCrgAcUndo: code = "QcstCrgAcUndo";

```

```

        break;
    case QcstCrgEndNode:    code = "QcstCrgEndNode";
                           break;
    case QcstCrgAcAddDevEnt: code = "QcstCrgAcAddDevEnt";
                           break;
    case QcstCrgAcRmvDevEnt: code = "QcstCrgAcRmvDevEnt";
                           break;
    case QcstCrgAcChgDevEnt: code = "QcstCrgAcChgDevEnt";
                           break;
    case QcstCrgAcChgNodeStatus: code = "QcstCrgAcChgNodeStatus";
                           break;
    case QcstCrgAcCancelFailover: code = "QcstCrgAcCancelFailover";
                           break;
    case QcstCrgAcVerificationPhase: code = "QcstCrgAcVerificationPhase";
                           break;
    default:                code = "unknown action code";
                           break;
}
printf("%s", code);

return;
} /* end printActionCode */

/*****
/*
/* Vytlačte stav CRG.
/*
/*
*****/
static void printCrgStatus(int status) {

    char * str;
    switch (status) {
        case QcstCrgActive:    str = "QcstCrgActive";
                               break;
        case QcstCrgInactive: str = "QcstCrgInactive";
                               break;
        case QcstCrgIndoubt:  str = "QcstCrgIndoubt";
                               break;
        case QcstCrgRestored: str = "QcstCrgRestored";
                               break;
        case QcstCrgAddnodePending: str = "QcstCrgAddnodePending";
                               break;
        case QcstCrgDeletePending: str = "QcstCrgDeletePending";
                               break;
        case QcstCrgChangePending: str = "QcstCrgChangePending";
                               break;
        case QcstCrgEndCrgPending: str = "QcstCrgEndCrgPending";
                               break;
        case QcstCrgInitializePending: str = "QcstCrgInitializePending";
                               break;
        case QcstCrgRemovenodePending: str = "QcstCrgRemovenodePending";
                               break;
        case QcstCrgStartCrgPending: str = "QcstCrgStartCrgPending";
                               break;
        case QcstCrgSwitchOverPending: str = "QcstCrgSwitchOverPending";
                               break;
        case QcstCrgDeleteCmdPending: str = "QcstCrgDeleteCmdPending";
                               break;
        case QcstCrgAddDevEntPending: str = "QcstCrgAddDevEntPending";
                               break;
        case QcstCrgRmvDevEntPending: str = "QcstCrgRmvDevEntPending";
                               break;
        case QcstCrgChgDevEntPending: str = "QcstCrgChgDevEntPending";
                               break;
        case QcstCrgChgNodeStatusPending: str = "QcstCrgChgNodeStatusPending";
                               break;
    }
}

```



```

        default: str = "unknown CRG status";
    }
    printf("%s \n", str);

    return;
} /* end printCrgStatus */

/*****
/*
/* Vytlačte doménu obnovy.
/*
/*
*****/
static void printRcvyDomain(char *str,
                            unsigned int count,
                            Qcst_Rcvy_Domain_Array1_t *rd) {

    unsigned int i;
    printf("\n %s \n", str);
    for (i=1; i<=count; i++) {
        printStr("    Node_ID = ", rd->Node_ID, sizeof(Qcst_Node_Id_t));
        printf("%s %d \n", "    Node_Role = ", rd->Node_Role);
        printf("%s", "    Membership_Status = ");
        switch (rd->Membership_Status) {
            case 0: str = "Active";
                    break;
            case 1: str = "Inactive";
                    break;
            case 2: str = "Partition";
                    break;
            default: str = "unknown node status";
        }
        printf("%s \n", str);
        rd++;
    }
    return;
} /* end printRcvyDomain */

/*****
/*
/* Zrežazte reťazec ukončený znakom null a reťazec, ktorý nie je ukončený
/* znakom null, a vytlačte ho.
/*
/*
*****/
static void printStr(char *s1, char *s2, unsigned int len) {

    char buffer[132];
    memset(buffer, 0x00, sizeof(buffer));
    memcpy(buffer, s1, strlen(s1));
    strncat(buffer, s2, len);
    printf("%s \n", buffer);
    return;
} /* end printStr */

```

---

## Plán pre klaster

- | Dozviete sa, čo musíte vykonať pred nastavením klastrov serverov iSeries. Dozviete sa o podmienkach pre nastavenie
- | klastrov, ako aj niekoľko rád o tom, ako klaster navrhnuť. Nakoniec si budete môcť prečítať tipy pre nastavenie vašej
- | siete a niekoľko rád týkajúcich sa výkonu klastra.

Táto téma opisuje požiadavky, ktoré musíte splniť pred používaním klastrovania. Nasledujúce témy vám poskytnú všeobecné koncepty, požiadavky a úvahy pre návrh riešenia klastrovania.

## Riešenia konfigurácie a správy klastrov

Služby klastrových prostriedkov poskytujú základnú infraštruktúru klastrov. Existuje niekoľko metód, ktoré vám dovoľia využiť výhody schopností klastrovania, poskytované službami klastrových prostriedkov.

Služby klastrových prostriedkov i5/OS v iSeries poskytujú základnú infraštruktúru, ktorá vám umožňuje používať klaster. Služby klastrových prostriedkov poskytujú skupinu integrovaných služieb, ktoré udržiavajú topológiu klastra, vykonávajú monitorovanie pulzu a umožňujú vytváranie a administráciu administrácie klastra a skupín klastrových prostriedkov. Služby klastrových prostriedkov taktiež poskytujú funkcie spoľahlivého zasielania správ, ktoré sledujú každý uzol v klastri a zabezpečujú, aby mal každý uzol nepretržité informácie o stave klastrových prostriedkov.

Kým služby klastrových prostriedkov poskytujú základnú infraštruktúru klastra, existuje niekoľko spôsobov, ako môžete používať výhody týchto klastrových možností. Každý zo spôsobov má iné výhody a možnosti.

**Dôležité:** Použite len jedno z týchto riešení. Ak sa budete pokúšať používať viac, než jedno riešenie pre vytváranie a správu klastrov, môže dôjsť ku konfliktom, problémom a nepredvídateľným udalostiam. Informácie v Informačnom centre iSeries dokumentujú procedúry špecifické pre Navigátor iSeries, príkazy CL a rozhrania API pre služby klastrových prostriedkov. Ak používate riešenie softvéru strednej úrovne pre klastre od obchodného partnera IBM, informácie o procedúrach pre vykonanie úloh nájdete v dokumentácii, ktorá sa dodáva k produktu.

### Manažovanie klastrov v Navigátore iSeries

IBM ponúka rozhranie na manažovanie klastrov, ktoré je dostupné v Navigátore iSeries a ktoré poskytuje voľba 41 (i5/OS - Vysokodostupné prepínateľné prostriedky).

| Týmto rozhraním môžete vytvoriť a riadiť klaster, ktorý používa prepínateľnú nezávislú diskovú oblasť (prepínateľnú ASP) na skontrolovanie dostupnosti údajov. Umožňuje vám tiež vytvárať klastre, skupiny CRG, administratívne domény klastra a prostriedky.

| **Dôležité:** Rozhranie Navigátora iSeries na manažovanie klastrov neobsahuje všetky schopnosti, ktoré poskytujú služby klastrových prostriedkov. Berte na vedomie, že zatiaľ čo Navigátor iSeries poskytuje mnoho potrebných funkcií na konfiguráciu a manažovanie klastra, niektoré schopnosti sú dostupné len pomocou príkazov a rozhraní API pre klastre alebo pomocou aplikácie strednej úrovne pre klastre od obchodného partnera IBM, v závislosti od konkrétnej situácie. Napríklad architektúra klastrov iSeries podporuje v klastri až 128 uzlov, rozhranie Navigátora iSeries však podporuje v klastri maximálne štyri uzly. Manažment klastrov v Navigátore iSeries zahŕňa sprievodcu, ktorý vás prevedie vytvorením jednoduchého klastra so štyrmi uzlami. Ak to nepostačuje vašim požiadavkám na klaster, mali by ste zvážiť použitie príkazov a rozhraní API pre klastre IBM alebo produktov softvéru strednej úrovne pre klastre od obchodných partnerov IBM.

| Pomocou Navigátora iSeries môžete vykonávať tiež iné úlohy týkajúce sa klastrov. Niektoré z týchto úloh zahŕňajú:

- Pridanie uzla do existujúceho klastra
- Pridanie prepínateľných zariadení do klastra
- Pridanie prepínateľných aplikácií do klastra
- Pridanie prepínateľnej údajovej skupiny do klastra
- Zmena role uzlov v doméne obnovy
- Zmenenie opisu klastra
- Odstránenie klastra
- Spustenie klastrovania
- Zastavenie klastrovania
- Prezeranie správ o činnosti klastra
- Vytvorenie administratívnej domény klastra
- Pridanie položky monitorovaného prostriedku

| Celý zoznam úloh pre klastre, ktoré sú dostupné v Navigátore iSeries, nájdete v online pomoci pre klastre.

**Poznámka:** Rozhranie Navigátora iSeries na manažovanie klastrov nepodporuje logickú replikáciu objektov. Ak chcete použiť replikáciu, mali by ste zväziť použitie produktov pre klastre od obchodných partnerov IBM, zaoberajúcich sa vysokou dostupnosťou.

### **Súvisiace koncepty**

iSeries navigátor

“Klastrové príkazy a API”

Služby klastrových prostriedkov i5/OS poskytujú množinu príkazov riadiaceho jazyka (CL), aplikačných programových rozhraní (API) a funkcií, pomocou ktorých môžu poskytovatelia alebo užívatelia aplikácií iSeries zvýšiť ich dostupnosť.

“Obchodní partneri IBM poskytujúci softvér strednej úrovne pre klastre a dostupné produkty pre klastrovanie” na strane 79

Od obchodných partnerov IBM poskytujúcich softvér strednej úrovne pre klastre môžete získať produkt zabezpečujúci funkcie logickej replikácie, ktoré sú integrálnou súčasťou klastrovania a zjednodušujú vytváranie a manažovanie klastrov.

“Bežné problémy s klastrom” na strane 132

Táto téma uvádza niektoré bežné problémy, ku ktorým môže dôjsť v klastri, ako aj spôsoby, ako sa im vyhnúť alebo vykonať obnovu.

### **Súvisiaci odkaz**

“Často kladené otázky o manažovaní klastrov v Navigátore iSeries” na strane 141

Otázky a odpovede týkajúce sa grafického užívateľského prostredia Navigátora iSeries na vytváranie a manažovanie klastrov.

## **Klastrové príkazy a API**

Služby klastrových prostriedkov i5/OS poskytujú množinu príkazov riadiaceho jazyka (CL), aplikačných programových rozhraní (API) a funkcií, pomocou ktorých môžu poskytovatelia alebo užívatelia aplikácií iSeries zvýšiť ich dostupnosť.

Môžete si napísať vlastnú aplikáciu a konfigurovať a spravovať váš klaster použitím klastrových príkazov v riadiacom jazyku (CL) a rozhraní na programovanie aplikácií (API). Tieto príkazy a rozhrania API využívajú technológiu, ktorú poskytujú služby klastrových prostriedkov, dodávané ako súčasť systému i5/OS.

### **QUSRTOOL**

Služby klastrových prostriedkov tiež poskytujú vzorové príkazy v knižnici QUSRTOOL, ktoré sa mapujú na rozhrania API, ktoré nemajú podporované rozhranie príkazov. Príkazy QUSRTOOL môžu byť v niektorých prostrediach užitočné. Môžete napríklad zmeniť monitorovanie aktivity, alebo poslať informáciu cez celý klaster. Ak chcete získať viac informácií o týchto ukázkových príkazoch, pozrite si člen TCSTINFO v súbore QUSRTOOL/QATTINFO. V knižnici QUSRTOOL je tiež priložený vzorový ukončovací program CRG aplikácií. Vzorový zdrojový kód môže byť použitý ako základ pri písaní ukončovacieho programu. Príklady zdrojových kódov TCSTDTAEXT v súbore QATTSYSC obsahujú zdroje pre program na vytvorenie oblasti údajov QCSTHAAPPI a QCSTHAAPP0 a súboru QACSTOSDS (zadávateľ objektov).

### **Súvisiace úlohy**

“Pridanie uzla do klastra” na strane 99

Uzol môžete do klastra pridať pomocou Navigátora iSeries alebo príkazov.

## **Popisy príkazov jazyka CL pre riadenie klastra a API:**

Existuje mnoho rozhraní API a príkazov CL, pomocou ktorých môžete nakonfigurovať, aktivovať a manažovať klastre, uzly klastra a skupiny prostriedkov klastra.

Nasledovné tabuľky zobrazujú názov a stručný popis príkazov jazyka CL (control language) pre riadenie klastra a skupiny klastrových prostriedkov a API, ktoré sú k dispozícii. Príkazy CL pre klastre sú dostupné len v operačnom systéme OS/400 V5R2M0 alebo novšom.

- | Prvá tabuľka obsahuje príkazy a rozhrania API na konfigurovanie, aktivovanie a manažovanie **klastra a uzlov** v
- | klastru. Druhá tabuľka obsahuje príkazy a rozhrania API na konfigurovanie, aktivovanie a manažovanie **skupín**
- | **prostriedkov klastra** v klastru. Tretia tabuľka obsahuje príkazy na konfigurovanie a manažovanie **administratívnej**
- | **domény klastra**. Štvrtá tabuľka obsahuje opisy API integrovaného operačného systému, pomocou ktorých môžete
- | pridať alebo odstrániť položky monitorovaných prostriedkov z administratívnej domény klastra.

Tabuľka 8. Opisy príkazov CL a rozhraní API na riadenie klastra

Popis	Príkaz jazyka CL pre riadenie klastra	Názov API riadenia klastra
<b>Pridanie položky uzla klastra</b> Pridáva uzol k zoznamu členov existujúceho klastra. Takisto priradí adresy rozhrania IP, ktoré budú používať klastrové komunikácie.	ADDCLUNODE	Pridať zadanú hodnotu klastrového uzla QcstAddClusterNodeEntry (Add Cluster Node Entry)
<b>Pridanie položky domény zariadení</b> Pridáva uzol do zoznamu členov domény zariadenia tak, že sa môže zúčastňovať na obnovovacích akciách pre obnoviteľné zariadenia. Doména zariadenia sa vytvorí pridaním prvého uzla do danej domény.	ADDDEVDMNE	Pridať zadanú hodnotu domény zariadenia QcstAddDeviceDomainEntry (Add Device Domain Entry)
<b>Úprava verzie klastra, nastavenie verzie klastra</b> Prispôsobuje aktuálnu klastrovú verziu na ďalšej úrovni tak, že nová funkcia môže byť použitá v rámci klastra.	CHGCLUVER	QcstAdjustClusterVersion (Adjust Cluster Version)
<b>Zmena položky uzla klastra</b> Mení polia v zadanej hodnote klastrového uzla. Napríklad môžu byť zmenené adresy rozhrania IP, ktoré sa používajú pre komunikácie klastra.	CHGCLUNODE	QcstChangeClusterNodeEntry (Change Cluster Node Entry)
<b>Zmena služieb klastrových prostriedkov, Zmena konfigurácie klastra</b> Prispôsobuje výkon klastra a parametre konfigurácie ladenia tak, aby sa zhodovali s prostredím komunikácie siete, používaným pre komunikácie klastra.	CHGCLUCFG	Zmeniť služby klastrových prostriedkov QcstChgClusterResourceServices (Change Cluster Resource Services)
<b>Vytvorenie klastra</b> Vytvára nový klaster s jedným alebo viacerými uzlami.	CRTCLU	QcstCreateCluster (Create Cluster)
<b>Vymazanie klastra</b> Odstraňuje existujúci klaster. Služby klastrových prostriedkov sú ukončené na všetkých aktívnych klastrových uzloch a sú z klastra odstánené.	DLTCLU	QcstDeleteCluster (Delete Cluster)

Tabuľka 8. Opisy príkazov CL a rozhraní API na riadenie klastra (pokračovanie)

Popis	Príkaz jazyka CL pre riadenie klastra	Názov API riadenia klastra
<p><b>Ukončenie uzla klastra</b>                      Na jednom alebo všetkých uzloch v zozname členov existujúceho klastra ukončí služby klastrových prostriedkov. Pre klaster prestane byť uzol k dispozícii, až kým nie je reštartovaný použitím rozhrania Spusti uzol klastra (Start Cluster Node).</p>	ENDCLUNOD	QcstEndClusterNode (End Cluster Node)
<p><b>Výpis informácií o klastrí, Zobrazenie informácií o klastrí</b>                      Opakovane získava informácie o klastrí. Napríklad môže byť vrátený celý zoznam členov klastra.</p>	DSPCLUINF	QcstListClusterInfo (List Cluster Information)
<p><b>Výpis informácií o doméne zariadení, Zobrazenie informácií o klastrí</b>                      Vypisuje informácie domény zariadenia klastra. Napríklad môže byť vrátený zoznam práve definovaných domén zariadenia.</p>	DSPCLUINF	Vypísať informácie domény zariadenia QcstListDeviceDomainInfo (List Device Domain Information)
<p><b>Odstránenie položky uzla klastra</b>                      Odstraňuje uzol zo zoznamu členov klastra. Uzol je odstránený zo všetkých obnovovacích domén, operácie klastra sú na tomto uzle ukončené a všetky objekty služieb klastrových prostriedkov sú z uzla odstránené.</p>	RMVCLUNODE	QcstRemoveClusterNodeEntry (Remove Cluster Node Entry)
<p><b>Odstránenie položky domény zariadení</b>                      Odstraňuje uzol zo zoznamu členov domény zariadenia. Ak je to posledný uzol domény zariadenia, potom takisto z klastra odstraňuje aj doménu zariadenia.</p>	RMVDEVDMNE	QcstRemoveDeviceDomainEntry (Remove Device Domain Entry)
<p><b>Získanie informácií o klastrí, Zobrazenie informácií o klastrí</b>                      Opakovane získava informácie o klastrí. Napríklad môže byť vrátený názov klastra a aktuálna verzia klastra.</p>	DSPCLUINF	QcstRetrieveClusterInfo (Retrieve Cluster Information)
<p><b>Získanie informácií o službách klastrových prostriedkov, Zobrazenie informácií o klastrí</b>                      Opakovane získava informácie o ladení výkonu služieb klastrových prostriedkov a o konfiguračných parametroch.</p>	DSPCLUINF	QcstRetrieveCRSInfo (Retrieve Cluster Resource Services Information)
<p><b>Spustenie uzla klastra</b>                      Spúšťa služby klastrových prostriedkov na uzle, ktorý je súčasťou klastra, ale práve nie je aktívny.</p>	STRCLUNOD	QcstStartClusterNode (Start Cluster Node)
<p><b>Práca s klastrom</b>                      Zobrazenie uzlov a objektov klastra a práca s nimi.</p>	WRKCLU	žiadny

Tabuľka 9. Opis príkazov CL a rozhraní API pre skupiny prostriedkov klastra

Popis	Príkaz jazyka CL pre riadenie skupiny klastrových prostriedkov	Skupina klastrových prostriedkov API
<b>Pridanie položky zariadenia skupiny prostriedkov klastra</b> Pridáva novú zadanú hodnotu zariadenia ku skupine prostriedkov klastra. Zariadenie sa tak stáva členom skupiny prepínateľných zariadení.	ADDCRGDEVE	QcstAddClusterResourceGroupDevice (Add Cluster Resource Group Device Entry)
<b>Pridanie uzla do domény obnovy, Pridanie položky uzla skupiny prostriedkov klastra</b> Pridáva nový uzol do obnovovacej domény existujúcej skupiny klastrových prostriedkov.	ADDCRGNODE	QcstAddNodeToRcvyDomain (Add Node to Recovery Domain)
<b>Zmena skupiny klastrových prostriedkov</b> Mení atribúty skupiny klastrových prostriedkov. Napríklad môže byť modifikovaná preberacia adresa IP pre aplikáciu CRG.	CHGCRG	Zmeniť skupinu klastrových prostriedkov QcstChangeClusterResourceGroup (Change Cluster Resource Group)
<b>Zmena položky zariadenia skupiny prostriedkov klastra</b> Mení zadanú hodnotu zariadenia v skupine klastrových prostriedkov. Týmto napríklad môže byť modifikovaná voľba pre zmenu online konfigurácie objektu pri prepínaní alebo preberaní služby.	CHGCRGDEVE	Zmeniť zadanú hodnotu zariadenia pre skupinu klastrových prostriedkov QcstChangeClusterResourceGroupDev (Change Cluster Resource Group Device Entry)
<b>Vytvorenie skupiny klastrových prostriedkov</b> Vytvára objekt v skupine klastrových prostriedkov. Tento objekt identifikuje obnovovaciu doménu, tvorenú množinou uzlov v klastri, ktorý bude zohrávať úlohu v obnovovaní.	CRTCRG	QcstCreateClusterResourceGroup (Create Cluster Resource Group)
<b>Zmazanie skupiny klastrových prostriedkov</b> Vymaže skupinu prostriedkov klastra (CRG) len v lokálnom uzle. Vymazanie lokálnej skupiny prostriedkov klastra vyžaduje, aby boli služby klastrových prostriedkov neaktívne.	DLTCRG	žiadny
<b>Zmazanie skupiny klastrových prostriedkov, Vymazanie klastra CRG</b> Odstraňuje skupinu klastrových prostriedkov z daného klastra. Zo všetkých aktívnych uzlov v doméne obnovy sa vymaže objekt CRG.	DLTCRGCLU	QcstDeleteClusterResourceGroup (Delete Cluster Resource Group)
<b>Distribúcia informácií</b> Doručuje informácie z uzla v obnovovacej doméne CRG do ďalších uzlov tejto domény.	žiadny	QcstDistributeInformation (Distribute Information)

Tabuľka 9. Opisy príkazov CL a rozhraní API pre skupiny prostriedkov klastra (pokračovanie)

Popis	Príkaz jazyka CL pre riadenie skupiny klastrových prostriedkov	Skupina klastrových prostriedkov API
<p><b>Ukončenie skupiny klastrových prostriedkov</b> Zabraňuje odolnosti voči chybám zadanej skupiny klastrových prostriedkov. Pre úspešné dokončenie tejto API je stav skupiny klastrových prostriedkov nastavený do neaktívnej polohy.</p>	ENDCRG	QcstEndClusterResourceGroup (End Cluster Resource Group)
<p><b>Inicializácia prepnutia, Zmena primárneho uzla skupiny prostriedkov klastra</b> Spôsobuje správne prepnutie, ktoré bude vykonané pre skupinu klastrových prostriedkov. Obnovovacia doména je zmenená tak, že aktuálny primárny uzol sa stáva posledným zálohovým a aktuálny prvý zálohový uzol sa stáva novým primárnym.</p>	CHGCRGPRI	QcstInitiateSwitchover (Initiate Switchover)
<p><b>Výpis skupín prostriedkov klastra, Zobrazenie informácií o skupine prostriedkov klastra</b> Generuje zoznam skupín klastrových prostriedkov a niektoré informácie o tejto skupine v danom klastru.</p>	DSPCRGINF	QcstListClusterResourceGroups (List Cluster Resource Groups)
<p><b>Výpis informácií o skupine prostriedkov klastra, Zobrazenie informácií o skupine prostriedkov klastra</b> Vracia obsah objektu v skupine klastrových prostriedkov. Napríklad môže byť vrátená obnovovacia doména s aktuálnymi úlohami uzla.</p>	DSPCRGINF	QcstListClusterResourceGroupInf (List Cluster Resource Group Information)
<p><b>Odstránenie položky zariadenia skupiny prostriedkov klastra</b> Odstraňuje zadanú hodnotu zariadenia zo skupiny prostriedkov klastra. Toto zariadenie tak už nebude prostriedkom prepínania.</p>	RMVCRGDEVE	Odstrániť zadanú hodnotu zariadenia zo skupiny klastrových prostriedkov QcstRemoveClusterResourceGroupDev (Remove Cluster Resource Group Device Entry)
<p><b>Odstránenie uzla z domény obnovy, Odstránenie položky uzla skupiny prostriedkov klastra</b> Odstraňuje uzol z obnovovacej domény existujúcej skupiny klastrových prostriedkov. Uzol sa už naďalej nebude zúčastňovať obnovovacích akcií pre túto skupinu prostriedkov.</p>	RMVCRGNODE	QcstRemoveNodeFromRcvyDomain (Remove Node From Recovery Domain)
<p><b>Spustenie skupiny klastrových prostriedkov</b> Povoľuje odolnosť voči chybám pre špecifikovanú skupinu klastrových prostriedkov. V rámci tohto klastra sa táto skupina stáva aktívnou.</p>	STRCRG	QcstStartClusterResourceGroup (Start Cluster Resource Group)



**Poznámka:** Služby prostriedkov klastra tiež poskytujú množinu vzorových príkladov v knižnici QUSRTOOL, ktoré sa mapujú na príkazy CL a rozhrania API, uvedené hore. Príkazy QUSRTOOL môžu byť v niektorých prostrediach užitočné. Napríklad môže byť veľmi jednoducho nastavený klaster pre testovanie aplikácií, umožňujúcich vytvorenie klastra. Ak chcete získať viac informácií o týchto ukázkových príkazoch, pozrite si člen TCSTINFO v súbore QUSRTOOL/QATTINFO.

Tabuľka 10. *Opisy príkazov CL pre administratívnu doménu*

Popis	Príkaz CL pre administratívnu doménu	Rozhrania API pre administratívnu doménu
<p><b>Vytvorenie administratívnej domény</b>            Vytvorí CRG s partnerskými uzlami, ktoré reprezentuje administratívnu doménu klastra. Po vytvorení administratívnej domény klastra do nej môžete pridať položky monitorovaných prostriedkov (MRE) na synchronizáciu zmien prostriedkov.  <b>Poznámka:</b> Po vytvorení administratívnej domény klastra ju môžete manažovať pomocou príkazov pre CRG z časti Tabuľka 9 na strane 76.</p>	CRTADMDMN	Žiadne
<p><b>Vymazanie administratívnej domény</b>            Vymaže CRG s partnerskými uzlami, ktoré reprezentuje administratívnu doménu klastra. Po dokončení sa z domény odstráni všetky položky MRE a zmeny prostriedkov, ktoré sa monitorovali, sa už nebudú šíriť.</p>	DLTADMDMN	Žiadne

Tabuľka 11. *Opisy rozhraní API integrovaného operačného systému.* Okrem týchto príkazov CL pre administratívnu doménu klastra existuje tiež niekoľko aplikačných programových rozhraní (API) integrovaného operačného systému, pomocou ktorých môžete pridať a odstrániť položky monitorovaných prostriedkov.

Popis	Príkazy CL <sup>1</sup>	API integrovaného operačného systému
<p><b>Pridanie položky monitorovaného prostriedku</b>            Pridá položku monitorovaného prostriedku pre prostriedok systému a jeho atribúty.</p>	Žiadne	QfpadAddMonitoredResourceEntry (Add Monitored Resource Entry)
<p><b>Odstránenie položky monitorovaného prostriedku</b>            Odstráni položku monitorovaného prostriedku (MRE) z adresára monitorovaných prostriedkov.</p>	Žiadne	QfpadRmvMonitoredResourceEntry (Remove Monitored Resource Entry)



Tabuľka 11. Opisy rozhraní API integrovaného operačného systému (pokračovanie). Okrem týchto príkazov CL pre administratívnu doménu klastra existuje tiež niekoľko aplikačných programových rozhraní (API) integrovaného operačného systému, pomocou ktorých môžete pridať a odstrániť položky monitorovaných prostriedkov.

Popis	Príkazy CL <sup>1</sup>	API integrovaného operačného systému
<b>Získanie informácií o monitorovanom prostriedku</b> Vrátí informácie o monitorovaných prostriedkoch.	Žiadne	QfpadRtvMonitoredResourceInfo (Retrieve Monitored Resource Information)
<b>Poznámka:</b> 1. Pre túto funkciu neexistuje žiadny podporovaný ekvivalentný príkaz CL. Knižnica QUSRTOOL obsahuje zdroj nepodporovaný príkaz a program, spracúvajúci príkaz (CPP). Ak sa chcete dozvedieť o tomto zdroji príkazov a CPP, pozrite si člen QFPADINFO v súbore QATTINFO.		

#### Súvisiaci odkaz

Rozhrania API pre klastre

## Obchodní partneri IBM poskytujúci softvér strednej úrovne pre klastre a dostupné produkty pre klastrovanie

Od obchodných partnerov IBM poskytujúcich softvér strednej úrovne pre klastre môžete získať produkt zabezpečujúci funkcie logickej replikácie, ktoré sú integrálnou súčasťou klastrovania a zjednodušujú vytváranie a manažovanie klastrov.

Obchodní partneri IBM, ktorí poskytujú softvér strednej úrovne pre klastre, dodávajú softvérové riešenia pre vyhradené funkcie replikácie a manažovania klastrov. Ak chcete získať produkt zabezpečujúci funkcie logickej replikácie, ktoré sú integrálnou súčasťou klastrovania a zjednodušujú vytváranie a manažovanie klastrov, kontaktujte vášho obchodného zástupcu alebo obchodného partnera IBM. Môžu vám poskytnúť zoznam produktov pre klastrovanie, ktoré dodávajú obchodní partneri IBM pre softvér strednej úrovne pre klastre.

### Produkt na manažovanie klastrov od obchodného partnera IBM dodávajúceho softvér strednej úrovne pre klastre:

- Poskytuje užívateľské rozhranie na definovanie a údržbu konfigurácie klastra
- Poskytuje užívateľské rozhranie na definovanie a správu skupín klastrových prostriedkov zariadenia, údajov a aplikácie
- Udržiava vedomosti o používaní API klastra, o to, aké skupiny klastrových prostriedkov sú definované v klastri a aké vzťahy sú požadované.
- Vytvára skupiny klastrových prostriedkov zariadení, údajov a aplikácií.

### Produkt na replikáciu od obchodného partnera IBM dodávajúceho softvér strednej úrovne pre klastre:

- Vytvára riadiace štruktúry, ktoré identifikujú dáta a objekty, u ktorých je požadovaná obnoviteľnosť.
- Vytvára skupinu klastrových prostriedkov pre kritické údaje a pričleňuje tento objekt k jeho riadiacej štruktúre.
- Poskytuje ukončovaci program pre skupinu klastrových prostriedkov údajov.

#### Súvisiace úlohy

“Pridanie uzla do klastra” na strane 99

Uzol môžete do klastra pridať pomocou Navigátora iSeries alebo príkazov.

## Požiadavky klastra

Opisuje požiadavky na hardvér, softvér a komunikáciu pri používaní klastrov.

Požiadavky pre používanie klastrov závisia od toho, ktoré schopnosti klastra chcete používať. Môžete napríklad zvoliť použitie jednoduchého klastra s dvoma uzlami a využívať logickú replikáciu. Alebo môžete zvoliť použitie klastra navrhnutého na využívanie prepínateľných diskov a prepínateľných nezávislých diskových oblastí.

### Súvisiace koncepty

“Príklad: Konfigurácie klastra” na strane 115

Tieto príklady typických implementácií klastrov vám pomôžu pochopiť prečo, kedy a ako môže byť užitočné používať klastre.

## Hardvérové požiadavky klastrov

S klastrovaním je kompatibilný každý model iSeries, v ktorom je možné použiť operačný systém i5/OS V4R4M0 alebo novší.

Navyše by ste mali pomocou externého zálohového napájacieho zdroja alebo niečoho podobného, zabezpečiť ochranu pred výpadkom napájania. V opačnom prípade môže náhly výpadok napájania namiesto núdzového prepnutia spôsobiť stav rozkladu klastra.

Klastrovanie používa schopnosti vysielania viacerým (multicast) protokolu IP (Internet Protocol). Multicast nemapuje všetky typy fyzických médií. Viac informácií o obmedzeniach pre vysielanie typu multicast, ktoré môžu platiť pre váš hardvér, nájdete v téme Nastavenie TCP/IP.

Ak plánujete vo vašom klastrí používať nezávislé diskové oblasti, pozrite si časť o nezávislých diskových oblastiach v téme Hardvérové požiadavky. Vaše disky môžete chrániť tiež pomocou ochrany zrkadlením alebo pomocou ochrany zariadení paritou. Použitím týchto riešení na vašom primárnom systéme sa vyhnete zotaveniu po zlyhaní, ak by váš chránený disk zlyhal. Je vhodné mať takéto riešenia aj na vašom záložnom systéme, pre prípad, že by sa vyskytlo zotavenie po zlyhaní. Podrobnosti nájdete v časti Ochrana disku.

- | **Poznámka:** Detaily o ďalších požiadavkách, ktoré musíte splniť pred konfigurovaním klastrov, nájdete v téme  
| “Zoznam klastrovej konfigurácie” na strane 90.

### Súvisiace koncepty

Nepreerušiteľný zdroj energie

“Oddiel klastra” na strane 27

*Oddiel klastra* je podskupina aktívnych uzlov klastra, ktorá vzniká následkom zlyhania spojenia. Členovia podskupiny si medzi sebou udržiavajú vzájomnú pripojiteľnosť.

“Zlyhanie” na strane 17

K *núdzovému prepnutiu* dochádza, keď server v klastrí vykoná pri zlyhaní systému automatické prepnutie na jeden alebo viac záložných serverov.

## Požiadavky na softvér a licencie pre klastre

Aby ste mohli používať klastrovanie, musíte mať správny softvér a licencie.

1. Operačný systém OS/400 V4R4M0 alebo novší s nakonfigurovaným TCP/IP (TCP/IP Connectivity Utilities).
2. Konfigurácia klastra a riadenie softvérového riešenia. Môže to byť ľubovoľný z týchto produktov:
  - Manažovanie klastrov v Navigátore iSeries
  - Riešenie softvéru na strednej úrovni pre klastre od obchodného partnera IBM.
  - Váš vlastný aplikačný program riadenia klastra, ktorý je vytvorený s použitím príkazov služieb prostriedkov a API.
3. Pozrite si tému “Zoznam klastrovej konfigurácie” na strane 90

**Dôležité:** Ak plánujete používať nezávislé diskové oblasti, aby ste mohli využiť prepínateľné zariadenia, musíte splniť ďalšie požiadavky. Podrobnejšie informácie nájdete v téme Plánovanie nezávislých diskových oblastí.

### Súvisiace koncepty

“Riešenia konfigurácie a správy klastrov” na strane 72

Služby klastrových prostriedkov poskytujú základnú infraštruktúru klastrov. Existuje niekoľko metód, ktoré vám dovoľia využiť výhody schopností klastrovania, poskytované službami klastrových prostriedkov.

“Verzia klastra” na strane 13

*Verzia klastra* reprezentuje úroveň funkcií dostupných v klastrí.

## Komunikačné požiadavky klastrov

Vo vašom prostredí s klastrami môžete použiť ľubovoľný typ komunikačného média, ktorý podporuje protokol IP (Internet Protocol).

Služby klastrových prostriedkov používajú na komunikáciu medzi uzlami výhradne protokoly TCP/IP. Podporované sú lokálne siete (LAN), rozšírené siete (WAN), siete so systémom OptiConnect (SAN), alebo akékoľvek kombinácie týchto pripájacích zariadení. Pri výbere by ste mali zväziť tieto faktory:

- Objeme transakcií
- Požiadavkách na dobu odozvy
- Vzdialenosti medzi uzlami
- Nákladoch

Rovnaké úvahy môžete použiť pri výbere média pripojenia, ktoré má byť použité na prepojenie primárnych a záložných umiestnení prostriedkov. Pri plánovaní vášho klastra vám odporúčame, aby ste si naprojektovali jeden, alebo viac záložných uzlov na vzdialených umiestenách, čím sa predídete úplnému zlyhaniu pri havárii s následkom úplného zničenia jedného miesta.

Aby ste sa vyhli problémom výkonu, ktoré by mohli byť zapríčinené nedostatočnou kapacitou, potrebujete zväziť použitie komunikačného média schopného zvládnuť objem informácií posielaných medzi uzlami. Môžete sa rozhodnúť, ktoré fyzické médium uprednostníte, ako napríklad token ring, Ethernet, režim asynchrónneho prenosu (ATM), SPD OptiConnect, vysokorýchlostnú linku (HSL) OptiConnect alebo Virtuálny OptiConnect (a vysokorýchlostné interné spojenie medzi logickými oddielmi).

HSL OptiConnect je technológia, ktorú poskytuje softvér OptiConnect for i5/OS (Voľba 23 systému i5/OS - i5/OS OptiConnect). Je vhodný pri tvorbe riešení s vysokou dostupnosťou. HSL OptiConnect je systémová sieť, ktorá pomocou technológie High Speed Link (HSL) Loop poskytuje vysokorýchlostné pripojenie typu point-to-point medzi klastrovými uzlami. HSL OptiConnect si vyžaduje štandardné káble HSL, ale nepotrebuje žiaden dodatočný hardvér.

Pre prepínateľný hardvér, tiež nazývaný CRG obnoviteľných zariadení, potrebujete nezávislú diskovú oblasť, ktorá je vo vašom prostredí prepínateľná. V prostredí logických oddielov je to skupina diskových jednotiek, ktoré sú na zbernici, ktorá je zdieľaná logickými oddielmi, alebo ktoré sú pripojené k vstupno/výstupnému procesoru, ktorý bol priradený k I/O oblasti. V prostredí viacerých systémov je to jedna, alebo viac prepínateľných rozširujúcich jednotiek (veží) správne nakonfigurovaných na slučku HSL, na ktorých sú systémy domény obnovy. Prepínateľná veža sa môže tiež použiť v prostredí LPAR. Viac informácií o plánovaní pre prepínateľný hardvér a nezávislé diskové oblasti nájdete v časti Plán pre nezávislé diskové oblasti.

**Poznámka:** Ak používate sieťové adaptéry 2810 len s protokolom TCP/IP a nepoužívate architektúru SNA (Systems Network Architecture) ani IPX, môžete výkon adaptéra v serveri V4R5M0 zvýšiť tak, že pre špecifický opis linky povolíte pomocou príkazu WRKLIND (Work with Line Descriptions) len protokol TCP (\*YES). Povolí len pre TCP(\*YES) je vo vydaniach V5R1M0 a vyšších nastavené automaticky.

### Súvisiace informácie

OptiConnect for i5/OS

## Návrh klastra

Určuje, čo potrebujete pri určovaní, ako projektovať váš klaster.

V závislosti od požadovaného výsledku existuje viacero spôsobov použitia klastrovania, preto je dôležité stráviť určitý čas identifikovaním vašich potrieb, aby ste zistili, ako navrhnuť váš klaster.

## Návrh siete pre klastre

Skôr, než budete svoju sieť konfigurovať pre klastrovanie, musíte si starostlivo naplánovať a vykonať niektoré úvodné predklastrové konfigurácie týkajúce sa TCP/IP.

Je dôležité, aby ste si prečítali tieto témy skôr, ako budete konfigurovať váš klaster. Povedia vám kedy a ako:

- Nastavenie adres IP
- Nastavenie konfiguračných atribútov TCP/IP
- Zabránenie rozdeleniu klastra

Informácie o nastavovaní redundantných komunikačných ciest a o tom, či potrebujete pre klastrovanie vyhradenú sieť, nájdete v téme *Vyhradenie siete pre klastre*.

### **Nastavenie adres IP:**

Pretože služby klastrových prostriedkov používajú na komunikáciu s inými uzlami klastra *len* IP, všetky uzly klastra musia byť *dosiahnuteľné cez IP*.

To znamená, že musíte mať rozhrania IP nakonfigurované na spojenie s uzlami vo vašom klastri. Tieto IP adresy musia byť nastavené buď manuálne správcom siete v smerovacích tabuľkách TCP/IP na každom uzle klastra alebo môžu byť generované smerovacími protokolmi, ktoré bežia na smerovačoch v sieti. Táto smerovacia tabuľka TCP/IP je mapou, ktorú klastrovanie používa na vyhľadanie každého uzla; preto musí mať každý uzol vlastnú *unikátnu* IP adresu. Každý uzol môže mať priradené maximálne dve IP adresy. Tieto adresy nesmú byť zmenené v žiadnom prípade inými aplikáciami sieťových komunikácií. Pri priradovaní každej adresy berte ohľad na to, aký druh komunikačnej linky používa daná adresa. Ak preferujete používanie špecifického typu komunikačného média, zabezpečte, aby prvá nakonfigurovaná IP adresa používala vami preferované médium. Prvá adresa IP je tá, ktorú preferuje funkcia garantovaných správ a monitorovanie kontrolného signálu. Všetky adresy IP v uzle musia byť schopné dosiahnuť všetky ostatné adresy IP v klastri. Adresa môže dosiahnuť inú adresu vtedy, keď môžete v oboch smeroch použiť príkaz ping a sledovanie trasy správ UDP.

**Poznámka:** Ubezpečte sa, že adresa spätnej slučky (127.0.0.1) je aktívna a pripravená na klastrovanie. Táto adresa, ktorá sa používa na odosielanie správ späť na lokálny uzol, je štandardne aktívna. Ak však bola ukončená omylom, nemôže klastrové zasielanie správ znova fungovať, kým táto adresa nebude reštartovaná.

#### **Súvisiace koncepty**

“Funkcia spoľahlivých správ” na strane 26

*Funkcia spoľahlivých odkazov* služby klastrových prostriedkov sleduje každý uzol v klastri a zabezpečuje, že všetky uzly majú nepretržité informácie o stave klastrových prostriedkov.

“Sledovanie aktivity” na strane 25

*Monitorovanie kontrolného signálu* je funkcia služieb klastrových prostriedkov, ktorá zabezpečuje, že každý uzol je aktívny. Funkcia odosiela z každého uzla v klastri signál do všetkých ostatných uzlov v klastri na potvrdenie, že sú uzly stále aktívne.

### **Nastavenie konfiguračných atribútov TCP/IP:**

Aby boli umožnené služby klastrových prostriedkov, je potrebné nastaviť určité atribúty v konfigurácii TCP/IP vo vašej sieti.

Skôr ako budete môcť pridať nejaký uzol do klastra, musíte nastaviť tieto atribúty :

- Ak plánujete ako smerovač pre komunikáciu s ostatnými sieťami používať server iSeries a v danom serveri nepoužívate žiadne iné smerovacie protokoly, pomocou príkazu CHGTCPA (Change TCP/IP Attributes) nastavte postupovanie IP datagramov na hodnotu \*YES.
- Nastavte server INETD na START. Informácie o spúšťaní servera INETD nájdete v téme Server INETD.
- Pomocou príkazu CHGTCPA (Change TCP/IP Attributes) nastavte hodnotu pre kontrolný súčet UDP (User Datagram Protocol) na \*YES.
- Nastavte posielanie MCAST na \*YES, ak používate mosty na spojenie vašich sietí typu token ring.
- Ak na komunikáciu medzi uzlami klastra používate OptiConnect for i5/OS, pomocou príkazu STRSBS(QSOC/QSOC) spustíte podsystém QSOC.

### **Tipy: Klastrová komunikácia:**

Pri nastavovaní komunikačných ciest zvážte tieto tipy.

- Uistite sa, že vaše komunikačné linky majú dostatočnú šírku pásma na to, aby zvládli neklastrovú aktivitu spolu s funkciou monitorovania aktivity a neprestávajú sledovať prípadné zvýšenie aktivity.
- Pre zvýšenú spoľahlivosť nekonfigurujte jedinú komunikačnú cestu na spojenie jedného, alebo viacerých uzlov.
- Nepreťažte linku, ktorá je zodpovedná za zaistenie stálej komunikácie s uzlom.
- Eliminujte toľko kritických bodov, ako bude možné, napríklad dve komunikačné linky vchádzajúce do jediného adaptéra, do rovnakého vstupno-výstupného procesora (IOP) alebo do rovnakej veže.
- Ak cez vaše komunikačné linky prechádza extrémne veľký objem údajov, mali by ste zvážiť umiestnenie replikácie údajov a monitorovania kontrolného signálu do samostatných sietí.
- Ak používate vysielanie viacerým (multicast) pomocou protokolu IP (Internet Protocol), mali by ste si pozrieť publikáciu TCP/IP Configuration and Reference, kde nájdete obmedzenia pre tento typ prenosu, týkajúce sa rôznych typov fyzických médií.
- User Datagram Protocol (UDP) multicast je preferovaný protokol, ktorý infraštruktúra klastrovej komunikácie používa na posielanie informácií o správe klastra medzi uzlami v klastri. Keď fyzické médium podporuje multicasting, klastrové komunikácie využívajú UDP multicast na posielanie riadiacich správ z daného uzla na všetky lokálne uzly klastra, ktoré podporujú rovnakú adresu podsiete. Správy, ktoré sú odosielané k uzlom vo vzdialených sieťach sú vždy posielané prostredníctvom schopnosti UDP point-to-point. Klastrové komunikácie sa nespoliehajú na smerovacie schopnosti pre multicastové správy.
- Pre multicastovú prevádzku, ktorá podporuje posielanie riadiacich správ klastra, je prirodzené, že má tendencie ku kolísaniu. V závislosti od počtu uzlov v danej LAN (ktorá podporuje spoločnú adresu podsiete) a komplexnosti štruktúry riadenia klastra, ktorá je vybratá správcom klastra, môžu multicastové pakety vzťahujúce sa ku klastru prekročiť i 40 paketov za sekundu. Kolísania tohto druhu môžu mať negatívny dopad na staršie sieťové zariadenia. Príkladom môžu byť problémy s preťažením zariadení v LAN, ktoré slúžia ako agenti protokolu SNMP (Simple Network Management Protocol) s potrebou vyhodnocovať každý paket UDP poslaný viacerým. Niektoré zo skorších sieťových vybavení nemajú adekvátnu šírku pásma, aby držali krok s týmto typom prevádzky. Mali by ste zabezpečiť, že vy alebo správca siete posúdite kapacitu sietí s premávkou UDP multicast, aby ste zaručili, že klastrovanie nebude mať negatívny dopad na výkonnosť týchto sietí.

#### **Súvisiace koncepty**

“Plán pre logickú replikáciu” na strane 87

Pri logickej replikácii sa uchováva viacero kópií údajov. Údaje sú replikované, alebo kopírované z primárneho uzla v klastri na uzly, ktoré sú v doméne obnovy určené ako záložné. Keď dôjde na primárnom uzle k výpadku, ostávajú údaje dostupné, keď záložný uzol prevezme primárny bod prístupu.

“Funkcia spoľahlivých správ” na strane 26

*Funkcia spoľahlivých odkazov* služby klastrových prostriedkov sleduje každý uzol v klastri a zabezpečuje, že všetky uzly majú nepretržité informácie o stave klastrových prostriedkov.

#### **Súvisiace informácie**

Nastavenie TCP/IP

#### **Zabránenie rozdeleniu klastra:**

Typickému rozkladu klastra súvisiacemu so sieťou môžete najlepšie predísť tak, že medzi všetkými uzlami klastra nakonfigurujete redundantné komunikačné cesty.

*Redundantná komunikačná cesta* znamená, že sú nakonfigurované dve linky medzi dvoma uzlami v klastri. Ak zlyhá prvá komunikačná cesta, komunikáciu môže prevziať druhá cesta, aby sa zachovala funkčnosť komunikácie medzi uzlami a obmedzili sa podmienky, ktoré môžu spôsobiť izolovanie jedného alebo viacerých uzlov do oddielu klastra. Pri konfigurácii týchto ciest ale nesmiete zabudnúť na to, že ak obe komunikačné cesty vedú k tomu istému adaptéru v sieti, stále je riziko, že zlyhá práve tento adaptér.

- | Treba však podotknúť, že nie vždy sa dá predísť rozkladu klastra. Ak sa preruší napájanie systému alebo dôjde k poruche hardvéru, klaster sa môže rozložiť.

#### **Súvisiace koncepty**

“Oddiel klastra” na strane 27

*Oddiel klastra* je podskupina aktívnych uzlov klastra, ktorá vzniká následkom zlyhania spojenia. Členovia podskupiny si medzi sebou udržiavajú vzájomnú pripojiteľnosť.

“Tipy: Klastrová komunikácia” na strane 82

Pri nastavovaní komunikačných ciest zvážte tieto tipy.

“Chyby oddielov” na strane 134

Určité klastrové stavy je možné jednoducho opraviť. Ak došlo k rozkladu klastra, dozviete sa tu, ako vykonať obnovu. Tiež sa dozviete, ako môžete zabrániť deleniu klastra a uvedie príklady, ako oddiely spojiť naspäť.

### Vyhradenie siete pre klastre:

Počas normálnych operácií bude úroveň základnej klastrovej komunikácie minimálna. Odporúčame vám ale, aby ste pre každý uzol klastra konfigurovali redundantné komunikačné cesty.

V konfigurácii dvoch liniek je možné určiť jednu linku na klastrovanie prevádzky a druhá linka môže spravovať normálne operácie a takisto zálohovanie, pokiaľ klastrovacia (prvá) linka zlyhá.

#### Súvisiace koncepty

“Zabránenie rozdeleniu klastra” na strane 83

Typickému rozkladu klastra súvisiacemu so sieťou môžete najlepšie predísť tak, že medzi všetkými uzlami klastra nakonfigurujete redundantné komunikačné cesty.

## Klastre s viacerými vydaniaми

Ak vytvárate klastre, ktorý bude obsahovať uzly s rôznymi verziami klastrovania, musíte vykonať určité kroky.

Štandardne je ako aktuálna verzia klastra stanovená potenciálna verzia klastra prvého pridaného uzla. Tento prístup je správny, ak je potenciálna verzia klastra tohto uzla v klastri najnižšia. Ak je však úroveň verzie tohto uzla vyššia, nebudete ďalej môcť pridávať uzly s nižšou úrovňou verzie. Alternatívou je pomocou hodnoty cieľovej verzie klastra na vytvorenom klastri nastaviť aktuálnu verziu klastra na hodnotu nižšiu, než je potenciálna verzia klastra prvého pridaného uzla.

Predstavte si, napríklad, že vytvárate dvojuzlový klastre. Uzly pre tento klastre sú:

Identifikátor uzla	Vydanie	Potenciálna verzia klastra
Uzol A	V5R3	4
Uzol B	V5R4	5

Ak by bol klastre vytvorený Uzlom B, museli by ste sa si dať pozor, aby bolo jasné, že ide o klastre s rôznymi vydaniaми. Musí byť nastavená cieľová verzia klastra, aby bolo zabezpečené, že uzly v klastri budú komunikovať verziami nižšou, než bola potenciálna verzia vyzývajúceho uzla.

#### Súvisiace koncepty

“Verzia klastra” na strane 13

*Verzia klastra* reprezentuje úroveň funkcií dostupných v klastri.

## Identifikovanie serverov patriacich do klastra

Na identifikáciu serverov na zahrnutie do klastra potrebujete určiť, ktoré servery sú schopné poskytnúť adekvátnu zálohu pre údaje a aplikácie, ktoré sú potrebné pre vašu spoločnosť.

Je potrebné určiť:

- Ktoré servery obsahujú vaše kritické údaje a aplikácie?
- Ktoré servery budú tieto systémy zálohovať?

Keď ste si odpovedali na tieto otázky, máte zoznam serverov, ktoré chcete pridať do vášho klastra.



## | **Porovnanie modelu s primárnymi/záložnými uzlami a modelu s partnerskými uzlami**

| Skupiny CRG s primárnymi/záložnými uzlami a skupiny CRG s partnerskými uzlami zabezpečujú obnoviteľnosť pre prostriedky v klastri; je však dôležité porozumieť ich rozdielom a použitiu.

| Klustre podporujú dva modely definovania skupín CRG vo vašom prostredí. Pre modely s primárnymi/záložnými uzlami aj s partnerskými uzlami sú definované roly. V modeli s primárnymi/záložnými uzlami je nutné definovať poradie. Uzly, ktoré sú definované ako záložné, poskytujú prístup k prostriedkom primárneho uzla v prípade, že tento uzol zlyhá. V modeli s partnerskými uzlami sú všetky uzly rovnocenné a môžu poskytovať prístup k prostriedku; neexistuje tu koncept poradia.

## | **Model s primárnymi/záložnými uzlami**

| V prípade modelu s primárnymi/záložnými uzlami musia užívatelia definovať pre uzol rolu primárneho, záložného alebo replikovaného uzla. Tieto roly sú definované a manažované v rámci domény obnovy. Ak bol uzol definovaný ako primárny prístupový bod k prostriedku, ostatné uzly poskytujú zálohu v prípade, že primárny uzol zlyhá.

## | **Model s partnerskými uzlami**

| Skupiny CRG používajúce model s partnerskými uzlami eliminujú potrebu definovať doménu obnovy s poradím. V prípade modelu s partnerskými uzlami môžu byť uzly definované ako partnerské alebo replikované. Ak sú uzly definované ako partnerské, všetky uzly v doméne obnovy sú rovnocenné a poskytujú prístupový bod k prostriedku.

## | **Identifikovanie aplikácií patriacich do klastra**

| Nie každá aplikácia vám poskytne výhody dostupnosti, ktoré dosiahnete klastrovaním.

| Aby ste mohli využívať možnosti prepnutia a zotavenia po zlyhaní, ktoré vám klastrovanie ponúka, musí byť aplikácia obnoviteľná. Obnoviteľnosť aplikácie umožňuje reštart aplikácie na záložnom uzle bez toho, aby ste museli rekonfigurovať klientov, ktorí aplikáciu používajú. Preto musí vaša aplikácia pri využívaní možností, ktoré klastrovanie ponúka spĺňať určité požiadavky. Viac informácií o obnoviteľných aplikáciách nájdete v téme Aplikácie klastra.

## | **Plán pre údaje odolné voči zlyhaniu**

| Obnoviteľnosť údajov sa dosiahne, ak sú údaje pre aplikáciu alebo užívateľa vždy dostupné. Obnoviteľnosť údajov môžete dosiahnuť pomocou logickej replikácie alebo prepínateľných nezávislých diskových oblastí.

### | **Určenie, ktoré údaje by mali byť vytvorené ako odolné voči zlyhaniu:**

| Uvedomte si, pri akých typoch údajov by ste mali považovať o obnoviteľnosti.

| Určenie toho, ktoré údaje majú byť replikovateľné, je veľmi podobné tomu, ako keď určujeme, aký typ údajov treba zálohovať, keď pre svoje systémy pripravujete stratégiu zálohovania a obnovy. Je potrebné určiť, ktoré dáta sú pre chod vašej práce najdôležitejšie.

| Ak napríklad pracujete na Internete, kritickými dátami pre vás môžu byť:

- | • Dnešné objednávky
- | • Inventár
- | • Záznamy o zákazníkoch

| Vo všeobecnosti, ak sa informácie nemenia často, alebo ak ich nepotrebuje používať denne, pravdepodobne nemusia byť obnoviteľné.

### | **Súvisiace koncepty**

| Plánovanie stratégie zálohovania a obnovy

## | **Porovnanie logickej replikácie, prepínaných diskov a zrkadlenia lokalít:**

- | Táto téma poskytuje prehľad rôznych technológií obnoviteľnosti údajov, ktoré môžete použiť v klastroch na zvýšenie dostupnosti.
- | *Obnoviteľnosť údajov* umožňuje, aby údaje zostali dostupné pre aplikácie a užívateľov aj keď systém, ktorý ich pôvodne obsahoval, zlyhá. Výber správnej množiny technológií obnoviteľnosti údajov v kontexte celkovej stratégie zachovania prevádzky vášho podniku môže byť komplexný a náročný. Je dôležité oboznámiť sa s rôznymi riešeniami obnoviteľnosti údajov, pomocou ktorých môžete zvýšiť dostupnosť v prostredí s viacerými systémami. Môžete vybrať jedno riešenie alebo použiť kombináciu technológií, ktorá bude zodpovedať vašim požiadavkám.

Viac detailov o týchto riešeniach nájdete v dokumente *Data Resilience Solutions for IBM i5/OS High Availability Clusters*. Časť s názvom "Data Resilience Solutions for IBM i5/OS High Availability Clusters" obsahuje detailné porovnanie atribútov každej z týchto technológií.

## Logická replikácia

*Logická replikácia* je proces kopírovania objektov z jedného uzla klastra do jedného alebo viacerých iných uzlov klastra, takže objekty vo všetkých systémoch budú identické.

Replikované prostriedky umožňujú objektom, ako napríklad aplikáciám a ich údajom, aby boli kopírované z jedného uzla v klastri na iný uzol, alebo iné uzly v klastri. Tento proces udržiava objekty na všetkých serveroch v doméne obnovy identické. Ak vykonáte zmeny na objekte na jednom uzle klastra, sú zmeny replikované na jeho ostatné uzly. Potom, ak by došlo k zotaveniu po zlyhaní alebo k prepnutiu, môže záložný uzol bez prerušenia prevziať úlohu primárneho uzla. Server, alebo server, ktoré pôsobia ako zálohy, sú definované v doméne obnovy. Ak sa na serveri, určenom ako primárny uzol domény obnovy, objaví výpadok a iniciuje sa zotavenie po zlyhaní alebo prepnutí, stane sa primárnym prístupovým bodom uzol, ktorý je v doméne obnovy označený ako záložný.

Replikácia si vyžaduje použitie na mieru napísanej aplikácie, alebo softvérovú aplikáciu napísanú obchodným partnerom z oblasti klastrovania. Detaily nájdete v téme *Plánovanie logickej replikácie*.

## Prepínateľné disky

*Prepínateľné disky* umožňujú prepínať prostriedky (napríklad údaje a aplikácie), umiestnené v rozširujúcej jednotke alebo vstupno-výstupnom procesore (IOP) na zdieľanej zbernici alebo v I/O oblasti logického oddielu, medzi primárnym a záložným uzlom klastra. To umožňuje, aby bola skupina diskových jednotiek prístupná z druhého servera, určeného ako záložný uzol domény obnovy skupiny klastrových prostriedkov, ak sa na serveri, ktorý zatiaľ na tieto diskové jednotky pristupoval, objavil výpadok a jeho zabezpečenie, alebo prepnutie.

Ak chcete vo svojom klastri využívať výhody prepínateľných prostriedkov, musíte používať nezávislé diskové oblasti. Viac informácií nájdete v časti *Plánovanie nezávislých diskových oblastí*.

## Zrkadlenie lokalít

*Zrkadlenie lokalít* vám v kombinácii s funkciou geografického zrkadlenia umožňuje zrkadliť údaje na diskoch v lokalitách, ktoré môžu byť od seba geograficky značne vzdialené. Táto technológia sa môže použiť na rozšírenie funkčnosti skupiny prostriedkov klastra (CRG) zariadenia za obmedzenia fyzického prepojenia komponentov. Geografické zrkadlenie poskytuje schopnosť replikovať zmeny spravené v produkčnej kópii nezávislej diskovej oblasti do zrkadlovej kópie danej nezávislej diskovej oblasti. Pri zapisovaní údajov do produkčnej kópie nezávislej diskovej oblasti, operačný systém zrkadlí údaje do druhej kópie nezávislej diskovej oblasti cez iný systém. Tento proces udržiava viacero identických kópií údajov.

Keď sa má prostredníctvom CRG zariadenia vykonať núdzové prepnutie alebo prepnutie, záložný uzol môže plynulo prebrať rolu primárneho uzla. Server, alebo server, ktoré pôsobia ako zálohy, sú definované v doméne obnovy. Záložné uzly môžu byť v rovnakom alebo inom fyzickom umiestnení ako primárne uzly. Ak dôjde k výpadku servera definovaného ako hlavný uzol v doméne obnovy a inicializuje sa prepnutie alebo núdzové prepnutie, uzol označený ako záložný v doméne obnovy sa stane primárnym prístupovým bodom pre prostriedok a bude vlastniť produkčnú



kópiu nezávislej diskovej oblasti. Takto môžete získať ochranu pred zlyhaním jedného miesta spojeného s prepínateľnými prostriedkami.

Tabuľka 12. Porovnanie technológií obnoviteľnosti údajov, ktoré je možné použiť s klastrami. Dozviete sa o charakteristikách rôznych technológií obnoviteľnosti údajov, čo vám pomôže pri určení najvýhodnejšieho riešenia pre váš klaster.

Faktor	Replikácia	Prepínateľné disky	Zrkadlenie lokalít
Flexibilita	Desiatky serverov	2 servery	4 servery
Jeden bod zlyhania	Žiadne	Podsystemy disku	Žiadne
Náklady	<ul style="list-style-type: none"> <li>Je požadovaná dodatočná kapacita diskov.</li> <li>Softvér na replikáciu od obchodného partnera.</li> <li>Duplicitné disky</li> </ul>	<ul style="list-style-type: none"> <li>Prepínateľná rozširujúca veža I/O alebo IOP</li> <li>Voľba 41</li> </ul>	<ul style="list-style-type: none"> <li>Dodatočný disk pre zrkadlovú kópiu nezávislej diskovej oblasti</li> <li>Voliteľne prepnutelná rozširujúca jednotka I/O</li> <li>Voľba 41</li> </ul>
Výkon	Réžia replikácie	Malý účinok	Réžia geografického zrkadlenia
Pokrytie v reálnom čase	Zaznamenané objekty	Objekty obsiahnuté v nezávislej diskovej oblasti	Objekty obsiahnuté v nezávislej diskovej oblasti
Geografické rozmiestnenie	Obmedzené kritériami výkonu	Obmedzená vzdialenosť pripojenia akou musia byť servery a skladovacie jednotky pripojené k slučke HSL OptiConnect loop (maximálne 250 metrov)	Obmedzené požiadavkami na výkon (Systém nekladie žiadne limity. Doba odozvy a priepustnosť vybraných komunikačných liniek však môžu určovať isté praktické obmedzenie.)
Ochrana pri obnove po havárii	Áno	Nie	Áno
Súbežné zálohovanie	Áno	Nie	Nie
Nastavenie	<ul style="list-style-type: none"> <li>Replikačné prostredie.</li> <li>Určuje, čo replikovať.</li> </ul>	<ul style="list-style-type: none"> <li>Prostredie nezávislej diskovej oblasti.</li> <li>Využitie nezávislej diskovej oblasti</li> </ul>	<ul style="list-style-type: none"> <li>Prostredie nezávislých diskových oblastí (vrátane nastavenia geografického zrkadlenia)</li> <li>Využitie nezávislej diskovej oblasti</li> </ul>

### Súvisiace koncepty

“Plán pre logickú replikáciu”

Pri logickej replikácii sa uchováva viacero kópií údajov. Údaje sú replikované, alebo kopírované z primárneho uzla v klastrí na uzly, ktoré sú v doméne obnovy určené ako záložné. Keď dôjde na primárnom uzle k výpadku, ostávajú údaje dostupné, keď záložný uzol prevezme primárny bod prístupu.

“Plán pre prepínateľné nezávislé diskové oblasti a zrkadlenie lokalít (XSM)” na strane 88

Jediná kópia údajov je udržiavaná na prepínateľnom hardvéri; tým je buď skladovacia jednotka (veža), alebo, v prostredí logických oddielov, IOP.

### Plán pre logickú replikáciu:

Pri logickej replikácii sa uchováva viacero kópií údajov. Údaje sú replikované, alebo kopírované z primárneho uzla v klastrí na uzly, ktoré sú v doméne obnovy určené ako záložné. Keď dôjde na primárnom uzle k výpadku, ostávajú údaje dostupné, keď záložný uzol prevezme primárny bod prístupu.

**Replikácia** vytvorí kópiu niečoho v reálnom čase. Je to proces kopírovania objektov z jedného uzla na iný, alebo iné uzly v klastrí. Replikácia urobí a udržiava objekty na vašom systéme identickými. Ak vykonáte zmeny objektu na jednom uzle klastra, replikujú sa tieto zmeny aj na ostatné uzly.

| Musíte sa rozhodnúť, ktorú softvérovú technológiu použijete pre logickú replikáciu. Na dosiahnutie logickej replikácie v klastrí sú dostupné tieto riešenia:

| • **Produkty od obchodných partnerov IBM**

| Softvér na replikáciu údajov od obchodných partnerov IBM pre klastré vám umožňuje replikovať objekty medzi viacerými uzlami. Detaily nájdete v téme “Obchodní partneri IBM poskytujúci softvér strednej úrovne pre klastré a dostupné produkty pre klastrovanie” na strane 79.

| • **Vlastná aplikácia na replikáciu**

| Manažment žurnálov IBM poskytuje prostriedky, pomocou ktorých môžete zaznamenávať aktivitu objektov vo vašom systéme. Môžete napísať aplikáciu, ktorá bude na dosiahnutie logickej replikácie využívať manažment žurnálov. Viac detailov o fungovaní manažmentu žurnálov nájdete v dokumentácii k manažmentu žurnálov iSeries.

| **Súvisiace koncepty**

| Manažment žurnálov

| *Určenie, ktoré systémy použiť pre logickú replikáciu:*

| Pri určovaní, ktoré systémy použiť pre logickú replikáciu, by ste mali zväziť niekoľko aspektov.

| Sú to tieto aspekty:

- | • Kapacita výkonu
- | • Disková kapacita
- | • Kritické dáta
- | • Prevencia možných porúch

| Ak váš systém stále zlyháva, musíte zistiť, ktoré údaje a aplikácie bežia na vašom primárnom systéme a vašom záložnom systéme. Kritické údaje sa snažite umiestniť na systém, ktorý je v prípade zlyhania schopný zvládnuť zaťaženie. Nechcete prečerpať diskový priestor. Ak sa na vašom primárnom systéme dôjde vyčerpať diskový priestor a dôjde k zlyhaniu, je vysoko pravdepodobné, že aj na vašom záložnom systéme dôjde k zlyhaniu z dôvodu nedostatku diskového priestoru. Aby nebolo vaše dátové centrum úplne zničené v prípade prírodnej pohromy, ako je potopa, tornádo alebo hurikán, mali by ste umiestniť replikovaný systém vo vzdialenej lokalite.

### **Plán pre prepínateľné nezávislé diskové oblasti a zrkadlenie lokalít (XSM):**

Jediná kópia údajov je udržiavaná na prepínateľnom hardvéri; tým je buď skladovacia jednotka (veža), alebo, v prostredí logických oddielov, IOP.

Keď dôjde k výpadku primárneho uzla, prístup k údajom na prepínateľnom hardvéri sa prepne na určený záložný uzol. Okrem toho môžete nezávislé diskové oblasti použiť v prostredí zrkadlenia lokalít (XSM). Toto umožňuje pre účely dostupnosti a ochrany uchovávať zrkadlovú kópiu nezávislej diskovej oblasti v systéme, ktorý je (voliteľne) geograficky vzdialený od pôvodnej lokality.

Ak chcete využívať prepínateľné prostriedky nachádzajúce sa v prepínateľných nezávislých diskových oblastiach alebo zrkadlenie lokalít (XSM), je potrebné vykonať dôkladné plánovanie.

#### **Súvisiace koncepty**

Plán pre nezávislé diskové oblasti

## **Bezpečnosť klastra**

Keď plánujete používanie klastrovanie vo vašich systémoch, musíte zväziť niektoré bezpečnostné aspekty.

### **Povolenie pridania uzla do klastra**

Skôr ako budete môcť pridať uzol do klastra, musíte nastaviť hodnotu pre sieťový atribút ALWADDCLU (Allow add to cluster).

V každom serveri, ktorý chcete nastaviť ako uzol klastra, použite príkaz CHGNETA (Change Network Attributes). Príkaz CHGNETA (Change Network Attributes) zmení sieťové atribúty systému. Sieťový atribút ALWADDCLU určuje, či uzol povolí inému systému, aby ho pridal ako uzol do klastra.

**Poznámka:** Na zmenu sieťového atribútu ALWADDCLU musíte mať oprávnenie \*IOSYSCFG.

Môžete si vybrať z týchto hodnôt:

**\*SAME**

Hodnota sa nezmení. Systém je distribuovaný s hodnotou \*NONE.

**\*NONE**

Žiaden iný systém nemôže pridať tento systém ako uzol do klastra.

**\*ANY** Akýkoľvek iný systém môže pridať tento systém ako uzol do klastra.

**\*RQSAUT**

Akýkoľvek iný systém môže pridať tento systém ako uzol do klastra, ale len ak boli požiadavka a klaster autentifikovaní.

Sieťový atribút ALWADDCLU sa skontroluje za účelom zistenia, či uzol, ktorým má byť pridaný do klastra, možno pridať alebo či treba vyhodnotiť požiadavky klastra pomocou digitálnych certifikátov X.509. **Digitálny certifikát** je formulár osobnej identifikácie, ktorý môže byť overený elektronicky. Ak je požadované overenie, požadujúci uzol a uzol, ktorý je pridávaný, musia mať nainštalované na systéme nasledujúce:

- Voľba 34 systému i5/OS (Správca digitálnych certifikátov)
- Cryptographic Access Provider

Keď je vybrané \*RQSAUT, musí byť správne nastavený zoznam dôveryhodných certifikačných autorít pre aplikáciu bezpečnostného servera klastra i5/OS. Identifikátor aplikácie servera je QIBM\_QCST\_CLUSTER\_SECURITY. Certifikačné oprávnenia pridajte aspoň pre tie uzly, ktoré povoľujete pripojiť ku klastru.

**Súvisiace koncepty**

Správa digitálnych certifikátov

“Bežné problémy s klastrom” na strane 132

Táto téma uvádza niektoré bežné problémy, ku ktorým môže dôjsť v klastri, ako aj spôsoby, ako sa im vyhnúť alebo vykonať obnovu.

**Súvisiaci odkaz**

Príkaz CHGNETA (Change Network Attributes)

## Distribúcia informácií v celom klastri

I Dozviete sa o aspektoch bezpečnosti pri používaní a manažovaní informácií pre celý klaster.

Pomocou API QcstDistributeInformation (Distribute Information) môžete odosielať správy z jedného uzla v doméne obnovy skupiny prostriedkov klastra do ostatných uzlov danej domény obnovy. To môže byť užitočné pri spracovaní ukončovacieho programu. Avšak treba poznamenať, že tieto informácie nie sú šifrované. Bezpečné informácie by nemali byť posielané pomocou tohto mechanizmu pokiaľ používate bezpečnú sieť.

Pomocou rozhraní API pre klastrované hašovacie tabuľky je možné zdieľať a replikovať neperzistentné údaje medzi uzlami klastra. Údaje sú uložené v neperzistentnom úložnom priestore. To znamená, že sú tieto údaje podržané len pokiaľ je uzol klastra súčasťou klastrovej hašovacej tabuľky. Tieto rozhrania API môžu byť použité len na klastrovom uzle, ktorý je určený v doméne klastrovej hašovacej tabuľky. Tento klastrový uzol musí byť aktívny v rámci klastra.

Ostatné informácie, distribuované cez správy klastra nie sú takisto bezpečné. Toto zahŕňa spodnú úroveň posielania správ klastra. Takže pri vykonaní zmien nie sú v údajoch ukončovacieho programu šifrované správy, obsahujúce tieto údaje bezpečné.

**Súvisiaci odkaz**

API QcstDistributeInformation (Distribute Information)

## Udržiavanie užívateľských profilov vo všetkých uzloch

- | Na údržbu užívateľských profilov vo všetkých uzloch klastra môžete použiť dva mechanizmy.
- | Jedným mechanizmom je vytvorenie administratívnej domény klastra na monitorovanie zdieľaných prostriedkov v uzloch klastra. Administratívna doména klastra môže okrem užívateľských profilov monitorovať rôzne iné typy prostriedkov, čo zabezpečuje jednoduché manažovanie prostriedkov, ktoré sa zdieľajú medzi uzlami. Detaily o týchto prostriedkoch nájdete v téme Monitorované prostriedky. Ak je administratívna doména klastra aktívna a užívateľské profily sa zaktualizujú, zmeny sa automaticky rozšíria do ostatných uzlov. Ak administratívna doména nie je aktívna, zmeny sa rozšíria po jej aktivovaní.
- | **Poznámka:** Ak plánujete zdieľať užívateľské profily používajúce synchronizáciu hesiel v klastri, musíte nastaviť systémovú hodnotu Zachovať bezpečnosť servera (QRETSVRSEC) na 1.
- | Pri použití druhého mechanizmu môžu administrátori pomocou Riadiacej centrály v Navigátore iSeries vykonávať tiež funkcie pre viacero systémov a skupín systémov. Táto podpora obsahuje niektoré spoločné úlohy správy užívateľov, ktoré musia operátori vykonať na viacerých systémoch v ich klastroch. Riadiaca centrála umožňuje, aby boli funkcie užívateľského profilu vykonané proti skupine systémov. Správca môže špecifikovať príkaz, ktorý sa vykoná na cieľových systémoch, keď vytvárate užívateľský profil.

### Súvisiace koncepty

“Úloha štruktúry a užívateľské fronty” na strane 112

Pri manažovaní klastra musíte poznať štruktúry úloh a užívateľské fronty.

“Administratívna doména klastra” na strane 8

*Administratívna doména klastra* sa používa na manažovanie prostriedkov, ktoré je nutné v uzloch prostredia s klastrami udržiavať konzistentné.

## Aspekty pre používanie klastrov s firewallmi

- | Ak používate klastrovanie v sieti s firewallmi, mali by ste poznať niektoré obmedzenia a požiadavky.
- | Ak používate klastrovanie s firewallom, musíte každému uzlu povoliť odosielanie odchádzajúcich správ a príjem prichádzajúcich správ od ostatných uzlov klastra. Pre každú adresu klastra v každom uzle musí existovať povolenie vo firewalli, aby sa umožnila komunikácia so každou adresou klastra vo všetkých ostatných uzloch. Pakety IP prenášané sieťou môžu predstavovať rôzne typy premávky. Klastrovanie používa funkciu ping (typ ICMP) a tiež UDP a TCP. Užívateľ môže nakonfigurovať firewall na filtrovanie premávky podľa typu. Aby klastrovanie fungovalo, firewall musí povoliť premávku protokolu ICMP, UDP a TCP. Odchádzajúca premávka sa môže odosielať na ľubovoľný port a prichádzajúca premávka sa prijíma na porte 5550 a 5551.

## Zoznam klastrovej konfigurácie

Splňte kontrolný zoznam pre konfiguráciu klastra, aby ste zabezpečili, že pred začatím konfigurovania klastra bude vaše prostredie správne pripravené.

Tabuľka 13. Kontrolný zoznam pre konfiguráciu TCP/IP v klastroch

Požiadavky pre TCP/IP	
___	Spustíte TCP/IP v každom uzle, ktorý chcete zahrnúť do klastra pomocou príkazu STRTCP (Start TCP/IP).
___	Nakonfigurujete adresu spätnej slučky TCP (127.0.0.1) a skontrolujete, že jej stav je <i>Aktívna</i> . Pomocou príkazu WRKTCPSTS (Work with TCP/IP Network Status) to skontrolujete v každom uzle klastra.
___	Pomocou príkazu WRKTCPSTS (Work with TCP/IP Network Status) v uzle skontrolujete, že adresy IP, používané pre klastrovanie pre daný uzol, zobrazujú hodnotu <i>Aktívna</i> .

Tabuľka 13. Kontrolný zoznam pre konfiguráciu TCP/IP v klastroch (pokračovanie)

Požiadavky pre TCP/IP	
—	<p>Skontrolujte, že je vo všetkých uzloch klastra aktívne INETD. Môžete to vykonať pomocou príkazu STRTCPSVR *INETD alebo v Navigátore iSeries podľa týchto krokov:</p> <ol style="list-style-type: none"> <li>1. V Navigátore iSeries rozviňte <b>Sieť</b>.</li> <li>2. Rozviňte <b>Servery</b>.</li> <li>3. Rozviňte <b>TCP/IP</b>.</li> <li>4. Kliknite pravým tlačidlom myši na <b>INETD</b> a označte <b>Štart</b>.</li> </ol> <p>Toto je možné overiť prítomnosťou úlohy Užívateľ QTCP (QTOGINTD) v zozname aktívnych úloh na uzle subjektu.</p>
—	<p>Skontrolujte, že užívateľský profil pre INETD, určený v /QIBM/ProdData/OS400/INETD/inetd.config, nemá väčšie než minimálne oprávnenie. Ak má tento užívateľský profil väčšie než minimálne oprávnenie, príkaz STRCLUNOD (Start Cluster Node) zlyhá. Štandardne je pre INETD zadaný užívateľský profil QUSER.</p>
—	<p>Skontrolujte, že každá adresa IP v klastri sa môže smerovať a odosielať datagramy UDP všetkým ostatným adresám IP v klastri. Použite príkaz PING, pričom zadáte lokálnu adresu IP a príkaz TRACEROUTE, kde zadáte správy UDP.</p>
—	<p>Skontrolujte, že porty 5550 a 5551 nepoužívajú iné aplikácie. Tieto porty sú rezervované pre klastrovanie IBM. Využitie portov môžete zobrazit pomocou príkazu WRKTCPSTS (Work with TCP/IP Network Status). Hneď, ako sa spustí INETD, bude port 5550 klastrovaním otvorený a v stave 'Načúvať'.</p>

Ak plánujete vo vašom klastri používať prepínateľné zariadenia, musíte splniť tieto požiadavky:

Tabuľka 14. Kontrolný zoznam pre konfiguráciu obnoviteľných zariadení v klastroch

Požiadavky pre obnoviteľné zariadenia	
—	<p>Skontrolujte, že je vo všetkých uzloch klastra, ktoré budú v doméne zariadení, nainštalovaná voľba 41 (Vysokodostupné prepínateľné zariadenia) s platným licenčným kľúčom. Každé použitie rozhrania na manažovanie klastrov v Navigátore iSeries vyžaduje túto voľbu.</p>
—	<p>Ak chcete používať funkcie Navigátora iSeries na manažovanie diskov, nakonfigurujte v serveri servisných nástrojov (STS) prístup k DST a užívateľské profily. Bližšie informácie nájdete na adrese komunikácia nastavenia.</p>
—	<p>Ak prepínate pružné zariadenia medzi logickými oddielmi na systéme a používate na riadenie vašich logických oddielov niečo iné ako HMC, povoľte pre oddiely Virtual OptiConnect. Toto bude vykonané pri prihlásení vyhradených servisných nástrojov DST (dedicated service tools). Bližšie informácie nájdete na adrese Virtual OptiConnect.</p> <p>Ak na manažovanie vašich oddielov používate hardvérovú riadiacu konzolu, zmeňte vlastnosti vášho profilu oddielu na záložke OptiConnect, aby ste povolili Virtuálne pripojenie OptiConnect pre každý oddiel v prepínateľnej konfigurácii. Aby sa zmena prejavila, musíte aktivovať profil oddielu.</p>
—	<p>Ak sa veža v slučke HSL OptiConnect prepína medzi dvoma systémami a jeden zo systémov má logické oddiely, povoľte pre oddiely HSL OptiConnect. Ak na manažovanie logických oddielov používate iný nástroj ako HMC, vykoná sa to pri prihlásení k vyhradeným servisným nástrojom (DST).</p> <p>Ak na manažovanie vašich oddielov používate hardvérovú riadiacu konzolu, zmeňte vlastnosti vášho profilu oddielu na záložke OptiConnect, aby ste povolili HSL OptiConnect pre každý oddiel v prepínateľnej konfigurácii. Aby sa zmena prejavila, musíte aktivovať profil oddielu.</p>
—	<p>Ak prepínate obnoviteľné zariadenia medzi logickými oddielmi a na manažovanie logických oddielov používate iný prostriedok ako konzolu HMC, musíte nakonfigurovať zbernicu, aby bola zdieľaná medzi oddielmi alebo nakonfigurovať I/O oblasť. Zbernica musí byť v jednom oddiele nakonfigurovaná ako "vlastná zdieľaná zbernica" a všetky ostatné oddiely, ktoré sa budú zúčastňovať prepínania zariadenia, musia byť nakonfigurované s hodnotou "použiť zdieľanú zbernicu".</p> <p>Ak na manažovanie logických oddielov používate hardvérovú riadiacu konzolu, musíte nakonfigurovať I/O oblasť, ktorá zahŕňa I/O procesor, I/O adaptér a všetky pripojené prostriedky, aby sa mohla nezávislá disková oblasť prepínať medzi oddielmi. Každý oddiel musí mať prístup k I/O oblasti. Viac detailov nájdete v časti Zabezpečenie prepínateľnosti vášho hardvéru. Detaily o požiadavkách pri fyzickom plánovaní pre prepínateľné prostriedky nájdete v časti Požiadavky pri fyzickom plánovaní.</p>
—	<p>Pri prepínaní veže na slučke HSL medzi dvoma rôznymi systémami musí byť veža nakonfigurovaná ako prepínateľná. Ďalšie informácie nájdete na adrese Urob svoj hardvér prepínateľným.</p>

Tabuľka 14. Kontrolný zoznam pre konfiguráciu obnoviteľných zariadení v klastroch (pokračovanie)

Požiadavky pre obnoviteľné zariadenia	
—	Keď sa do existujúcej slučky HSL pridá veža, reštartujte všetky servery v danej slučke.
—	Jednotka maximálneho prenosu MTU (maximum transmission unit) pre vaše komunikačné trasy musí byť väčšia ako laditeľný parameter komunikácie klastra, veľkosť fragmentu správy. MTU pre adresu IP klastra môžete v danom uzle skontrolovať pomocou príkazu WRKTCPS (Work with TCP/IP Network Status). MTU musí byť takisto overená na každom kroku po celej ceste komunikácie. Môže byť jednoduchšie zmenšiť parameter veľkosti fragmentu správy, keď je klaster už vytvorený, ako zvýšiť MTU pre komunikačné trasy. Viac informácií o veľkosti fragmentov správ nájdete v časti Nastaviteľné parametre pre komunikáciu v klastri. Na zobrazenie aktuálnych nastavení ladiacich parametrov môžete použiť API QcstRetrieveCRSInfo (Retrieve Cluster Resource Services Information) a na ich zmenu môžete použiť API QcstChgClusterResourceServices (Change Cluster Resource Services).

Tabuľka 15. Kontrolný zoznam pre konfiguráciu bezpečnosti v klastroch

Požiadavky pre bezpečnosť	
—	Ak sa pokúšate naštartovať vzdialený uzol, musia byť na cieľovom uzle vhodne nastavené sieťové atribúty Povolíť pridanie ku klastru ALWADDCLU (Allow Add to Cluster). Tieto by mali byť nastavené na *ANY alebo *RQSAUT v závislosti od vášho prostredia. Ak je nastavená hodnota *RQSAUT, musí byť nainštalovaná voľba 34 systému i5/OS (Správca digitálnych certifikátov) a produkt Cryptographic Access Provider. Viac informácií o nastavení sieťových atribútov ALWADDCLU nájdete v téme Povolíť pridanie uzla do klastra.
—	Povoľte stav užívateľského profilu pre INETD, zadaného v /QIBM/ProdData/OS400/INETD/inetd.config. Nemusíte mať špeciálne oprávnenie *SECADM ani *ALLOBJ.Štandardne je pre INETD zadaný užívateľský profil QUSER.
—	Skontrolujte, že užívateľský profil volajúci rozhrania API služieb klastrových prostriedkov existuje vo všetkých uzloch a má oprávnenie *IOSYSCFG.
—	Skontrolujte, že vo všetkých uzloch domény obnovy existuje užívateľský profil, ktorý má vykonať ukončovací program pre skupinu prostriedkov klastra (CRG).

Tabuľka 16. Kontrolný zoznam pre konfiguráciu úloh v klastroch

Aspekty pre úlohy	
—	Úlohy môžu byť odoslané službami prostriedkov klastra API požiadavkám procesu. Budú spustené buď pod užívateľským profilom pre spustenie ukončovacieho programu, zadaného pri vytváraní skupiny prostriedkov klastra, alebo pod užívateľským profilom, ktorý vyžadoval API (iba pre zmenu na zariadeniach v pružnom zariadení CRG). Užívateľ musí skontrolovať, či subsystém, ktorý spravuje front úlohy, priradený k užívateľskému profilu, je konfigurovaný ako: *NOMAX pre počet úloh, ktoré môže spustiť z toho frontu úloh.
—	Úlohy budú odoslané do frontu úloh, zadaných popisom úlohy, ktoré sú získané z užívateľského profilu definovaného pre CRG. Predvolený popis úlohy spôsobí, že úlohy budú odoslané do frontu úloh QBATCH. Keďže tento front úloh sa používa pre pre viacero užívateľských úloh, nemusí byť úloha ukončovacieho programu spúšťaná v časovom súlade. Užívatelia by mali zohľadniť popis jedinečnej úlohy s jedinečným užívateľským frontom.
—	Keď sú spustené úlohy ukončovacieho programu, použijú smerovacie dáta z popisu úlohy, aby vybrali, ktoré hlavné pamäťové oblasti a atribúty spúšťačieho času použijú. Predvolené hodnoty budú mať za následok úlohy, ktoré sa spúšťajú v oblasti s ďalšími dávkovými úlohami s prioritou vykonávania 50. Žiadne z nich nemôžu vytvoriť požadovaný výkon pre úlohy ukončovacieho programu. Subsystém, ktorý vyvoláva úlohy ukončovacieho programu (ten istý subsystém, ktorý používa jedinečný front úloh), by mal priradiť úlohy ukončovacieho programu k oblasti, ktorú nepoužívajú iné úlohy, vyvolávané tým istým subsystémom alebo inými subsystémami. Navyše, úlohy ukončovacieho programu by mali byť označené prioritou vykonávania 15, aby boli spustené pred takmer všetkými ostatnými užívateľskými úlohami.
—	Systémová hodnota QMLTTHDACN musí byť nastavená na 1 alebo 2.

K dispozícii je niekoľko riešení softvéru pre konfiguráciu a riadenie vášho klastra. Jedným z týchto riešení je manažovanie klastrov v Navigátore iSeries. Ak zvolíte používanie Navigátora iSeries, musíte splniť tieto požiadavky:



Tabuľka 17. Kontrolný zoznam pre Navigátor iSeries pre klastre

Aspekty pre manažovanie klastrov v Navigátore iSeries	
—	Vo všetkých uzloch klastra, ktoré budú v doméne zariadení, musí byť nainštalovaná voľba 41 (i5/OS - Vysokodostupné prepínateľné zariadenia) s platným licenčným kľúčom.
—	Pomocou príkazu STRHOSTSVR (Start Host Server) skontrolujte, že sú spustené všetky hostiteľské servery: STRHOSTSVR SERVER(*ALL)
—	Pomocou príkazu STRTCPSVR (Start TCP/IP Server) skontrolujte, že je spustený server Riadiacej centrály: STRTCPSVR SERVER(*MGTC)

### Súvisiace koncepty

“Manažovanie klastrov v Navigátore iSeries” na strane 72

IBM ponúka rozhranie na manažovanie klastrov, ktoré je dostupné v Navigátore iSeries a ktoré poskytuje voľba 41 (i5/OS - Vysokodostupné prepínateľné prostriedky).

“Server INETD”

Aby mohol byť uzol pridaný, alebo spustený, ako aj pri procese spájania oddielov, musí byť spustený Server internetového démona (INETD).

“Laditeľné parametre komunikácií klastra” na strane 94

API QcstChgClusterResourceServices (Change Cluster Resource Services) umožňuje vyladenie niektorých služieb topológie klastra a konfiguračných a výkonových parametrov pre komunikáciu v klastrí, aby lepšie zodpovedali rôznym jedinečným aplikačným a sieťovým prostrediam, kde sa používa klastrovanie. Takéto API je k dispozícii akémukoľvek klastru, ktorý je spustený na klastrovej verzii 2 alebo novej.

### Súvisiaci odkaz

“Kontrolný zoznam pre administratívnu doménu klastra” na strane 96

Uvádza všetky požiadavky, ktoré je nutné splniť pred vytvorením administratívnej domény klastra.

## Server INETD

Aby mohol byť uzol pridaný, alebo spustený, ako aj pri procese spájania oddielov, musí byť spustený Server internetového démona (INETD).

Odporúčame, aby bol server INETD vo vašom klastrí spustený vždy.

## Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (i5/OS - Vysokodostupné prepínateľné prostriedky).

Pri spúšťaní servera INETD dodržte tento postup:

1. V Navigátore iSeries rozviňte **Sieť**.
2. Rozviňte **Servery**.
3. Rozviňte **TCP/IP**.
4. Kliknite pravým tlačidlom myši na **INETD** a označte **Štart**.

## Použitie príkazov CL a rozhraní API

Server INETD môžete spustiť aj pomocou príkazu STRTCPSVR (Start TCP/IP Server) so zadaným parametrom \*INETD. Keď je server INETD spustený, objaví sa v Zozname aktívnych úloh toho ktorého uzla úloha QTOGINTD (User QTCP).

### Súvisiace koncepty

“Bežné problémy s klastrom” na strane 132

Táto téma uvádza niektoré bežné problémy, ku ktorým môže dôjsť v klastrí, ako aj spôsoby, ako sa im vyhnúť alebo vykonať obnovu.

### Súvisiaci odkaz

## Laditeľné parametre komunikácií klastra

API QcstChgClusterResourceServices (Change Cluster Resource Services) umožňuje vyladenie niektorých služieb topológie klastra a konfiguračných a výkonových parametrov pre komunikáciu v klastrí, aby lepšie zodpovedali rôznym jedinečným aplikačným a sieťovým prostrediam, kde sa používa klastrovanie. Takéto API je k dispozícii akémukoľvek klastru, ktorý je spustený na klastrovej verzii 2 alebo novšej.

Príkaz CHGCLUCFG (Change Cluster Configuration Tuning) poskytuje základnú úroveň ladenia, zatiaľ čo API QcstChgClusterResourceServices zabezpečuje základnú aj rozšírenú úroveň ladenia.

Pomocou API QcstChgClusterResourceServices a príkazu CHGCLUCFG môžete vyladiť výkon a konfiguráciu klastra. API a daný príkaz zabezpečujú základnú úroveň podpory ladenia tam, kde sa klaster nastavi na preddefinovanú množinu hodnôt, identifikovaných pre vysoký, nízky a normálny interval vyhradeného času a hodnoty intervalu posielania správ. Ak potrebujete vykonať ladenie na rozšírenej úrovni, obvyčajne za pomoci personálu podpory IBM, jednotlivé parametre môžete pomocou API vyladiť v rámci preddefinovaného rozsahu hodnôt. Nesprávne zmeny jednotlivých parametrov môžu mať ľahko za následok zníženie výkonu klastra.

### Kedy a ako vyladiť parametre klastra?

Príkaz CHGCLUCFG a API QcstChgClusterResourceServices poskytuje rýchly spôsob nastavenia výkonových a konfiguračných parametrov klastra bez toho, aby ste museli poznať detaily. Táto úroveň číselnej sústavy ladenia hlavne ovplyvňuje pulzovú citlivosť a hodnoty času, vyhradeného na odosielanie správ. Platné hodnoty pre úroveň číselnej sústavy podpory ladenia sú:

1 Vysoké hodnoty uplynutia vyhradeného času/Málo frekvenčný pulz

#### 2 Predvolené hodnoty

Normálne predvolené hodnoty sa používajú pre výkon klastrových komunikácií a konfiguračné parametre. Toto nastavenie sa môže použiť aj pre vrátenie všetkých parametrov na pôvodné predvolené hodnoty.

#### 3 Nízke hodnoty uplynutia vyhradeného času/Viac frekvenčný pulz

Na zníženie intervalu pulzu a hodnôt času, vyhradeného na odosielanie správ sú ku klastrovým komunikáciám pridávané prispôsobenia. S viac frekvenčnými pulzmi a kratšími časmi na odosielanie správ bude klaster rýchlejšie (viac citlivo) reagovať na komunikačné zlyhania.

V nasledujúcej tabuľke sú zobrazené príklady výsledných časov odozvy pre zlyhanie pulzu, vedúceho k rozdeleniu uzla:

	1 (Menej citlivé)			2 (Predvolené)			3 (Citlivejšie)		
	Detekcia problému s kontrolným signálom	Analýza	Spolu	Detekcia problému s kontrolným signálom	Analýza	Spolu	Detekcia problému s kontrolným signálom	Analýza	Spolu
Jednoduchá podsieť	00:24	01:02	01:26	00:12	00:30	00:42	00:04	00:14	00:18
Viacnásobná podsieť	00:24	08:30	08:54	00:12	04:14	04:26	00:04	02:02	02:06

! **Poznámka:** Časy sú uvedené vo formáte minúty:sekundy.

V závislosti od bežného zaťaženia siete a špecifického fyzického média môže klastrový administrátor vybrať prispôsobenie citlivosti pulzu a úrovni vyhradeného času pre posielanie správ. Napríklad pri použití vysokorýchlostného spoľahlivého transportu, napríklad OptiConnect, ak sú všetky systémy klastrí na spoločnej zbernici OptiConnect, môžete chcieť vytvoriť citlivejšie prostredie, aby ste zabezpečili rýchlu detekciu za účelom rýchlejšieho



núdzového prepnutia. Vyberiete voľbu 3. Ak by niekto spúšťal program na ťažko zaťaženej 10-megabitovej ethernetovej zbernici a predvolené nastavenia by viedli k občasným rozdeleniam len kvôli špičkovým zaťaženiám siete, potom by bola vybraná voľba 1 na zníženie citlivosti klastrovania na špičkové zaťaženie.

Zmeniť služby klastrových prostriedkov API takisto umožňuje naladenie špecifických individuálnych parametrov tam, kde požiadavky prostredia siete vytvárajú jedinečné situácie. Uvažujte napríklad znova klastre so všetkými uzlami na spoločnej zbernici OptiConnect. Výkon správ klastra môžete výrazne zvýšiť, ak nastavíte parameter Veľkosť fragmentov správ na maximálnu hodnotu 32500 bajtov, aby lepšie zodpovedala veľkosti maximálnej prenosovej jednotky (MTU) zbernice OptiConnect, než predvolená hodnota 1464 bajtov. Toto znižuje režiu fragmentácie a opätovné zhromaždenie veľkých správ. Výsledok, samozrejme, závisí od klastrových aplikácií a využitia klastrového posielania správ, vyplývajúceho z týchto aplikácií. Ostatné parametre sú definované v dokumentácii k API a môžete ich použiť na vyladenie výkonu pri výmene správ v klastri alebo na zmenu citlivosti klastra na rozloženie.

### Súvisiace koncepty

“Výkonnosť optimalizovaného klastra” na strane 110

Pretože existujú potenciálne podstatné rozdiely vo vašom komunikačnom prostredí, máte možnosť prispôbiť premenné, ktoré sa týkajú klastrovej komunikácie, aby čo najlepšie vyhovovali vášmu prostrediu.

## Kontrolný zoznam pre zrušenie konfigurácie klastra

Ak potrebujete zrušiť klastre alebo CRG, musíte systematicky odstrániť rôzne komponenty klastra, aby ste zabezpečili úplné zrušenie konfigurácie.

Tabuľka 18. Kontrolný zoznam pre zrušenie konfigurácie nezávislej diskovej oblasti v klastroch

Požiadavky pre nezávislú diskovú oblasť	
—	Ak plánujete odstrániť podmnožinu skupiny nezávislých diskových oblastí alebo odstrániť poslednú nezávislú diskovú oblasť z prepínateľných zariadení, musíte najprv ukončiť CRG. Použite príkaz ENDCRG (End Cluster Resource Group).
—	<p>Ak chcete vymazať nezávislú diskovú oblasť, ktorá je súčasťou klastra, dôrazne sa odporúča najprv odstrániť objekt konfigurácie diskovej oblasti z prepínateľného zariadenia, známy tiež ako skupina prostriedkov klastra (CRG) zariadenia. Ak chcete vymazať objekt konfigurácie diskovej oblasti z prepínateľného zariadenia, vykonajte tieto kroky:</p> <p>Ak chcete vymazať diskovú oblasť z prepínateľného zariadenia, vykonajte tieto kroky:</p> <ol style="list-style-type: none"> <li>1. V Navigátore iSeries rozviňte položku <b>Riadiaca centrála</b> → <b>Klastre</b>.</li> <li>2. Rozviňte <i>názov klastra obsahujúceho prepínateľné zariadenie</i> → <b>Prepínateľné zariadenia</b>.</li> <li>3. Kliknite na názov prepínateľného zariadenia.</li> <li>4. V pravej časti okna Navigátora iSeries kliknite pravým tlačidlom myši na diskovú oblasť a vyberte voľbu <b>Odstrániť</b>.</li> </ol> <p>Na odstránenie objektu konfigurácie nezávislej diskovej oblasti z CRG môžete použiť tiež príkaz RMVCRGDEVE (Remove CRG Device Entry).</p>
—	Keď odstránite objekt konfigurácie nezávislej diskovej oblasti z prepínateľného zariadenia klastra, môžete vymazať nezávislú diskovú oblasť .
—	<p>Podľa týchto úloh vymažte opis zariadenia pre nezávislú oblasť:</p> <ol style="list-style-type: none"> <li>1. V rozhraní príkazového riadka zadajte WRKDEVD DEVD(*ASP) a stlačte kláves Enter.</li> <li>2. Pomocou klávesu Page down rolujte nadol, až kým sa nezobrazí opis zariadenia pre nezávislú diskovú oblasť, ktorú chcete vymazať.</li> <li>3. Vyberte voľbu 4 (Vymazať) vedľa názvu opisu zariadenia a stlačte kláves Enter.</li> </ol>

Tabuľka 19. Kontrolný zoznam pre zrušenie konfigurácie skupiny prostriedkov klastra v klastroch

Požiadavky pre skupinu prostriedkov klastra	
—	Ak chcete vymazať skupinu prostriedkov klastra, vykonajte jednu z týchto úloh: <ol style="list-style-type: none"> <li>1. Ak v uzle nie je aktívne klastrovanie, v rozhraní príkazového riadka zadajte DLTCRG CRG(NAZOVCRG). NAZOVCRG je názov CRG, ktoré chcete vymazať. Stlačte kláves Enter</li> <li>2. Ak je v uzle aktívne klastrovanie, v rozhraní príkazového riadka zadajte DLTCRGCLU CLUSTER(NAZOVKLAstra) CRG(NAZOVCRG). NAZOVKLAstra je názov klastra. NAZOVCRG je názov CRG, ktoré chcete vymazať. Stlačte kláves Enter</li> </ol>

## Plán pre administratívnu doménu klastra

Administratívna doména klastra vyžaduje plánovanie manažovania prostriedkov, ktoré sa zdieľajú medzi jej uzlami.

Keď vytvoríte administratívnu doménu klastra, automaticky sa vytvorí CRG s partnerskými uzlami, ktoré bude reprezentovať danú doménu. Administratívnu doménu klastra môžete manažovať pomocou rozhraní API, príkazov CL a pomocou Navigátora iSeries.

Administrátor klastra môže vytvoriť administratívnu doménu klastra a potom do nej pridať monitorované prostriedky, ktoré sa zdieľajú medzi uzlami. Klaster i5/OS poskytuje zoznam systémových prostriedkov, ktoré je možné zdieľať medzi uzlami administratívnej domény klastra, reprezentovaných pomocou *položiek monitorovaných prostriedkov (MRE)*. Celý zoznam systémových prostriedkov, ktoré môžete monitorovať, nájdete v téme Monitorované prostriedky.

Pri navrhovaní administratívnej domény klastra by ste si mali odpovedať na tieto otázky:

### Aké prostriedky sa budú zdieľať?

Budete musieť určiť, ktoré prostriedky bude potrebné zdieľať. Výberom atribútov každého z týchto prostriedkov môžete prispôsobiť, čo sa bude zdieľať medzi uzlami. Aplikácie, ktoré sa vykonávajú vo viacerých uzloch, môžu na správne fungovanie potrebovať špecifické premenné prostredia. Okrem toho, údaje, ktoré sú rozšírené v medzi viacero uzlov, môžu vyžadovať prístup k určitým užívateľským profilom. Skôr než určíte, ktoré prostriedky sa budú zdieľať, by ste mali poznať prevádzkové požiadavky vašich aplikácií a údajov.

### Ktoré uzly budú patriť do administratívnej domény klastra?

Mali by ste určiť, ktoré uzly klastra bude manažovať administratívna doména klastra. Uzly nemôžu byť vo viacerých administratívnych doménach klastra. Máte napríklad klaster so štyrmi uzlami (Uzol A, Uzol B, Uzol C a Uzol D). Uzly A a B môžu byť v jednej administratívnej doméne klastra a uzly B a C v inej. Avšak Uzol B a C nemôžete mať v ďalšej administratívnej doméne klastra.

### Aká bude konvencia pomenovania pre administratívne domény klastra?

V závislosti od komplexnosti a veľkosti vášho prostredia s klastrami môžete chcieť vytvoriť štandardnú konvenciu pomenovania pre skupiny CRG s partnerskými uzlami a pre administratívne domény klastra. Nakoľko sa na reprezentáciu administratívnej domény klastra vytvorí CRG s partnerskými uzlami, pravdepodobne ho budete chcieť odlišiť od tých, ktoré monitorujú prostriedky v klastri. Napríklad skupiny CRG s partnerskými uzlami, ktoré reprezentujú administratívne domény klastra, môžete pomenovať *ADMDMN1*, *ADMDMN2* a tak ďalej, zatiaľ čo ostatné skupiny CRG s partnerskými uzlami môžete pomenovať *PARTNER1*. Ak chcete zistiť, či sa CRG s partnerskými uzlami používa ako administratívna doména klastra, môžete použiť tiež API `QcstListClusterResourceGroupIn` (List Cluster Resource Group Information).

## Kontrolný zoznam pre administratívnu doménu klastra

Uvádza všetky požiadavky, ktoré je nutné splniť pred vytvorením administratívnej domény klastra.

Tabuľka 20. Kontrolný zoznam pre administratívnu doménu klastra

Požiadavky pre administratívnu doménu klastra	
—	Skontrolujte, že bol nakonfigurovaný klaster. Pozrite si Kontrolný zoznam pre konfiguráciu klastra.
—	Ak plánujete monitorovať užívateľské profily používajúce synchronizáciu hesiel v klastri, musíte nastaviť systémovú hodnotu Zachovať bezpečnosť servera (QRETSVRSEC) na 1.

Tabuľka 20. Kontrolný zoznam pre administratívnu doménu klastra (pokračovanie)

Požiadavky pre administratívnu doménu klastra	
—	Ak chcete pridať prostriedky do administratívnej domény klastra, všetky uzly v administratívnej doméne klastra musia byť aktívne, byť súčasťou skupiny a nemôžu mať oddiely.

## Konfigurácia klastrov

Pochopte postup pri vytváraní klastra.

Spoločnosť IBM a obchodní partneri IBM pre softvér strednej úrovne pre klastre vytvorili spoločný tím za účelom poskytnúť vyspelé funkcie služieb klastrových prostriedkov spolu s grafickým užívateľským prostredím (GUI) na manažovanie klastrov. Služby klastrových prostriedkov i5/OS poskytujú množinu integrovaných služieb, ktoré zabezpečujú údržbu topológie klastra, monitorujú kontrolné signály a umožňujú vytvárať a spravovať konfiguráciu klastra a skupiny prostriedkov klastra. Služby klastrových prostriedkov taktiež poskytujú funkcie spoľahlivého zasielania správ, ktoré sledujú každý uzol v klastri a zabezpečujú, aby mal každý uzol nepretržité informácie o stave klastrových prostriedkov. Okrem toho poskytujú služby klastrových prostriedkov množinu príkazov riadiaceho jazyka (CL), aplikačných programových rozhraní (API) a funkcií, pomocou ktorých môžu poskytovatelia alebo užívatelia aplikácií iSeries zvýšiť ich dostupnosť. Služby klastrových prostriedkov sú dostupné tiež cez riešenia s grafickým užívateľským rozhraním, ktoré poskytuje manažovanie klastrov v Navigátore iSeries a produkty softvéru strednej úrovne pre klastre od obchodných partnerov IBM.

### Začínate

**Ak chcete nakonfigurovať klaster, vykonajte tieto kroky:**

**1. Výber softvérového riešenia.**

Celkový pohľad na voľby pre konfigurovanie a manažovanie klastrov nájdete v téme “Riešenia konfigurácie a správy klastrov” na strane 72.

**2. Splňte požiadavky na hardvér, softvér a komunikáciu.**

Prezrite si požiadavky klastrovania v časti Plánovanie klastra.

**3. Nastavenie sieťového a serverového prostredia pre klastre.**

Použite “Zoznam klastrovej konfigurácie” na strane 90 a uistite sa, že ste pripravený na konfiguráciu klastrov vo vašom prostredí.

**4. Konfigurácia klastra.**

**Súvisiace koncepty**

“Kam volať pre podporu klastrov” na strane 147

Túto tému si pozrite, ak potrebujete kontaktovať IBM ohľadom otázok týkajúcich sa klastrov.

## Vytvorenie klastra

Ak chcete vytvoriť a nakonfigurovať klaster, musíte do klastra zahrnúť aspoň jeden uzol a musíte mať prístup aspoň k jednému z uzlov, ktoré budú v klastri.

Ak je zadaný iba jeden uzol v klastri, tak to musí byť server, ku ktorému práve prístupujete. Celý zoznam požiadaviek na vytvorenie klastra nájdete v téme “Zoznam klastrovej konfigurácie” na strane 90.

Ak budete vo vašom klastri používať prepínateľné zariadenia, je potrebné zohľadniť také prípady, pri ktorých sa takéto zariadenia nepoužívajú. Pri nastavení prostredia klastra, v ktorom sú obsiahnuté prepínateľné zariadenia, musíte zabezpečiť, aby ste sa vyhli konfliktom v klastri. Pozrite si tému o podrobných inštrukciách konfigurovania klastra s použitím prepínateľných zariadení Vytvorenie prepínateľných diskových oblastí.

## Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Manažovanie klastrov v Navigátore iSeries zahŕňa sprievodcu, ktorý vás prevedie krokmi vytvorenia a spustenia jednoduchého klastra, pozostávajúceho z jedného alebo dvoch uzlov. Najskôr vytvoríte jednouzlový alebo dvojzlový klaster, potom môžete do neho pridať uzol. Klaster vytvorený v Navigátore iSeries môže obsahovať maximálne štyri uzly. Tento sprievodca vás prevedie krokmi na zadanie serverov na zahrnutie a vytvorenie skupín prostriedkov klastra. Keď vytvoríte jednoduchý klaster, server, na ktorom vytvoríte klaster, musí byť jedným z uzlov.

Ak chcete pomocou Sprievodcu Nový klaster v Navigátore iSeries vytvoriť jednoduchý klaster, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Pravým tlačidlom myši kliknite na **Klaster** a vyberte voľbu **Nový klaster**.
3. Postupujete podľa sprievodcu, a tak vytvoríte nový klaster.

Keď vytvoríte klaster, skontrolujte:

1. Že ste pridali všetky uzly, ktoré chcete zahrnúť do klastra. Do klastra, ktorý vytvoríte a manažujete pomocou Navigátora iSeries, môžete pridať maximálne štyri uzly.
2. Či ste pridali všetky uzly do domény obnovy (na použitie s prepínateľnými hardvérovými skupinami a nezávislými diskovými oblasťami).
3. Že ste vytvorili a spustili prepínateľné prostriedky (prepínateľné zariadenie, prepínateľná aplikácia a prepínateľné údaje).

Online pomoc v Navigátore iSeries obsahuje podrobný opis procedúr na vykonanie týchto úloh.

## Použitie príkazov CL a rozhraní API

Na vytvorenie klastra tiež môžete použiť príkazy CL alebo rozhrania API:

### 1. Vytvorenie klastra.

Príkaz CRTCLU (Create Cluster)

API QcstCreateCluster (Create Cluste)

### 2. Pridanie uzlov do vášho klastra z aktívneho klastrového uzla.

Príkaz ADDCLUNODE (Add Cluster Node Entry)

API QcstAddClusterNodeEntry (Add Cluster Node Entry)

### 3. Spustenie uzla klastra.

Príkaz STRCLUNOD (Start Cluster Node)

API QcstStartClusterNode (Start Cluster Node)

### 4. Definovanie domén zariadení. Ak plánujete používať prepínateľné zariadenia, musíte v doméne zariadenia zahrnúť požadované uzly.

Príkaz ADDDEVDMNE (Add Device Domain Entry)

API QcstAddDeviceDomainEntry (Add Device Domain Entry)

### 5. Vytvorenie skupín prostriedkov klastra (CRG).

Príkaz CRTCRG (Create Cluster Resource Group)

API QcstCreateClusterResourceGroup (Create Cluster Resource Group)

### 6. Spustenie skupín klastrových prostriedkov (CRG).

Príkaz STRCRG (Start Cluster Resource Group)

API QcstStartClusterResourceGroup (Start Cluster Resource Group)

---

## Riadenie klastrov

V tejto téme nájdete informácie o niektorých úlohách, ktoré sa týkajú správy vašich klastrov.

Ak ste neuvažovali o type rozhrania, ktoré budete používať na manažovanie vašich klastrov, najprv si pozrite časť Riešenia pre manažovanie klastrov.

Toto sú niektoré zmeny, ktoré môžete vykonať na klastri po jeho konfigurácii:

## Úlohy pre klastre

- Pridanie uzla do klastra
- Odstránenie uzla z klastra
- Spustenie klastrového uzla
- Ukončenie klastrového uzla
- Nastavenie verzie klastra na najnovšiu úroveň
- Vymazanie klastra
- Zmena uzla klastra

## Úlohy pre skupinu prostriedkov klastra

- Vytvorenie nových skupín klastrových prostriedkov
- Vymazanie jestvujúcich skupín klastrových prostriedkov
- Spustenie skupiny klastrových prostriedkov
- Pridanie uzla do skupiny prostriedkov klastra
- Odstránenie uzla zo skupiny prostriedkov klastra
- Ukončenie skupiny klastrových prostriedkov
- Zmena domény obnovy pre skupinu klastrových prostriedkov
- Vykonanie prepnutia
- Pridanie uzla do domény pre zariadenie
- Odstránenie uzla z domény pre zariadenie

Táto téma vám tiež pomôže pri ukladaní konfigurácií klastra. Možno sa budete chcieť dozvedieť o štruktúre úloh služieb klastrových prostriedkov a o tom, ako rozhrania API pre klastre používajú užívateľské fronty. Dozviete sa, ako správne ukončovať úlohy klastra a ako monitorovať stav klastra. Môžete sa tiež dozvedieť, ako vás funkcia garantovaných správ a monitorovanie kontrolného signálu informuje o aktuálnom stave vášho klastra.

## Úlohy pre administratívnu doménu klastra

- Vytvorenie administratívnej domény klastra
- Pridanie monitorovaných prostriedkov
- Vymazanie administratívnej domény klastra

### Súvisiace koncepty

“Funkcia spoľahlivých správ” na strane 26

*Funkcia spoľahlivých odkazov* služby klastrových prostriedkov sleduje každý uzol v klastri a zabezpečuje, že všetky uzly majú nepretržité informácie o stave klastrových prostriedkov.

“Sledovanie aktivity” na strane 25

*Monitorovanie kontrolného signálu* je funkcia služieb klastrových prostriedkov, ktorá zabezpečuje, že každý uzol je aktívny. Funkcia odosiela z každého uzla v klastri signál do všetkých ostatných uzlov v klastri na potvrdenie, že sú uzly stále aktívne.

## Pridanie uzla do klastra

Uzol môžete do klastra pridať pomocou Navigátora iSeries alebo príkazov.

### Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Jednoduchý klaster, ktorý podporuje Navigátor iSeries môže pozostávať maximálne zo štyroch uzlov. Ak už v klastri existujú štyri uzly, voľba **Pridať uzol** je deaktivovaná. Ak potrebujete mať v klastri viac než štyri uzly, mali by ste použiť príkazy a rozhrania API pre klastre alebo produkt softvéru strednej úrovne pre klastre od obchodného partnera IBM, ktorý podporuje až 128 uzlov.

Pri pridaní uzlu do klastra vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Rozviňte klaster, do ktorého chcete pridať uzol.
4. Kliknite pravým tlačidlom myši na **Uzly** a vyberte **Pridať uzol...**

### Použitie príkazov a rozhraní API pre klastre

Na pridanie uzlu do klastra môžete ešte použiť:

- Príkaz ADDCLUNODE (Add Cluster Node Entry)
- API QcstAddClusterNodeEntry (Add Cluster Node Entry)

#### Súvisiace koncepty

“Klastrové príkazy a API” na strane 73

Služby klastrových prostriedkov i5/OS poskytujú množinu príkazov riadiaceho jazyka (CL), aplikačných programových rozhraní (API) a funkcií, pomocou ktorých môžu poskytovatelia alebo užívatelia aplikácií iSeries zvýšiť ich dostupnosť.

“Obchodní partneri IBM poskytujúci softvér strednej úrovne pre klastre a dostupné produkty pre klastrovanie” na strane 79

Od obchodných partnerov IBM poskytujúcich softvér strednej úrovne pre klastre môžete získať produkt zabezpečujúci funkcie logickej replikácie, ktoré sú integrálnou súčasťou klastrovania a zjednodušujú vytváranie a manažovanie klastrov.

## Spustenie klastrového uzla

Spustenie klastrového uzla spustí na tomto uzle služby klastrových prostriedkov. Od verzie 3 sa uzol môže spustiť sám a bude schopný opätovne sa pripojiť k aktívnemu klastru, ak v ňom nájde aktívny uzol.

### Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Keď sú služby klastrových prostriedkov na určenom uzle úspešne spustené, nastaví sa stav uzla na *Spustený*.

Pri spustení klastrovania na uzle dodržte tento postup:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Rozviňte klaster obsahujúci uzol, v ktorom chcete spustiť klastrovanie.
4. Kliknite na **Uzly**.
5. Pravým tlačidlom myši kliknite na uzol, v ktorom chcete spustiť klastrovanie a vyberte voľbu **Klastre** → **Spustiť**.

Klaster

### Použitie príkazov CL a rozhraní API

Na spustenie uzla môžete použiť aj príkazy CL a API. Keď sú služby klastrových prostriedkov na určenom uzle úspešne spustené, nastaví sa stav uzla na *Aktívny*.

- Príkaz STRCLUNOD (Start Cluster Node)
- API QcstStartClusterNode (Start Cluster Node)

#### Súvisiace úlohy

“Ukončenie klastrových úloh” na strane 111

Úlohu klastra sa nikdy sa nepokúšajte ukončiť priamo.

“Obnova po zlyhaní úloh klastra” na strane 138  
Zlyhanie úlohy služby klastrových prostriedkov zvyčajne naznačuje iný problém.

## | Ukončenie klastrového uzla

| Zastavenie, alebo ukončenie uzla, zastavuje na uzle služby klastrových prostriedkov.

### | Použitie Navigátora iSeries

| Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

| Keď sú služby klastrových prostriedkov na určenom uzle úspešne zastavené, nastaví sa stav uzla na *Zastavený*.

| Pri ukončovaní klastrovania na uzle dodržte tento postup:

| 1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.

| 2. Rozviňte **Klastre**.

| 3. Rozviňte klaster obsahujúci uzol, v ktorom chcete zastaviť klastrovanie.

| 4. Kliknite na **Uzly**.

| 5. Pravým tlačidlom myši kliknite na uzol, v ktorom chcete zastaviť klastrovanie a vyberte voľbu **Klastre → Zastaviť**.

### | Použitie príkazov CL a rozhraní API

| Na ukončenie uzla môžete použiť aj príkazy CL a API. Keď sú služby klastrových prostriedkov na určenom uzle úspešne zastavené, nastaví sa stav uzla na *Neaktívny*.

- | • Príkaz ENDCLUNOD (End Cluster Node)
- | • API QcstEndClusterNode (End Cluster Node)

#### | Súvisiace úlohy

| “Ukončenie klastrových úloh” na strane 111

| Úlohu klastra sa nikdy sa nepokúšajte ukončiť priamo.

| “Obnova po zlyhaní úloh klastra” na strane 138

| Zlyhanie úlohy služby klastrových prostriedkov zvyčajne naznačuje iný problém.

## Prispôsobenie klastrovej verzie pre klaster

Verzia klastra definuje úroveň, na ktorej navzájom aktívne komunikujú všetky uzly klastra.

Verziovanie klastra je metóda, ktorá umožňuje klastru, aby obsahoval systémy rôznych úrovní vydania, ktoré plne spolupracujú s použitím určenej úrovne komunikačného protokolu.

Keď chcete zmeniť verziu klastra, všetky uzly klastra musia byť v tej istej potenciálnej verzii. Verziu klastra môžete zmeniť spojením potenciálnych verzií. Tým je povolené použitie novej funkcie. Verziu klastra môžete zvyšovať iba postupne o jednu úroveň. Znížiť ju môžete iba tak, že klaster vymažete a vytvoríte ho v nižšej verzii. Aktuálna verzia klastra je na začiatku nastavená podľa prvého uzla v klastri. Ďalšie uzly, ktoré pridáte do klastra, musia mať rovnakú alebo nasledujúcu verziu, inak sa nedajú do klastra pridať.

### Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Ak chcete nastaviť verziu klastra, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Stlačte pravé tlačidlo na klastri a vyberte **Vlastnosti**.



4. Zmeňte verziu klastra na verziu, ktorú potrebujete.

### Použitie príkazov a rozhraní API pre klastre

Ak chcete nastaviť verziu klastra, môžete použiť aj nasledovné:

- Príkaz CHGCLUVER (Change Cluster Version)
- API QcstAdjustClusterVersion (Adjust Cluster Version)

#### Súvisiace koncepty

“Verzia klastra” na strane 13

*Verzia klastra* reprezentuje úroveň funkcií dostupných v klastri.

“Bežné problémy s klastrom” na strane 132

Táto téma uvádza niektoré bežné problémy, ku ktorým môže dôjsť v klastri, ako aj spôsoby, ako sa im vyhnúť alebo vykonať obnovu.

#### Súvisiace úlohy

“Vymazanie klastra”

Keď vymažete klaster, služby klastrových prostriedkov sa ukončia vo všetkých aktívnych uzloch klastra a odstránia sa z klastra.

## Vymazanie klastra

Keď vymažete klaster, služby klastrových prostriedkov sa ukončia vo všetkých aktívnych uzloch klastra a odstránia sa z klastra.

- | **Dôležité:** Ak máte v klastri nezávislé diskové oblasti, pred vymazaním klastra by ste mali najprv pomocou príkazu RMVDEVDMNE (Remove Device Domain Entry) odstrániť všetky uzly z domény zariadení.

### Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Ak chcete vymazať klaster, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Pravým tlačidlom myši kliknite na klaster, ktorý chcete vymazať a vyberte voľbu **Vymazať**.

### Použitie príkazov CL a rozhraní API

Na vymazanie klastra tiež môžete použiť príkazy CL alebo rozhrania API.

- | • Príkaz DLTCLU (Delete Cluster)
- | • API QcstDeleteCluster (Delete Cluster)

#### Súvisiace úlohy

“Prispôsobenie klastrovej verzie pre klaster” na strane 101

Verzia klastra definuje úroveň, na ktorej navzájom aktívne komunikujú všetky uzly klastra.

## Vytvorenie CRG

- | Môžete vytvoriť niekoľko typov skupín CRG: CRG aplikácie, údajov, zariadenia a CRG s partnerskými uzlami.

| Ak chcete v klastri vytvoriť CRG, vykonajte tieto kroky:

- | 1. V Navigátore iSeries rozviňte položku **Riadiaca centrála** → **Klastre**.
- | 2. Rozviňte klaster, do ktorého chcete pridať CRG.



- a. Ak chcete vytvoriť CRG zariadenia, kliknite pravým tlačidlom myši na **Prepínateľný hardvér** a vyberte voľbu **Nová skupina**. Poznámka: Voľba **Nová skupina** je dostupná, len keď sú spustené všetky uzly domény obnovy. Detaily nájdete v časti Spustenie uzla klastra.
- b. Ak chcete vytvoriť CRG aplikácie, kliknite pravým tlačidlom myši na **Prepínateľný softvér** a vyberte voľbu **Pridať produkt**.
- c. Ak chcete vytvoriť CRG údajov, kliknite pravým tlačidlom myši na **Prepínateľné údaje** a vyberte voľbu **Nová skupina**.
- d. Ak chcete vytvoriť CRG s partnerskými uzlami, kliknite pravým tlačidlom myši na **Partnerské prostriedky** a vyberte voľbu **Nové CRG s partnerskými uzlami**.

## Použitie príkazov CL a rozhraní API

Na vytvorenie CRG môžete použiť tieto príkazy a rozhrania API:

- Príkaz CRTCRG (Create Cluster Resource Group)
- API QcstCreateClusterResourceGroup (Create Cluster Resource Group)

## Vytvorenie CRG aplikácie s aktívnou adresou IP pre prevzatie

Pri vytváraní CRG aplikácie môžete nastaviť, aby sa povolila aktívna adresa IP pre prevzatie. Toto je možné, len ak užívateľ nakonfiguruje adresu IP pre prevzatie.

Predtým ste mohli vytvoriť CRG aplikácie s aktívnou adresou IP pre prevzatie za predpokladu, že ju užívateľ nakonfiguroval. Ale CRG aplikácie sa nemohlo spustiť, ak už bola adresa IP pre prevzatie aktívna. Teraz môžete pri vytváraní CRG aplikácie nastaviť, aby sa povolila aktívna adresa IP pre prevzatie. Keď spustíte CRG aplikácie, ktoré podporuje aktívne adresy IP pre prevzatie, CRG sa bude môcť spustiť.

Ak chcete pri vytváraní CRG aplikácie povoliť aktívne adresy IP pre prevzatie, vykonajte tieto kroky:

1. V rozhraní príkazového riadka zadajte:

```
CRTCRG CLUSTER(MOJKLASTER) CRG(MOJECRG) CRGTYPE(*APP) EXITPGM(QDEVELOP/EXITPGM)  
USRPRF(UZIVATEL) RCYDMN((UZOL1 *PRIMARY) (UZOL2 *BACKUP)) TKVINTNETA('10.1.2.1') CFGINTNETA(*USR *YES)
```

Parameter **TKVINTNETA** identifikuje požadovanú adresu IP pre prevzatie a parameter **CFGINTNETA** určuje, že užívateľ nakonfiguruje adresu IP pre prevzatie a že táto adresa môže byť v čase spustenia CRG aktívna.

Keď nakonfigurujete CRG aplikácie, aby sa povolila aktívna adresa IP pre prevzatie, môžete spustiť CRG.

## Spustenie CRG

Môžete spustiť niekoľko typov skupín CRG: CRG aplikácie, údajov, zariadenia a CRG s partnerskými uzlami.

Ak chcete spustiť CRG, vykonajte tieto úlohy:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála** → **Klastre**.
2. Rozviňte klaster, v ktorom chcete spustiť CRG.
  - a. Ak chcete spustiť CRG zariadenia, kliknite na **Prepínateľný hardvér**, pravým tlačidlom myši kliknite na skupinu prepínateľného hardvéru, ktorú chcete spustiť a vyberte voľbu **Spustiť**.
  - b. Ak chcete spustiť CRG aplikácie, kliknite na **Prepínateľný softvér**, pravým tlačidlom myši kliknite na prepínateľný softvérový produkt, ktorý chcete spustiť a vyberte voľbu **Spustiť**.
  - c. Ak chcete spustiť CRG údajov, kliknite na **Prepínateľné údaje**, pravým tlačidlom myši kliknite na skupinu prepínateľných údajov a vyberte voľbu **Spustiť**.
  - d. Ak chcete spustiť CRG s partnerskými uzlami, kliknite na **Partnerské prostriedky**, aby sa zobrazil zoznam skupín CRG s partnerskými uzlami, pravým tlačidlom myši kliknite na CRG, ktoré chcete spustiť a vyberte voľbu **Spustiť**.

## Použitie príkazov CL a rozhraní API

- | Na spustenie CRG môžete použiť tieto príkazy a rozhrania API:
- | • Príkaz STRCRG (Start Cluster Resource Group)
- | • API QcstStartClusterResourceGroup (Start Cluster Resource Group)

## Zmena domény obnovy pre skupinu klastrových prostriedkov

Môžete zmeniť roly uzlov v doméne obnovy pre skupinu prostriedkov klastra, ako aj pridať alebo odstrániť uzly z domény obnovy. Pre skupinu prostriedkov klastra zariadenia tiež môžete zmeniť názov lokality a adresy IP dátových portov pre uzol v doméne obnovy.

### Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Ak chcete meniť úlohy uzlov v doméne obnovy pre skupinu klastrových prostriedkov (prepínateľný hardvér, prepínateľný softvér, alebo prepínateľné údaje), alebo ak chcete pridávať a odstraňovať uzly z domény obnovy, postupujte nasledovne:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Rozviňte klaster obsahujúci prepínateľný hardvér, softvér alebo údaje, pre ktoré chcete zmeniť doménu obnovy.
4. Vyberte konkrétny prepínateľný hardvér, softvér, alebo údaje.
5. Kliknite pravým tlačidlom myši na prepínateľný hardvér, softvér, alebo údaje a označte **Vlastnosti**.
6. Vyberte stránku **Doména obnovy**.

Kliknutím na stránku Pomoc k doméne obnovy získate inštrukcie o tom, ako meniť úlohy uzlov, ako ich pridávať a odberať.

### Použitie príkazov CL a rozhraní API

Ak chcete meniť úlohu uzlov v doméne obnovy, alebo ak ich chcete pridávať a odberať, použite nasledujúce príkazy CL a rozhrania API:

- Príkaz ADDCRGNOE (Add Cluster Resource Group Node Entry)
- API QcstAddNodeToRcvyDomain (Add a Node to Recovery Domain)
- | • Príkaz CHGCRG (Change Cluster Resource Group)
- | • API QcstChangeClusterResourceGroup (Change Cluster Resource Group)
- Príkaz RMVCRGNOE (Remove Cluster Resource Group Node Entry)
- API QcstRemoveNodeFromRcvyDomain (Remove Node from Recovery Domain)

#### Súvisiace koncepty

“Doména obnovy” na strane 11

- | *Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

## Vykonanie prepnutia

Vykonanie manuálneho prepnutia spôsobí, že sa aktuálny primárny uzol prepne na záložný uzol, ako je to definované v doméne obnovy skupiny prostriedkov klastra.

Pri prepnutí sa aktuálna rola uzla v doméne obnovy skupiny prostriedkov klastra zmení takto:

- Aktuálnemu primárnemu uzlu sa priradí rola poslednej aktívnej zálohy.
- K aktuálnej prvej zálohe sa priradí primárna rola.
- Nasledujúce zálohy sa posunú o jednu v poradí zálohovania.

| Prepnutie je povolené len v skupinách CRG používajúcich model s primárnymi/záložnými uzlami, ktoré sú v aktívnom  
| stave.

**Poznámka:** Ak vykonávate prepnutie prepínateľného zariadenia (nazývaného tiež CRG zariadenia), z výkonnostných dôvodov by ste mali zosynchronizovať názov užívateľského profilu, UID a GID.

### Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Ak chcete prepnúť prostriedok - skupinu prepínateľného hardvéru, prepínateľnú aplikáciu alebo skupinu prepínateľných údajov - z primárneho uzla do záložného uzla domény obnovy, prostriedok musí mať stav **Spustený**.

Pri vykonaní prepnutia prostriedku vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Rozviňte klaster obsahujúci požadovaný prostriedok.
4. Kliknite na **Prepínateľný hardvér, Prepínateľný softvér** alebo **Prepínateľné údaje**.
5. Pravým tlačidlom myši kliknite na požadovaný prostriedok a vyberte voľbu **Prepnúť**.

### Použitie klastrových API

Na vykonanie prepnutia môžete použiť aj:

- Príkaz CHGCRGPRI (Change Cluster Resource Group Primary)
- API QcstInitiateSwitchOver (Initiate Switchover)

#### Súvisiace koncepty

“Doména obnovy” na strane 11

| *Doména obnovy* je podmnožina uzlov klastra, ktoré sú zoskupené do skupiny prostriedkov klastra (CRG) za  
| spoločným účelom, ako napríklad vykonanie akcie obnovy alebo synchronizácia udalostí.

#### Súvisiace úlohy

“Prepnutie” na strane 20

K *prepnutiu* dochádza, keď manuálne prepnete prístup k prostriedku z jedného servera na iný.

Synchronizácia názvu profilu, UID a GID

## Pridanie uzla do domény pre zariadenie

Doména zariadení je podmnožina uzlov v klasteri, ktoré zdieľajú prostriedky zariadení.

Skôr, než môže byť uzol pridaný do domény obnovy skupiny klastrových prostriedkov (CRG), musí byť najprv určený ako člen domény zariadení. Všetky uzly, ktoré budú v doméne obnovy pre CRG zariadenia, musia byť v rovnakej doméne zariadení. Uzol klastra môže patriť najviac jednej doméne zariadení.

Ak chcete vytvoriť a manažovať domény zariadení, musíte mať vo všetkých uzloch klastra, ktoré budú v doméne zariadení, nainštalovanú voľbu 41 (Vysokodostupné prepínateľné zariadenia) s platným licenčným kľúčom.

### Použitie Navigátora iSeries

Ak chcete pomocou Navigátora iSeries pridať uzol do domény zariadení, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Rozviňte klaster obsahujúci uzol, ktorý chcete pridať do domény zariadení.
4. Kliknite na **Uzly**.

5. Pravým tlačidlom myši kliknite na uzol, ktorý chcete pridať do domény zariadení a vyberte voľbu **Vlastnosti**.
6. Na stránke **Klastrovanie** zadajte do poľa **Doména zariadení** názov domény zariadení, do ktorej chcete pridať uzol.

### Použitie príkazov CL a rozhraní API

Na pridanie uzla do domény zariadení môžete použiť:

- Príkaz ADDDEVDMNE (Add Device Domain Entry)
- API QcstAddDeviceDomainEntry (Add Device Domain Entry)

#### Súvisiace koncepty

“Domény zariadení” na strane 16

*Doména zariadení* je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení. Konkrétne sa môžu uzly v doméne zariadení zúčastňovať akcie prepnutia pre určitú kolekciu obnoviteľných prostriedkov zariadení.

#### Súvisiace úlohy

“Odstránenie uzla z domény pre zariadenie”

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení.

## Odstránenie uzla z domény pre zariadenie

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení.

### Dôležité:

Buďte opatrní pri odstraňovaní uzla z domény pre zariadenia. Ak odstraňujete z domény pre zariadenia uzol, ktorý je práve primárnym bodom prístupu pre ľubovoľné nezávislé diskové oblasti, tak sa odstránia aj tieto nezávislé diskové oblasti spolu s uzlom, ktorý je odstránený. Následkom toho sa k týmto nezávislým diskovým oblastiam nebude môcť prístupovať zo žiadnych iných uzlov v doméne pre zariadenia.

Ak odstránite uzol z domény pre zariadenia, tak ho nebudete môcť pridať do tej istej domény pre zariadenia v prípade, že jeden alebo viac klastrových uzlov patria do tej istej domény pre zariadenia. Aby bolo možné pridať uzol späť do domény pre zariadenia, musíte vykonať tieto kroky:

1. Vymažte nezávislé diskové oblasti, ktoré aktuálne vlastní uzol pridávaný do domény zariadení.
2. Vykonajte reštart systému (IPL) v uzle.
3. Pridajte uzol do domény zariadení. Pozrite si časť Pridanie uzla do domény zariadení.
4. Znova vytvorte nezávislé diskové oblasti, ktoré ste vymazali v kroku 1.

### Použitie Navigátora iSeries

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Ak chcete pomocou Navigátora iSeries odstrániť uzol z domény zariadení:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Rozviňte klaster obsahujúci uzol, ktorý chcete odstrániť z domény zariadení.
4. Kliknite na **Uzly**.
5. Pravým tlačidlom myši kliknite na uzol, ktorý chcete odstrániť z domény zariadení a vyberte voľbu **Vlastnosti**.
6. Na strane Klastrovanie vymažte zadanú hodnotu v poli **Doména pre zariadenie**.

### Použitie príkazov CL a rozhraní API

Na odstránenie uzla z domény pre zariadenie môžete použiť:

- Príkaz RMVDEVDMNE (Remove Device Domain Entry)
- API QcstRemoveDeviceDomainEntry (Remove Device Domain Entry)

### Súvisiace koncepty

“Domény zariadení” na strane 16

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení. Konkrétne sa môžu uzly v doméne zariadení zúčastňovať akcie prepnutia pre určitú kolekciu obnoviteľných prostriedkov zariadení.

### Súvisiace úlohy

“Pridanie uzla do domény pre zariadenie” na strane 105

Doména zariadení je podmnožina uzlov v klastrí, ktoré zdieľajú prostriedky zariadení.

Pridanie diskovej jednotky alebo diskovej oblasti

## Ako systémová udalosť ovplyvňuje klaster

Určité príkazy na ukončovanie systémových funkcií, napríklad PWRDWN SYS (Power Down System), ENDSYS (End System) a ENDSBS (End Subsystem), môžu náhle ukončiť klaster a spôsobiť jeho rozklad.

Vo vydaní V5R4 boli vylepšené príkazy PWRDWN SYS, ENDSYS a ENDSBS. Ak je v čase vykonania týchto príkazov aktívne klastrovanie, zavolá sa API QcstEndClusterNode (End Cluster Node).

Ak chcete, aby sa tieto príkazy dokončili, mali by ste použiť voľbu OPTION(\*CNTRLD) a v parametri DELAY zadať vhodné časové oneskorenie. V opačnom prípade sa nemusí API QcstEndClusterNode (End Cluster Node) dokončiť skôr, než sa riadenie vráti funkcii na ukončenie systému.

**Poznámka:** Ak užívateľ zadá voľbu **OPTION(\*IMMED)**, API QcstEndClusterNode (End Cluster Node) má približne 30 s dokončenie pred tým, ako sa ukončí systém. To môže spôsobiť, že namiesto ukončenia uzla klastra sa vykoná núdzové prepnutie.

## Vytvorenie administratívnej domény klastra

Administratívnu doménu klastra môžete vytvoriť v Navigátore iSeries alebo pomocou príkazu CRTADMDMN (Create Cluster Administrative Domain).

Aby mohol užívateľ vytvoriť a manažovať administratívnu doménu klastra, musí mať oprávnenie na vytváranie CRG, na príkazy pre CRG a užívateľský profil QCLUSTER.

### Použitie Navigátora iSeries

Ak chcete vytvoriť administratívnu doménu klastra, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála → Klastre**.
2. Rozviňte klaster, pre ktorý chcete pridať administratívnu doménu klastra.
3. Pravým tlačidlom myši kliknite na **Partnerské prostriedky** a vyberte voľbu **Nová administratívna doména**.

### Použitie príkazov CL a rozhraní API

Na vytvorenie administratívnej domény klastra môžete použiť tieto príkazy a rozhrania API:

- Príkaz CRTADMDMN (Create Cluster Administrative Domain)
- Na vytvorenie administratívnej domény klastra neexistuje žiadne rozhranie API.

### Súvisiace koncepty

“Klastrové príkazy a API” na strane 73

Služby klastrových prostriedkov i5/OS poskytujú množinu príkazov riadiaceho jazyka (CL), aplikačných programových rozhraní (API) a funkcií, pomocou ktorých môžu poskytovatelia alebo užívatelia aplikácií iSeries zvýšiť ich dostupnosť.

## Pridanie položiek monitorovaných prostriedkov

Do administratívnej domény môžete pridať položku monitorovaného prostriedku, ktorá reprezentuje prostriedok zdieľaný medzi uzlami.

- | Ak chcete pridať položku monitorovaného prostriedku, vykonajte tieto kroky:
- | 1. V Navigátore iSeries rozviňte položku **Riadiaca centrála** → **Klastre**.
  - | 2. Rozviňte klaster, do ktorého chcete pridať položku monitorovaného prostriedku.
  - | 3. Rozviňte **Partnerské prostriedky**, aby sa zobrazil zoznam všetkých partnerských prostriedkov v klasteri.
  - | 4. Rozviňte administratívnu doménu klastra, do ktorej chcete pridať položku monitorovaného prostriedku.
  - | 5. Pravým tlačidlom myši kliknite na typ monitorovaného prostriedku a vyberte voľbu **Pridať položku monitorovaného prostriedku**.
  - | 6. Vyberte atribúty, ktoré sa majú monitorovať pre položku monitorovaného prostriedku a kliknite na **OK**.

| **Poznámka:** Ak ako položky monitorovaných prostriedkov pridávate užívateľské profily používajúce synchronizáciu hesiel, systémová hodnota Zachovať bezpečnosť servera (QRETSVRSEC) musí byť nastavená na 1.

### | Použitie príkazov CL a rozhraní API

| Na pridanie monitorovaných prostriedkov môžete použiť tieto príkazy a rozhrania API:

- | • Pre túto funkciu neexistuje žiadny ekvivalentný príkaz CL. Knižnica QUSRTOOL obsahuje zdroj pre nepodporovaný príkaz a program, spracúvajúci príkaz (CPP). Ak sa chcete dozvedieť o tomto zdroji príkazov a CPP, pozrite si člen QFPADINFO v súbore QATTINFO.
- | • API QfpadAddMonitoredResourceEntry (Add Monitored Resource Entry)

## | Monitorovanie administratívnej domény klastra

| Po vytvorení administratívnej domény klastra a pridaní požadovaných položiek monitorovaných prostriedkov by mal administrátor klastra monitorovať aktivitu v administratívnej doméne, aby zabezpečil neustálu konzistentnosť monitorovaných prostriedkov.

| Ak je globálny stav monitorovaného prostriedku nekonzistentný, administrátor by mal vykonať potrebné kroky na určenie príčiny nekonzistencie prostriedku, opravenie problému a obnovenie synchronizácie prostriedku.

| Ak je prostriedok nekonzistentný kvôli zlyhaniu aktualizácie v jednom alebo viacerých uzloch, pre MRE sú uchované informácie, ktoré vám pomôžu určiť príčinu zlyhania. V uzle, kde došlo k zlyhaniu, sa zaprotokoluje správa s MRE, ktoré je príčinou zlyhanej aktualizácie. V ostatných uzloch sa zaprotokoluje správa oznamujúca, že došlo k zlyhaniu, spolu so zoznamom uzlov, kde aktualizácia zlyhala.

| Po určení príčiny nekonzistencie je možné obnoviť synchronizáciu prostriedku buď ako dôsledok operácie aktualizovania v uzle, kde došlo k zlyhaniu, alebo ukončením a opakovaným spustením administratívnej domény.

| Keď vymažete, premenujete alebo presuniete monitorovaný prostriedok v ľubovoľnom uzle domény, jeho globálny stav sa vždy nastaví na nekonzistentný. V tomto prípade by ste mali odstrániť MRE, pretože administratívna doména klastra už nebude vykonávať synchronizáciu prostriedku.

### | Použitie Navigátora iSeries

| Ak chcete monitorovať administratívnu doménu klastra, vykonajte tieto kroky:

- | 1. V Navigátore iSeries rozviňte položku **Riadiaca centrála** → **Klastre**.
- | 2. Rozviňte klaster, ku ktorému je priradená administratívna doména klastra.
- | 3. Rozviňte **Partnerské prostriedky** a pravým tlačidlom myši kliknite na voľbu **Nová administratívna doména**, potom vyberte **Vlastnosti**. Možné hodnoty pre stav prostriedku v rámci aktívnej administratívnej domény klastra sú:

#### | **Konzistentný**

| Hodnoty všetkých systémom monitorovaných atribútov prostriedku sú rovnaké vo všetkých aktívnych uzloch administratívnej domény klastra.



### **Nekonzistentný**

Hodnoty všetkých systémom monitorovaných atribútov prostriedku nie sú rovnaké vo všetkých aktívnych uzloch administratívnej domény klastra alebo administratívna doména klastra nie je aktívna.

### **Prebieha**

Hodnoty monitorovaných atribútov v administratívnej doméne klastra sú v procese synchronizácie.

### **Pridaný**

Položka monitorovaného prostriedku sa pridala do adresára monitorovaných prostriedkov v administratívnej doméne klastra, ale ešte sa nevykonala jej synchronizácia.

## **Použitie príkazov CL a rozhraní API**

Na monitorovanie administratívnej domény klastra môžete použiť tieto príkazy a rozhrania API:

- Pre túto funkciu neexistuje žiadny podporovaný ekvivalentný príkaz CL. Knižnica QUSRTOOL obsahuje zdroj pre nepodporovaný príkaz a program, spracúvajúci príkaz (CPP). Ak sa chcete dozvedieť o tomto zdroji príkazov a CPP, pozrite si člen QFPADINFO v súbore QATTINFO.
- API QfpadRtvMonitoredResourceInfo (Retrieve Monitored Resource Information)

## **Monitorovanie stavu klastra**

Služby klastrových prostriedkov vykonávajú základné monitorovanie klastra a jeho komponentov pomocou funkcie garantovaných správ a monitorovania kontrolného signálu a v prípade potreby vykonávajú vhodné akcie.

Stav klastra a jeho komponentov môžete monitorovať aj manuálne.

### **Použitie Navigátora iSeries**

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (Vysokodostupné prepínateľné prostriedky).

Ak chcete monitorovať stav klastra pomocou Navigátora iSeries:

1. V Navigátore iSeries rozviňte položku Riadiaca centrála.
2. Rozviňte **Klastre**.
3. V zložkách Navigátora iSeries prejdite k požadovanému klastru, aby sa v stĺpci Stav v Navigátore iSeries zobrazil stav klastra, jeho uzlov a prostriedkov. Popisy možných hodnôt v stĺpci Stav vám ponúka online pomoc. Môžete tiež na komponent klastra kliknúť pravým tlačidlom myši, vybrať **Vlastnosti** a prezrieť si informácie o tomto klastru.

### **Použitie príkazov CL a rozhraní API**

Na monitorovanie stavu klastra môžete použiť aj tieto príkazy a API:

#### **Informácie o klastru**

Získava informácie o klastru, ako napríklad uzly v klastru, na ktorom uzle budú použité aké IP adresy adaptéra a stav každého uzla v klastru.

- Príkaz DSPCLUINF (Display Cluster Information)
- API QcstListClusterInfo (List Cluster Information)
- API QcstListDeviceDomainInfo (List Device Domain Info)
- API QcstRetrieveCRSInfo (Retrieve Cluster Resource Services Information)
- API QcstRetrieveClusterInfo (Retrieve Cluster Information)

#### **Informácie o skupine prostriedkov klastra**

Vytvorí zoznam skupín klastrových prostriedkov a informácií o skupine klastrových prostriedkov v klastru, ako napríklad názov primárneho uzla pre každú CRG v klastru.

- Príkaz DSPCRGINF (Display Cluster Resource Group Information)
- API QcstListClusterResourceGroups (List Cluster Resource Groups)



- API QcstListClusterResourceGroupInf (List Cluster Resource Group Information)

### Súvisiace koncepty

“Funkcia spoľahlivých správ” na strane 26

*Funkcia spoľahlivých odkazov* služby klastrových prostriedkov sleduje každý uzol v klastri a zabezpečuje, že všetky uzly majú nepretržité informácie o stave klastrových prostriedkov.

“Sledovanie aktivity” na strane 25

*Monitorovanie kontrolného signálu* je funkcia služieb klastrových prostriedkov, ktorá zabezpečuje, že každý uzol je aktívny. Funkcia odosiela z každého uzla v klastri signál do všetkých ostatných uzlov v klastri na potvrdenie, že sú uzly stále aktívne.

## Výkon klastra

Keď sú vykonané zmeny na klastri, môže to ovplyvniť réžiu spravovania klastra.

Klastrovanie vyžaduje len tie prostriedky, ktoré sú potrebné na vykonávanie monitorovania kontrolného signálu, manažovanie skupín prostriedkov klastra a uzlov klastra a na spracúvanie výmeny správ medzi skupinami prostriedkov klastra a uzlami klastra. Ak je vaše klastrovacie prostredie funkčné, jediné vylepšenie réžie môže nastať, ak urobíte zmeny na klastri.

V rámci normálneho fungujúceho prostredia by ste tak dosiahli minimálny zásah do činnosti klastrovaných systémov.

### Súvisiace koncepty

“Sledovanie aktivity” na strane 25

*Monitorovanie kontrolného signálu* je funkcia služieb klastrových prostriedkov, ktorá zabezpečuje, že každý uzol je aktívny. Funkcia odosiela z každého uzla v klastri signál do všetkých ostatných uzlov v klastri na potvrdenie, že sú uzly stále aktívne.

“Bežné problémy s klastrom” na strane 132

Táto téma uvádza niektoré bežné problémy, ku ktorým môže dôjsť v klastri, ako aj spôsoby, ako sa im vyhnúť alebo vykonať obnovu.

## Vyváženie zaťaženia siete pre klastre

Zaťaženie siete môžete rozložiť tak, že rozdelíte prácu medzi komunikačné linky, pomocou ktorých sú prepojené uzly v klastri.

Čím viac vyvážite zaťaženie s čo najmenším využitím prostriedkov, tým bezproblémovjšie bude fungovať váš systém.

### Zaťaženie CPU na záložných uzloch:

Záložné systémy by ste mali využívať v maximálnej možnej miere, ale berte na vedomie, že v prípade núdzového prepnutia sa môže do záložného uzla presunúť dodatočné pracovné zaťaženie.

Je veľmi dôležité, aby ste si boli vedomí, čo je dôležité a kritické pre vašu prácu a čo nie. Ak máte mimoriadne kritické aplikácie, ktoré zlyhávajú, musíte zabezpečiť, aby zaťaženie CPU na vašich záložných uzloch nebolo také veľké, že kritické aplikácie nemôžu bežať.

## Výkonnosť optimalizovaného klastra

Pretože existujú potenciálne podstatné rozdiely vo vašom komunikačnom prostredí, máte možnosť prispôsobiť premenné, ktoré sa týkajú klastrovej komunikácie, aby čo najlepšie vyhovovali vášmu prostrediu.

Štandardné hodnoty by mali byť akceptovateľné pre väčšinu bežných prostredí. Ak vaše špecifické prostredie nie je veľmi vhodné pre tieto štandardy, môžete si klastrovú komunikáciu prispôsobiť, aby lepšie vyhovovala vášmu prostrediu. Sú možné dve úrovne prispôbenia.

## Ladenie na základnej úrovni

Ladenie na základnej úrovni vám umožňuje nastaviť ladiace parametre na preddefinovanú množinu hodnôt, určených pre vysokú, nízku a normálnu úroveň časového limitu a intervalu výmeny správ. Keď je vybratá normálna úroveň, pre parametre výkonu komunikácií klastra a konfiguráciu sa použijú štandardné hodnoty. Výber nízkej úrovne spôsobí zvýšenie intervalu kontrolných signálov klastrovania a rôznych hodnôt prekročenia časového limitu správ. S menším počtom kontrolných signálov a dlhšími hodnotami časového limitu bude klaster menej citlivý na komunikačné zlyhania. Výber vysokej úrovne spôsobí, že klastrovanie zníži interval kontrolných signálov a rôzne hodnoty časového intervalu správ. S častejšími kontrolnými signálmi a kratšími časovými limitmi bude klaster citlivejší na komunikačné zlyhania.

## Ladenie na rozšírenej úrovni

Je dostupné aj ladenie na rozšírenej úrovni, pri ktorom môžete nastaviť jednotlivé parametre v rámci preddefinovaných rozsahov hodnôt. Toto umožňuje jemnejšie prispôbenie, aby sa vyhovelo všetkým zvláštnym požiadavkám vášho komunikačného prostredia. Ak potrebujete vykonať ladenie na rozšírenej úrovni, odporúča sa, aby ste využili pomoc od personálu podpory IBM a podobne. Nesprávne nastavenie jednotlivých hodnôt môže ľahko spôsobiť zníženie výkonu.

### Súvisiace koncepty

“Laditeľné parametre komunikácií klastra” na strane 94

API QcstChgClusterResourceServices (Change Cluster Resource Services) umožňuje vyladenie niektorých služieb topológie klastra a konfiguračných a výkonových parametrov pre komunikáciu v klastru, aby lepšie zodpovedali rôznym jedinečným aplikačným a sieťovým prostrediam, kde sa používa klastrovanie. Takéto API je k dispozícii akémukoľvek klastru, ktorý je spustený na klastrovej verzii 2 alebo novej.

### Súvisiaci odkaz

API QcstChgClusterResourceServices (Change Cluster Resource Services)

## Ukončenie klastrových úloh

Úlohu klastra sa nikdy sa nepokúšajte ukončiť priamo.

Ak potrebujete zastaviť čokoľvek, čo je spustené v klastrovom prostredí, mali by ste:

1. Ukončiť uzol klastra.
2. Vyriešiť problém.
3. Spustiť uzol klastra.

### Súvisiace úlohy

“Ukončenie klastrového uzla” na strane 101

Zastavenie, alebo ukončenie uzla, zastavuje na uzle služby klastrových prostriedkov.

“Spustenie klastrového uzla” na strane 100

Spustenie klastrového uzla spustí na tomto uzle služby klastrových prostriedkov. Od verzie 3 sa uzol môže spustiť sám a bude schopný opätovne sa pripojiť k aktívnemu klastru, ak v ňom nájde aktívny uzol.

## Monitorovanie a riadenie prostriedkov (RMC)

Monitorovanie a riadenie prostriedkov (RMC) je všeobecná kostra na manažovanie, monitorovanie a manipulovanie s prostriedkami, napríklad fyzickými alebo logickými entitami systému.

RMC sa používa ako komunikačný mechanizmus na nahlasovanie servisných udalostí hardvérovej riadiacej konzole (HMC). Ak RMC nie je aktívne, servisné udalosti sa nebudú nahlasovať konzole HMC. Tento zoznam opisuje služby, ktoré sú priradené k RMC:

### Démon CAS

**Účel:** Slúži ako autentifikačný server pre RMC.

**Názov úlohy:** QCSTCTCASD

### **Démon RMC**

**Účel:** Pomocou komunikácie so správcami prostriedkov monitoruje prostriedky.

**Názov úlohy:** QCSTCTRMCD

### **Démon SRC**

**Účel:** Monitoruje stav ostatných úloh RMC; ak sa niektorá úloha neočakávanie skončí, znova ju spustí.

**Názov úlohy:** QCSTSRCD

## **Správcovia prostriedkov (RM)**

Správca prostriedkov (RM) je úloha, ktorá manažuje a poskytuje rozhranie medzi RMC a konkrétnymi fyzickými alebo logickými entitami. Aj keď RMC poskytuje základné abstrakcie, napríklad triedy prostriedkov, prostriedky a atribúty na reprezentáciu fyzických alebo logických entít, samo nereprezentuje žiadne skutočné entity. RM mapuje skutočné entity na abstrakcie RMC. Tento zoznam opisuje rôznych správcov prostriedkov, ktorých podporuje RMC:

### **RM auditovacieho protokolu**

**Účel:** Poskytuje funkcie na záznam informácií o prevádzke systému.

**Názov úlohy:** QYUSALRMD

### **RM CSMAgent**

**Účel:** Poskytuje triedy prostriedkov na reprezentáciu riadiaceho servera, ktorým je HMC.

**Názov úlohy:** QYUSCMCRMD

### **RM hostiteľa**

**Účel:** Poskytuje triedy prostriedkov na reprezentáciu jednotlivých počítačov.

**Názov úlohy:** QCSTCTHRMD

### **Servisné RM**

**Účel:** Manažuje informácie o probléme a pripravuje ich na doručenie konzole HMC.

**Názov úlohy:** QSVRMSERMD

## **Spustenie alebo ukončenie RMC**

Všetky úlohy RMC, vrátane úloh RM, sa nachádzajú v podsystéme QSYSWRK a spúšťajú sa automaticky pri jeho spustení. Aby sa spustenie úspešne dokončilo, musí byť aktívne TCP/IP. Démon RMC vyžaduje aktívne TCP/IP. Keď prejde TCP/IP do neaktívneho stavu, démon RMC sa ukončí. Keď sa TCP/IP znova aktivuje, démon SRC automaticky znova spustí démon RMC. Za normálnych okolností nemusí užívateľ vykonať žiadne kroky. Ak potrebujete manuálne spustiť RMC, vykonajte tento príkaz:

```
SBMJOB CMD(CALL PGM(QSYS/QCSTCTSRCD)) JOBD(QSYS/QCSTSRCD) PRTDEV(*JOB) OUTQ(*JOB)
USER(*JOB) PRTTXT(*JOB) RTGDTA(RUNPTY50)
```

Ak potrebujete manuálne ukončiť RMC, pomocou príkazu ENDJOB ukončíte úlohu QCSTSRCD. Tento príkaz by mal ukončiť všetky úlohy RMC. Ak sa neukončia všetky úlohy, manuálne ukončíte každú z vyššie uvedených úloh.

## **Úloha štruktúry a užívateľské fronty**

Pri manažovaní klastra musíte poznať štruktúry úloh a užívateľské fronty.

### **Štruktúra úlohy služieb klastrových prostriedkov**

Služby klastrových prostriedkov sa skladajú z množiny súbežných úloh. Keď je na servere aktívne klastrovanie, nasledujúce úlohy sa spúšťajú v QSYSWRK subsystéme pod QSYS užívateľským profilom. Úlohy sa vykonávajú pomocou opisu úlohy QDFTSVR, ale s úrovňou protokolovania nastavenou tak, aby sa produkoval protokol úlohy.

- Kontrola klastra sa skladá z jednej úlohy, ktorá sa nazýva QCSTCTL.
- Riadenie skupiny klastrových prostriedkov sa skladá z jednej úlohy, ktorá sa nazýva QCSTCRGM.

- Skupina klastrových prostriedkov pozostáva z jednej úlohy na každý objekt skupiny klastrových prostriedkov. Názov úlohy je rovnaký ako názov skupiny klastrových prostriedkov. Toto zahŕňa administratívnu doménu klastra.
- Keď je odolnej CRG pre zariadenia nastavená jedna alebo viac položiek zo zoznamu zariadení tak, aby online synchronizovali pri prepnutí alebo prevzatí úlohy, odošlú sa ďalšie úlohy, aby vykonali zmenu vo funkcii.

Úlohy QCSTCTL a QCSTCRGM sú rozhodujúcimi úlohami klastra. To znamená, že úlohy musia bežať v danom poradí, aby bol uzol v klastru aktívny.

Väčšina rozhraní API skupín klastrových prostriedkov má za následok odoslanie osobitných úloh, ktoré používajú označený užívateľský profil, keď je vytvorená skupina klastrových prostriedkov. V odoslaných úlohách je volaný ukončovaci program, ktorý je definovaný v skupine klastrových prostriedkov. Štandardne sú úlohy odoslané do frontu úloh QBATCH. Obvykle je tento front úloh použitý pre tvorbu dávkových úloh a oneskorí alebo zamedzí dokončenie ukončovacích programov. Aby mohli byť API spustené efektívne, vytvorte osobitný užívateľský profil, popis úlohy a front úloh, ktoré použijú skupiny klastrových prostriedkov. Špecifikujte nový užívateľský profil pre všetky skupiny klastrových prostriedkov, ktoré vytvoríte. Ten istý program sa spracúva vo všetkých uzloch domény obnovy, ktorá je definovaná pre skupinu klastrových prostriedkov.

- | Ak chcete opakovane spustiť ukončenú úlohu skupiny prostriedkov klastra bez toho, aby ste ukončili a znova spustili klastrovanie v uzle, môžete použiť príkaz CHGCLURCY (Change Cluster Recovery).

## Používanie užívateľských frontov v rozhraniach API pre klastre

- | Funkcie vykonávané pomocou API, ktoré má parameter informácií o výsledku, fungujú asynchrónne a po dokončení spracovania API odosielajú svoje výsledky do užívateľského frontu. Užívateľský front musí byť vytvorený pred volaním API. Užívateľský front môžete vytvoriť pomocou API QUSCRTUQ (Create User Queue). Front musí byť vytvorený ako kľúčovaný front. Kľúč pre užívateľský front je popísaný vo formáte položky frontu užívateľa. Meno užívateľského frontu je poslané do API. Dokumentácia Rozhrania API pre klastre obsahuje príklady používania užívateľských frontov s týmito rozhraniami API.

Pri používaní API QcstDistributeInformation (Distribute Information) sa informácie odosielené medzi uzlami odkladajú do užívateľského frontu, zadaného pri vytvorení CRG. Pred použitím API QcstDistributeInformation (Distribute Information) musí užívateľ vytvoriť tento front vo všetkých aktívnych uzloch domény obnovy. Detaily o tom, kedy musí existovať front na distribúciu informácií, nájdete v téme API QcstCreateClusterResourceGroup (Create Cluster Resource Group).

Front správ o prevzatí služby dostáva správy ohľadne činností prevzatia služby.

### Súvisiace koncepty

“Udržiavanie užívateľských profilov vo všetkých uzloch” na strane 90

Na údržbu užívateľských profilov vo všetkých uzloch klastra môžete použiť dva mechanizmy.

“Identifikácia problémov s klastrami” na strane 121

Tu začnete určovať vaše klastrové problémy.

### Súvisiace úlohy

“Obnova po zlyhaní úloh klastra” na strane 138

Zlyhanie úlohy služby klastrových prostriedkov zvyčajne naznačuje iný problém.

## Front správ o prevzatí služby

Front správ o prevzatí služby dostáva správy ohľadne činností prevzatia služby.

Front správ o prevzatí služby umožňuje upozorniť administrátora ešte pred prevzatím služby. Administrátor tak môže zrušiť prevzatie služby, ak nie je žiaduce.

Front správ pre núdzové prepnutie sa definuje pri vytváraní skupiny prostriedkov klastra pomocou API QcstCreateCluster (Create Cluster). Môže byť modifikovaný aj CL príkazom a API na zmenu skupiny klastrových prostriedkov. Front správ pre núdzové prepnutie nie je možné použiť s rozhraním na manažovanie klastrov v Navigátore iSeries.

Viac detailov o fronte správ pre núdzové prepnutie nájdete v dokumentácii k API pre skupinu prostriedkov klastra. V nasledujúcich popisoch nájdete detaily o tom, ako používať front správ o prevzatí služby.

## Príkazy CL

- Príkaz CRTCRG (Create Cluster Resource Group)
- Príkaz CHGCRG (Change Cluster Resource Group)

## Rozhrania API

- API QcstCreateClusterResourceGroup (Create Cluster Resource Group)
- API QcstChangeClusterResourceGroup (Change Cluster Resource Group)

## Udržiavanie užívateľských profilov vo všetkých uzloch

Na údržbu užívateľských profilov vo všetkých uzloch klastra môžete použiť dva mechanizmy.

Jedným mechanizmom je vytvorenie administratívnej domény klastra na monitorovanie zdieľaných prostriedkov v uzloch klastra. Administratívna doména klastra môže okrem užívateľských profilov monitorovať rôzne iné typy prostriedkov, čo zabezpečuje jednoduché manažovanie prostriedkov, ktoré sa zdieľajú medzi uzlami. Detaily o týchto prostriedkoch nájdete v téme Monitorované prostriedky. Ak je administratívna doména klastra aktívna a užívateľské profily sa zaktualizujú, zmeny sa automaticky rozšíria do ostatných uzlov. Ak administratívna doména nie je aktívna, zmeny sa rozšíria po jej aktivovaní.

**Poznámka:** Ak plánujete zdieľať užívateľské profily používajúce synchronizáciu hesiel v klastri, musíte nastaviť systémovú hodnotu Zachovať bezpečnosť servera (QRETSVRSEC) na 1.

Pri použití druhého mechanizmu môžu administrátori pomocou Riadiacej centrály v Navigátore iSeries vykonávať tiež funkcie pre viacero systémov a skupín systémov. Táto podpora obsahuje niektoré spoločné úlohy správy užívateľov, ktoré musia operátori vykonať na viacerých systémoch v ich klastroch. Riadiaca centrála umožňuje, aby boli funkcie užívateľského profilu vykonané proti skupine systémov. Správca môže špecifikovať príkaz, ktorý sa vykoná na cieľových systémoch, keď vytvárate užívateľský profil.

## Zálohovanie a obnova klastrov

Aj ak vo vašich systémoch používate klastrovanie, stále je dôležité, aby ste vytvorili stratégiu zálohovania a obnovy na ochranu vašich údajov.

Ak zamýšľate použiť klastrovanie ako vašu zálohovaciu stratégiu tým spôsobom, že jeden systém je v prevádzke, zatiaľ čo iný systém je nefunkčný pokiaľ je v procese zálohovania, je vhodné, aby ste mali v klastri minimálne tri systémy. Tak budete mať v prípade zlyhania vždy jeden systém, na ktorý je možné urobiť prepnutie.

### Ukladanie a obnovovanie skupín klastrových prostriedkov

Ukladať skupinu klastrových prostriedkov je možné, keď je klaster aktívny alebo neaktívny. Pre obnovu skupiny klastrových prostriedkov platia nasledovné obmedzenia:

- Ak je klaster funkčný a skupina klastrových prostriedkov je klastru známa, nemôžete obnoviť skupinu klastrových prostriedkov.
- Ak uzol nie je nakonfigurovaný pre klaster, nemôžete obnoviť skupinu klastrových prostriedkov.

Skupinu klastrových prostriedkov môžete obnoviť, ak je klaster aktívny, tento klaster nepozná túto skupinu klastrových prostriedkov, uzol je v doméne obnovy tejto skupiny klastrových prostriedkov a názov klastra sa zhoduje s tým, ktorý je uvedený v tejto skupine klastrových prostriedkov. Skupinu klastrových prostriedkov môžete obnoviť, ak ja klaster nakonfigurovaný, ale nie je aktívny na tomto uzle, a ak tento uzol je v doméne obnovy tejto skupiny klastrových prostriedkov.

### Príprava na katastrofu

V prípade havárie budete musieť rekonfigurovať klaster. Aby ste na to boli pripravení, odporúčame vám, aby ste si uložili konfiguračné informácie vášho klastra a aby ste si odložili aj ich vytlačenú verziu.

1. Po vykonaní zmien konfigurácie klastra použite príkaz SAVCFG (Save Configuration) alebo príkaz SAVSYS (Save System), aby boli obnovené interné informácie o klastru konzistentné s ostatnými uzlami v klastru. Detaily o príkazoch SAVCFG a SAVSYS nájdete v téme Uloženie konfiguračných informácií.

2. Zakaždým, keď zmeníte konfiguračné informácie klastra, vytlačte si ich kópiu. Pomocou príkazu DSPCLUINF (Display Cluster Information) môžete vytlačiť konfiguráciu klastra. Jej kópiu uchovajte spolu so záložnými páskami, aby ste ju mohli použiť v prípade havárie, keď budete chcieť znova nakonfigurovať celý váš klaster.

#### Súvisiace koncepty

“Obnova klastrov zo záložných pásk” na strane 140

Počas normálnej prevádzky by nikdy nemalo byť nutné vykonať obnovu zo záložnej pásky.

Uloženie konfiguračných informácií

“Obnova klastrov po úplnej strate systému” na strane 140

Tieto informácie môžete spolu s vhodným kontrolným zoznamom z manuálu Zálohovanie a obnova použiť na obnovu celého systému po úplnej strate systému, ak dôjde k neočakávanému prerušeniu napájania vášho servera.

“Uloženie konfigurácií klastra”

Pomocou príkazov môžete uložiť vaše objekty skupín prostriedkov klastra.

“Obnova klastrov po havárii” na strane 140

V prípade havárie, pri ktorej ste stratili všetky uzly, potrebujete svoj klaster rekonfigurovať.

#### Súvisiace úlohy

Plánovanie stratégie zálohovania a obnovy

Tlač informácií o systéme

## Uloženie konfigurácií klastra

Pomocou príkazov môžete uložiť vaše objekty skupín prostriedkov klastra.

Pomocou príkazu SAVSYS (Save System) môžete uložiť celý systém, nie len nakonfigurovaný klaster. Na uloženie nakonfigurovaného klastra môžete použiť príkaz SAVCFG (Save Configuration).

```
SAVOBJ(QUSRSYS/*ALL) OBJTYPE (*CRG)
```

**Poznámka:** Objekty skupiny klastrových prostriedkov môžu byť uložené len pre aktuálne vydanie.

#### Súvisiace úlohy

“Zálohovanie a obnova klastrov” na strane 114

Aj ak vo vašich systémoch používate klastrovanie, stále je dôležité, aby ste vytvorili stratégiu zálohovania a obnovy na ochranu vašich údajov.

#### Súvisiaci odkaz

Príkaz SAVSYS (Save System)

Príkaz SAVCFG (Save Configuration)

---

## Príklad: Konfigurácie klastra

Tieto príklady typických implementácií klastrov vám pomôžu pochopiť prečo, kedy a ako môže byť užitočné používať klastre.

### Príklad: Jednoduchý dvojuzlový klaster

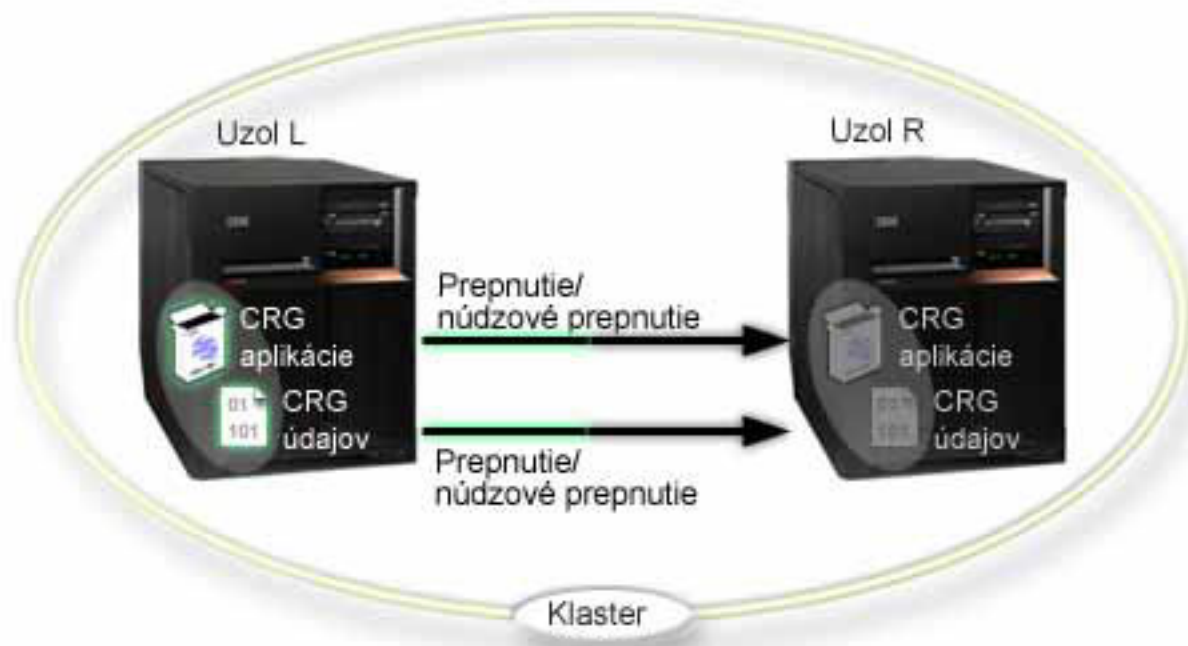
Tento príklad konfigurácie opisuje základný klaster, ktorý obsahuje dva uzly.

Táto ukážka konfigurácie poskytuje nasledovné:

- Jednosmerná replikácia a prevzatie služby



- Dvojvrstvé prostredie
- Spoločný presun aplikácií a údajov
- Zálohovanie používané pre offline spracovanie údajov
- Nepretržitá prevádzka v CRG s partnerskými uzlami



V tomto príklade je Uzol L primárnym uzlom dvoch skupín klastrových prostriedkov, aplikačnej CRG a údajovej CRG. Obsahuje tiež CRG s partnerskými uzlami, ktoré zabezpečuje nepretržitú prevádzku oboch uzlov. V Uzle L budú periodicky spustené dva ukončovacie programy aplikačnej CRG. Dôvodom pre dva ukončovacie programy bežiacie v rovnakom čase je situácia, keď zavoláte spustenie CRG API, spustí sa ukončovací program a beží plynule, pokiaľ je aplikačná CRG aktívna. Ak zavoláte ukončenie CRG API aplikačnej CRG, spustí sa ďalší ukončovací program. Uzol R je prvým a jediným zálohovým uzlom, ktorý je navrhnutý pre každú skupinu klastrových prostriedkov. Údaje, ktoré sú priradené k CRG údajov a súvisiace informácie aplikácie, ktorú sú priradené k CRG aplikácie, sa replikujú z Uzla L do Uzla R. Ak uzol L zlyhá alebo sa musí vypnúť pre administratívne účely, vykoná sa prepnutie alebo núdzové prepnutie a uzol R sa stane primárnym uzlom pre CRG aplikácií aj CRG údajov. Uzol R prevezme adresu internetového protokolu (IP), ktorá je definovaná pre aplikačnú CRG.

**Poznámka:** Kým je Uzol L vypnutý nie je dostupnosť systému chránená, pretože neexistuje žiaden záložný uzol, ak by došlo k zlyhaniu Uzla R. Keď sa Uzol L obnoví a pripojí ku klastru, nastaví sa ako záložný uzol pre obe skupiny prostriedkov klastra. Pritom sa vykoná replikácia z Uzla R do Uzla L. Ak chcete aby, Uzol L obnovil svoju rolu primárneho uzla, mali by ste vykonať administratívne prepnutie.

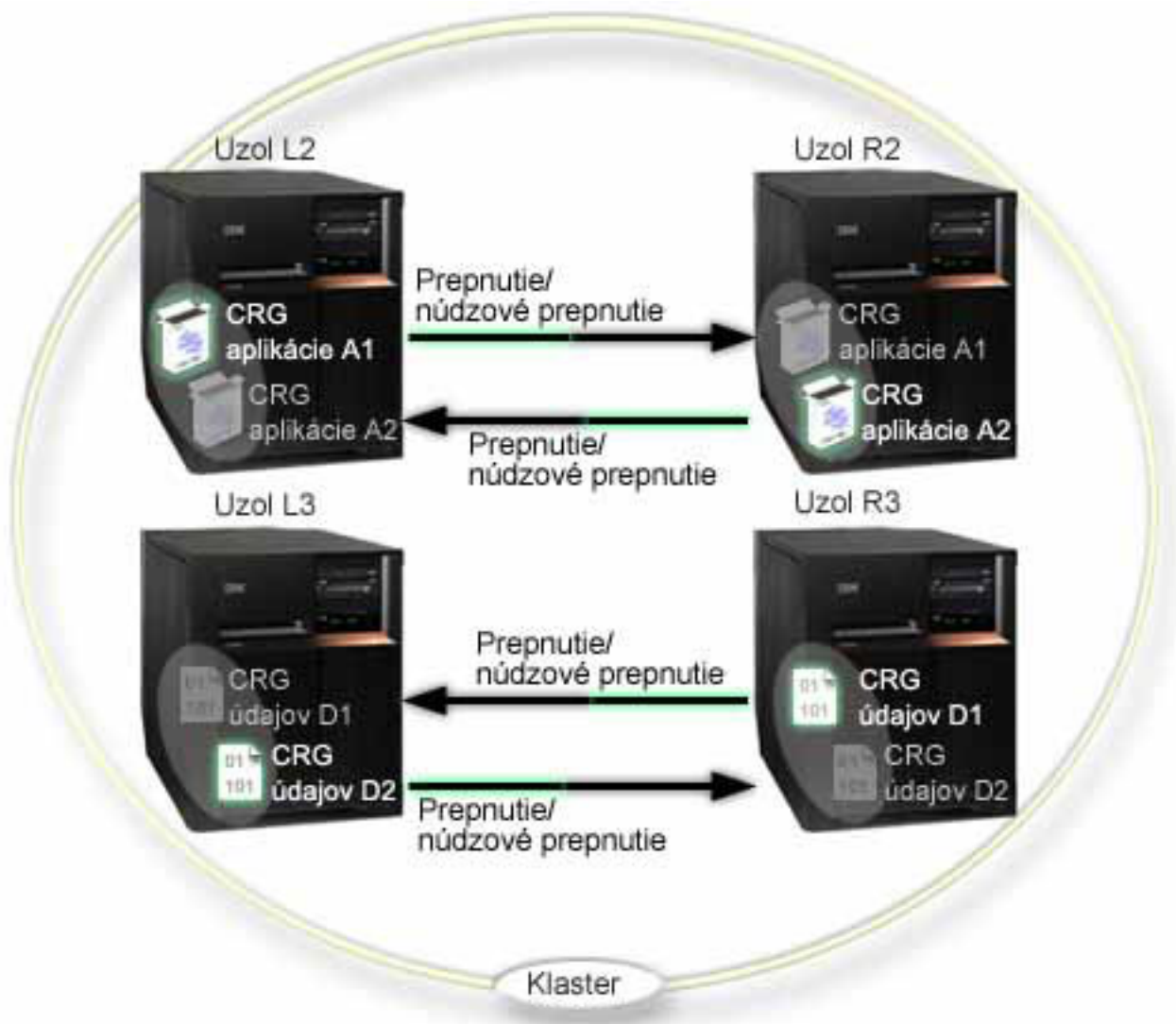
## Príklad: Štvoruzlový klaster

Pomocou tohto príkladu môžete vytvoriť komplexnejší klaster, ktorý bude obsahovať štyri uzly.

Táto ukážka konfigurácie poskytuje nasledovné:

- Dvojsmernú replikáciu a zotavenie po zlyhaní
- Trojvrstvé prostredie
- Aplikácie a údaje sa presúvajú nezávisle
- Zálohovanie je použité na normálnu produkciu rozličných pracovných zaťažení





Tento príklad so štyrmi uzlami ukazuje ďalšiu flexibilitu, ktorú poskytuje klaster iSeries. Sú tam dve aplikačné skupiny klastrových prostriedkov (A1 a A2) a dve údajové skupiny klastrových prostriedkov (D1 a D2). Údaje spojené s D1 sú kritickými údajmi aplikácie spojenej s A1. Údaje spojené s D2 sú kritickými údajmi pre aplikáciu spojenú s A2. Keďže ide o trojvrstvé prostredie, existuje aplikácia na druhej vrstve (Uzol L2 a Uzol R2) a údaje sú oddelené v tretej vrstve (Uzol L3 a Uzol R3).

Skupina prostriedkov klastra (CRG)	Primárny	Záložný
CRG aplikácií A1	L2	R2
CRG aplikácií A2	R2	L2
CRG údajov D1	R3	L3
CRG údajov D2	L3	R3

Toto umožňuje možnosť vzájomného prevzatia na aplikačnej, aj údajovej úrovni. Všetky štyri uzly sú použité pre normálnu produkciu. Tiež sú použité na zálohovanie iných systémov v klastri. V klastri by mali byť stále prístupné tieto dve aplikácie a s nimi spojené údaje. Samostatný výpadok ktoréhokoľvek uzla dostupnosť nenaruší. Dostupnosť navyše nenaruší ani súčasný výpadok jedného uzla na aplikačnej a jedného na údajovej úrovni.

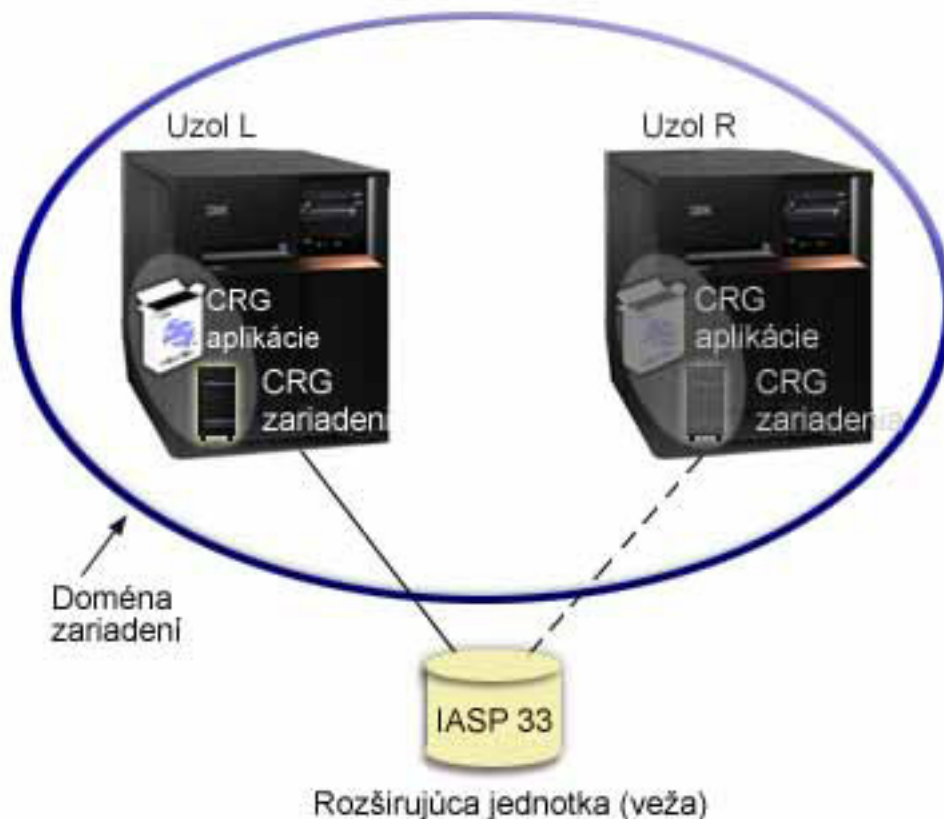
**Poznámka:** V akomkoľvek prípade je vtedy klaster nechránený a kým uzol nefunguje, nebudú niektoré klastrové prostriedky replikované. Tomu sa môžete vyhnúť, ak budete mať na každý kritický klastrový prostriedok viac, než jednu zálohu.

## Príklad: Klaster s prepínateľným diskom s použitím nezávislých diskových oblastí

Klaster, ktorý používa technológiu prepínania diskov, poskytuje alternatívu k replikácii údajov. V klastri s prepínateľnými diskami sú už údaje obsiahnuté v nezávislých diskových oblastiach (známych aj ako nezávislé ASP).

Táto ukážka konfigurácie poskytuje nasledovné:

- Jednu prepínateľnú diskovú oblasť s nevyužitým záložným serverom. Táto nezávislá disková oblasť je súčasťou skupiny prepínateľných diskových jednotiek.
- Dvojvrstvé prostredie
- Spoločný presun aplikácií a údajov
- Zálohovanie použité pri rozličných pracovných vyťaženiach nie je s týmito aplikačnými údajmi spojené
- Žiadna replikácia údajov; v tomto klastri existuje len jedna kópia týchto údajov



Podľa tohto príkladu patria Uzol L a Uzol R do rovnakej domény zariadení. Uzol L práve funguje ako primárny uzol pre dve skupiny klastrových prostriedkov - CRG aplikácií a CRG zariadení. Uzol R je prvý (a jediný) záložný uzol pre obe skupiny klastrových prostriedkov. Údaje, spojené s CRG zariadením, sú súčasťou prepínateľného zariadenia, ako napríklad externej skladovacej jednotky (veža). Relevantné informácie o aplikácii spojené s CRG aplikácií sú uložené buď v rovnakej veži, alebo sú replikované z Uzla L na Uzol R. Ak Uzol L zlyhá, alebo musí byť z administratívnych dôvodov vypnutý, stane sa Uzol R primárnym uzlom pre obe skupiny klastrových prostriedkov. Uzol R prevezme adresu protokolu IP (Internet Protocol), ktorá je definovaná pre CRG aplikácie. Uzol R taktiež prevezme vlastníctvo prepínateľného zariadenia určeného v CRG zariadení.

**Poznámka:** Kým je Uzol L vypnutý nie je dostupnosť systému chránená, pretože neexistuje žiaden záložný uzol, ak by došlo k zlyhaniu Uzla R. Keď sa Uzol L obnoví a pripojí ku klastru, vykoná sa záloha oboch skupín prostriedkov klastra. Ak chcete, aby znova prevzal rolu primárneho uzla, mali by ste vykonať administratívne prepnutie.

### Súvisiace koncepty

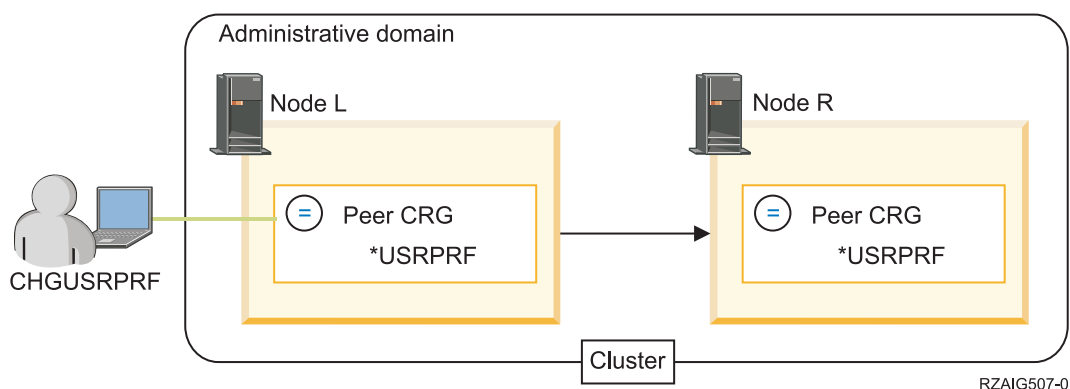
Konfigurácie nezávislých diskových oblastí

## Príklad: Administratívna doména klastra na manažovanie partnerských prostriedkov

Táto téma poskytuje príklad konfigurácie administratívnej domény klastra, používanej na monitorovanie prostriedkov v klastru.

Táto ukážka konfigurácie poskytuje nasledovné:

- Klaster s dvoma uzlami
- Administratívna doména klastra s dvoma uzlami v zozname uzlov domény
- Položka monitorovaného prostriedku (MRE) pre užívateľský profil, ktorý sa má v doméne synchronizovať.



V tomto príklade chce administrátor zabezpečiť, že užívateľský profil zostane v rámci klastra konzistentný, preto vytvorí administratívnu doménu klastra na monitorovanie a synchronizovanie zmien užívateľského profilu. Administratívnu doménu klastra reprezentuje CRG s partnerskými uzlami, obsahujúce Uzol L a Uzol R. Pre užívateľský profil sa pridá do administratívnej domény klastra položka monitorovaného prostriedku. V tomto príklade boli pri pridávaní MRE zadané všetky atribúty užívateľského profilu. To znamená, že keď sa zmení ľubovoľný atribút v Uzle L alebo Uzle R, po spustení CRG sa zmena automaticky rozšíri do aktívnych uzlov domény.

Tieto kroky opisujú, čo vykonal administrátor pri nastavovaní tohto príkladu:

1. Vytvoril klaster s Uzlami L a R.
2. Vytvoril administratívnu doménu klastra pre Uzly L a R
3. Pridal MRE na reprezentáciu užívateľského profilu
4. Spustil CRG s partnerskými uzlami, ktoré reprezentuje administratívnu doménu klastra
5. Zmenil užívateľský profil v Uzle L alebo Uzle R. Administratívna doména klastra automaticky zmení užívateľský profil v druhom uzle. Ak je zmena úspešná, globálny stav monitorovaného prostriedku bude konzistentný.

## Príklad: Nezávislé diskové oblasti s geografickým zrkadlením

Tento príklad zobrazuje jeden možný spôsob konfigurácie geografického zrkadlenia. Uzol A a Uzol B sa nachádzajú v meste New York. Uzol C a Uzol D sa nachádza v Bostone. Všetky štyri uzly sú nakonfigurované v rovnakej doméne obnovy. Produkčnú kópiu je možné prepínať medzi uzlami A a B. Zrkadlovú kópiu je možné prepínať medzi uzlami C a D. Keďže sú všetky uzly v rovnakej doméne obnovy, zdrojový systém v New Yorku si môže tiež vymeniť rolu s cieľovým systémom v Bostone, čím umožní systému v Bostone hosťovať produkčnú kópiu.



Spoločnosť definovala pre uzly v doméne obnovy tieto roly:

Uzol	Rola
Uzol A	Primárny
Uzol B	Záloha 1
Uzol C	Záloha 2
Uzol D	Záloha 3

V prípade prírodnej pohromy v New Yorku rozšíri Uzol C v Bostone svoju zrkadlovú kópiu na produkčnú kópiu a stane sa primárnym uzlom. Uzol C sa stane zdrojovým uzlom pre geografické zrkadlenie, aj keď sa geografické zrkadlenie pozastaví, nakoľko z dôvodu prírodnej pohromy v New Yorku nebude existovať žiadny cieľový uzol. Po obnove lokality v New Yorku sa Uzol A stane záložným uzlom a jeho predošlá produkčná kópia sa stane zrkadlovou kópiou.

## Riešenie problémov s klastrami

Tu nájdete riešenia pri problémoch, ktoré sú pre klastre typické.

Niekedy sa môže stať, že klaster nepracuje správne. Táto téma ponúka informácie o problémoch, ktoré sa môžu objaviť pri práci s klastrami.

## Identifikácia problémov s klastrami

Tu začnite určovať vaše klastrové problémy.

Niekedy sa môže stať, že váš klaster nepracuje správne. Ak si myslíte, že nastal problém, nasledujúci text vám pomôže zistiť, či nastal nejaký problém, a aké je povahy.

- **Zistite, či je vo vašom systéme aktívne klastrovanie.**

Ak chcete zistiť, či sú služby klastrových prostriedkov aktívne, vyhľadajte v subsystéme QSYSWRK dve úlohy - QCSTCTL a QCSTCRGM. Ak sú tieto úlohy aktívne, potom sú aktívne aj služby klastrových prostriedkov. Na zobrazenie úloh v podsystéme môžete použiť funkciu Riadenie prevádzky v Navigátore iSeries alebo príkaz WRKACTJOB (Work with Active Jobs). Na zobrazenie stavových informácií o klastru môžete použiť tiež príkaz DSPCLUINF (Display Cluster Information).

- Aktívne môžu byť aj ďalšie úlohy pre služby klastrových prostriedkov. Téma Štruktúra úloh služieb klastrových prostriedkov obsahuje informácie o formáte úloh služieb klastrových prostriedkov.

- **Vyhľadajte správy indikujúce problém.**

- Hľadajte správy dotazu v QSYSOPR, ktoré čakajú na odozvu.
- Hľadajte hlásenia chýb v QSYSOPR, ktoré poukazujú na klastrový problém. Všeobecne budú tieto hodnoty v rozsahu od CPFBB00 po CPFBBFF.
- Zobrazte protokol histórie (CL príkaz DSPLOG) pre správy, ktoré indikujú problém s klastrom. Všeobecne budú tieto hodnoty v rozsahu od CPFBB00 po CPFBBFF.

- **Vyhľadajte závažné chyby v protokoloch úloh klastra.**

Tieto úlohy sú doslova stanovené s úrovňou protokolovania (4 0 \*SECLVL) tak, že môžete vidieť všetky nevyhnutné hlásenia chýb. Skontrolujte, či tieto úlohy a úlohy ukončovacieho programu majú úroveň protokolovania vhodne stanovenú. Ak klastrovanie nie je aktívne, ešte stále môžete vyhľadať spoolové súbory pre klastrové úlohy a úlohy ukončovacieho programu.

- **Ak predpokladáte určitý druh stavu zaseknutia, pozrite si zásobníky volaní úloh klastra.**

Zistite, či sa objavil problém v niektorom z DEQW (čakanie na vyradenie z frontu). Ak áno, skontrolujte vyvolávací zásobník každého vlákna a pozrite sa, či niektorý z nich neobsahuje getSpecialMsg vo vyvolávacom zásobníku.

- **Skontrolujte položky v protokoloch zvislého licenčného interného kódu (VLIC) klastra.**

Tieto položky protokolu majú hlavný kód 4800.

- **Pomocou príkazu NETSTAT skontrolujte, či vo vašom komunikačnom prostredí nie sú anomálie.**

NETSTAT vráti informácie o stave sieťového smerovania TCP/IP, rozhraní, pripojení TCP a portov UDP na vašom systéme.

- Netstat voľba 1 (stav rozhrania Pracovať s TCP/IP) použite na kontrolu, či adresy IP pre klastrovanie sú v stave 'Aktívne'. Takisto skontrolujte, či adresa LOOPBACK (127.0.0.1) je tiež aktívna.
- Netstat voľba 3 (stav pripojenia Pracovať s TCP/IP) použite na zobrazenie čísel portov (F14). Lokálny port 5550 by mal byť v stave 'Načúva'. Tento port sa musí otvoriť cez príkaz STRTCPSVR \*INETD, zobrazený prítomnosťou úlohy QTOGINTD (Užívateľ QTCP) v zozname aktívnych úloh. Ak je na uzle spustené klastrovanie, potom musí byť otvorený lokálny port 5551 a musí byť v stave '\*UDP'. Ak klastrovanie nie je spustené, potom port 5551 nesmie byť otvorený. V opačnom prípade by to zamedzilo úspešnému spusteniu klastrovania na predmetnom uzle.

- Použite príkaz ping. Ak sa pokúsite spustiť uzol klastra a jeho dostupnosť sa nedá overiť pomocou príkazu ping, zobrazí sa interná chyba klastrovania (CPFBB46).

- **Pomocou makra CLUSTERINFO zobrazte pohľad služieb klastrových prostriedkov na uzly v klastru, uzly v rôznych skupinách prostriedkov klastra a aktuálne používané adresy IP klastra.**

Nájdené závislosti vám môžu pomôcť identifikovať problémové oblasti, ak klaster nefunguje podľa očakávaní. Detaily o používaní makra CLUSTERINFO a interpretovaní jeho výsledkov nájdete v téme "Zisťovanie problému pomocou makra CLUSTERINFO" na strane 126.

### Súvisiace koncepty

“Úloha štruktúry a užívateľské fronty” na strane 112

Pri manažovaní klastra musíte poznať štruktúry úloh a užívateľské fronty.

### Súvisiace úlohy

Zobrazenie úloh v podsystéme

### Súvisiaci odkaz

WRKACTJOB (Work with Active Jobs)

Príkaz DSPCLUINF (Display Cluster Information)

## Získanie informácií pre obnovu klastra

Na zhromaždenie informácií tvoriacich celkový obraz vášho klastra môžete použiť príkaz WRKCLU (Work with Cluster). Tieto informácie môžete použiť pri odstraňovaní chýb.

Príkaz WRKCLU (Work with Cluster) sa používa na zobrazenie uzlov a objektov klastra a na prácu s nimi. Po vykonaní tohto príkazu sa zobrazí obrazovka Práca s klastrom. Okrem zobrazenia uzlov v klastri a informácií o klastri môžete pomocou tohto príkazu získať údaje o vašom klastri.

Ak chcete získať informácie pre obnovu po chybe, vykonajte tieto kroky:

1. V znakovom rozhraní zadajte príkaz WRKCLU OPTION(VOLBA). Pomocou týchto volieb môžete určiť, s ktorými stavovými informáciami o klastri chcete pracovať.

#### \*SELECT

Zobrazenie ponuky Práca s klastrom.

#### \*NODE

Zobrazenie panelu Informácie o klastri, ktorý obsahuje zoznam uzlov v klastri.

\*CFG Zobrazenie všetkých konfiguračných parametrov pre klastre. Poskytuje tiež možnosť získať detailné informácie o skupine prostriedkov klastra.

\*CRG Zobrazenie zoznamu skupín prostriedkov klastra v klastri.

#### \*SERVICE

Získanie súvisiacich informácií zo sledovania a ladiacich informácií pre všetky úlohy služieb klastrových prostriedkov v klastri. Tieto informácie sa zapisujú do súboru s členom pre každú úlohu služieb klastrových prostriedkov. Túto voľbu použijete, len keď vás o to požiada poskytovateľ servisu. Zobrazí sa panel s výzvou pre príkaz DMPCLUTRC (Dump Cluster Trace).

## Zisťovanie problému pomocou príkazu DMPCLUTRC (Dump Cluster Trace)

Príkaz DMPCLUTRC (Dump Cluster Trace) vám môže pomôcť pri zisťovaní a riešení problémov s klastrom.

Príkaz DMPCLUTRC (Dump Cluster Trace) vám môže pomôcť zistiť, či sa dokončila úloha klastra alebo čo úloha aktuálne spracúva. Príkaz vypíše do súboru sledovanie a ladiace informácie týkajúce sa klastra. Informácie sa vypíšu lokálne v jednom alebo viacerých uzloch klastra. Pomocou príkazu môžete vykonať výpis jednej alebo všetkých úloh služieb klastrových prostriedkov (CRS). Každá vypísaná úloha CRS má v súbore svoj člen. Názov člena súboru tvorí názov úlohy CRS. Aby mohol príkaz vygenerovať výstup, klastrovanie musí byť aktívne. Výstup sa vytvorí len pre uzly, ktoré majú aktívnu úlohu CRS. Vypísané informácie pochádzajú z užívateľského sledovania, iné informácie sa získajú z objektov klastra. Množstvo vypisovaných informácií určuje úroveň výpisu. Rôzne úrovne výpisu sú: základné informácie, informácie o chybách, informačné informácie a podrobné informácie. Úroveň výpisu určuje, koľko informácií sa odošle do súboru. Vo väčšine prípadov vám predstaviteľ servisu IBM na základe vašich potrieb oznámi, akú úroveň máte zadať; úroveň LEVEL(\*ERROR) je však dostatočná pri väčšine scenárov odstraňovania problémov. Ak neviete, ktorá úroveň je vhodná vo vašej situácii, kontaktujte predstaviteľa servisu IBM.



## Interpretovanie výsledkov sledovania

Analýzou výsledkov sledovania sa môžete dozvedieť, čo vykonáva klastrovanie, napríklad ktorá úloha klastra spôsobuje čakanie protokolu. Výstup pochádzajúci z užívateľského sledovania bude obsahovať oddeľujúci riadok, pozostávajúci z postupnosti znakov rovnosti (=). Množstvo oddeľovacích riadkov v súbore bude závisieť od množstva zavolaní príkazu DMPCLUTRC. Jeden súbor môže obsahovať výstup z viacerých zavolaní príkazu DMPCLUTRC. Posledná množina výpisov zásobníkov predstavuje najaktuálnejšie informácie. V niektorých prípadoch môže mať úloha CRG dve skupiny. Každá skupina bude mať v súbore samostatnú časť výpisu.

V nasledujúcom príklade výsledkov výpisu sledovania klastra sú dva uzly (SYSTEM1 a SYSTEM2) v klastru s názvom MOJKLASTER. Používa sa jedno CRG s názvom MOJECRG. Oba uzly sú v doméne obnovy MOJECRG. Užívateľ zadal CL príkaz STRCRG a procesu trvá vrátenie výsledkov dlho. V rozhraní príkazového riadka inej pracovnej stanice zadal užívateľ príkaz DMPCLUTRC CLUSTER(MOJKLASTER) CRG(\*ALL) LEVEL(\*ERROR) FILE(MOJSUBOR).

V tomto príklade sa výstup z príkazu DMPCLUTRC nachádza v člene MOJECRG súboru MOJSUBOR. Pre jednoduchšie pochopenie obsahu bol člen MOJECRG rozdelený do dvoch častí. V častiach sú čísla zvýraznené v zátvorkách, čo identifikuje opisované informácie. Tieto detaily vám môžu pomôcť pri odstraňovaní problémov s klastrom.

**Poznámka:** Vertikálne znaky vynechania označujú, že časť sledovania bola odstránená a nie je vo výstupe zobrazená.

### 1. časť výsledkov príkazu DMPCLUTRC

User Trace Dump for job 073586/QSYS/MOJECRG. Size: 300K, Wrapped 0 times.

```
--- 08/22/2005 16:43:32 ---
(1a) 00000006:658536 Main thread handle 2
(1b) 00000008:748016 Work thread 1 handle 13
(1b) 00000007:754576 Work thread 2 handle 11
--- 08/22/2005 16:46:04 ---
00000008:269608 CSTDAMBR 1115: WaitForMsg 4 1005 CPFBB3C
--- 08/22/2005 16:48:17 ---
00000006:925112
(1c) DMPCLUTRC Node SYSTEM1 Group MOJECRG
=====
```

Prvá časť obsahuje čísla a identifikátory vlákien pre úlohu klastra. Úlohy klastra môžu mať dve alebo viac vlákien. V tomto príklade je jedno hlavné vlákno (1a), kam prichádza všetka práca a dve pracovné vlákna (1b). Táto časť obsahuje tiež informácie o tom, z ktorého systému pochádza toto sledovanie a ktorej úlohy klastra sa týka (1c).

### 2. časť výsledkov príkazu DMPCLUTRC

```
00000006:925168 Stack Dump For Target Thread: Handle 2 (0x00000002)
00000006:925192 Stack:
(2aa) Main Thread Stack MOJECRG
00000006:925256 Stack: Library / Program Module Stmt Procedure
00000006:933432 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 0 : _CXX_PEP__Fv
00000006:933488 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 46 : main
00000006:933536 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 65 : completeStartup_FP8CstDAMbr
00000006:933584 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 26 : mainQueueProcessLoop_FP8Cs
DAMbr
00000006:933616 Stack: QSYS / QCSTCMN CSTDAMBR 57 : processQueueMsgs__8CstDAMbrF
Q2_8CstDAMbr13CstQueueIndex
00000006:933664 Stack: QSYS / QCSTCMN CSTDAMBR 53 : processMsg__8CstDAMbrFP6CstM
sg
00000006:933712 Stack: QSYS / QCSTCMN CSTDAMBR 17 : callFnPtr__8CstDAMbrFPQ2_8Cs
tDAMbr19MsgFunctionPtrEntryP6
00000006:933744 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 94 : crgDump_FP6CstMsgP8CstDAMbr
00000006:933792 Stack: QSYS / QCSTCMN CSTACK 95 : CstAckQueryMsg
00000006:933832 Stack: QSYS / QP0ZCPA QP0ZDBG 3 : Qp0zDumpTargetStack
00000006:933864 Stack: QSYS / QP0ZCPA QP0ZDBG 12 : Qp0zSUDumpTargetStack
00000006:934016 Stack: Exception In Stack Dump Code
```



```

|      00000006:934040 Stack: thread is likely terminated or no longer running the same code as the
| captured stack
|      00000006:934080
| (2a) Work Thread Index 1 Group MOJECRG      Last or current values
| (2e) 00000006:934112 Request handle 8E3E1002 EE3218A1 824F0004 AC000456
| (2c) 00000006:934136 SPI name QcstStartClusterResourceGroup
|      00000006:934160 (2g) POF 10, Completed ack round 1 (2i)
|      00000006:934176 (2o) In waitForJobEnd QDFTJOB  MOJKLASTER      073590
|      00000006:934216 Node Ack Status POF (2bb) Nack Msg Id
|      00000006:934240 (2n) SYSTEM1 (2cc) Ready
|      00000006:934272 SYSTEM2 Ack 10 (2k)
|      00000006:934296 Messages
|      00000006:934320 Stack Dump For Target Thread: Handle 13 (0x0000000d)
|      00000006:934344 Stack: Work Thread 1 Stack MOJECRG
|      00000006:934792 Stack: Library / Program Module Stmt Procedure
|      00000006:934840 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 9 : workThreadRoutine_FPv
|      00000006:934888 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 28 : workQueueProcessLoop_FP8Cst
| DAMbr
|      00000006:941688 Stack: QSYS / QCSTCMN CSTDAMBR 57 : processQueueMsgs__8CstDAMbrF
| Q2_8CstDAMbr13CstQueueIndex
|      00000006:941696 Stack: QSYS / QCSTCMN CSTDAMBR 33 : processMsg__8CstDAMbrFP6CstM
| sg
|      00000006:941712 Stack: QSYS / QCSTCMN CSTDAMBR 17 : callFnPtr__8CstDAMbrFPQ2_8Cs
| tDAMbr19MsgFunctionPtrEntryP6
|      00000006:941728 Stack: QSYS / QCSTCMN CSTACK 3 : CstStripOffHeaderMsgPart
|      00000006:941736 Stack: QSYS / QCSTCMN CSTDAMBR 53 : processMsg__8CstDAMbrFP6CstM
| sg
|      00000006:941752 Stack: QSYS / QCSTCMN CSTDAMBR 17 : callFnPtr__8CstDAMbrFPQ2_8Cs
| tDAMbr19MsgFunctionPtrEntryP6
|      00000006:970888 Stack: QSYS / QCSTCRGS2 CSTCRGSS 39 : startCrg
|      00000006:970912 Stack: QSYS / QCSTCRGS2 CSTCRGSS 344 : doMessageProcessing_FP6CstM
| sgP8CstDAMbr
|      00000006:970928 Stack: QSYS / QCSTCRGS2 CSTCRGSS 57 : doExitPgmPhase_FP6CstMsgP8C
| stDAMbr
|      00000006:981984 Stack: QSYS / QCSTCMN CSTDAMBR 52 : waitForJobEnd__8CstDAMbrFPA2
| 6_ci
|      00000006:982000 Stack: QSYS / QCSTCMN CSTDAMBR 73 : waitForSpecialMsg__8CstDAMbr
| FP17CstSpecialMsgListPA8_ciT3
|      00000006:982016 Stack: QSYS / QC2UTIL1 QC2MI3 1 : (2dd) deq
|      00000006:982136 Stack: Exception In Stack Dump Code
|      00000006:982136 Stack: thread is likely terminated or no longer running the same code as the
| captured stack
|      00000006:982160
| (2b)Work Thread Index 2 Group MOJECRG      Last or current values
| (2f)00000006:982176 Request handle D9C3C8C3 E2E3F5F2 0003 0000
| (2cc)00000006:982176 SPI name
|      00000006:982184 (2h) POF 0, (2d)Completed ack (2j)round 0
|      00000006:982184 In getNextWorkMsg
|      00000006:982208 Node Ack Status POF Nack Msg Id
| (21) 00000006:982208 SYSTEM1 Ready
| (21) 00000006:982232 SYSTEM2 Ready
|      00000006:982248 Messages
|      00000006:982256 Stack Dump For Target Thread: Handle 11 (0x0000000b)
|      00000006:982256 Stack: Work Thread 2 Stack MOJECRG
|      00000006:982344 Stack: Library / Program Module Stmt Procedure
|      00000006:982360 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 9 : workThreadRoutine_FPv
|      00000006:982376 Stack: QSYS / QCSTCRGJOB CSTCRGJOB 28 : workQueueProcessLoop_FP8Cst
| DAMbr
|      00000006:982392 Stack: QSYS / QCSTCMN CSTDAMBR 51 : processQueueMsgs__8CstDAMbrF
| Q2_8CstDAMbr13CstQueueIndex
| (2m) 00000006:982400 Stack: QSYS / QCSTCMN CSTDAMBR 105 : getNextWorkMsg__8CstDAMbrFv
|      00000006:982416 Stack: QSYS / QC2UTIL1 QC2MI3 1 : deq
|      00000006:982480 Stack: Exception In Stack Dump Code
|      00000006:982480 Stack: thread is likely terminated or no longer running the same code as the
| captured stack

```

Druhá časť obsahuje zásobníky volaní pre všetky vlákna, ktoré sú súčasťou úlohy klastra. Vo väčšej časti bude hlavné vlákno obsahovať informácie o príkaze DMPCLUTRC, ktorý sa práve vykonal (2aa). Pracovné vlákna (2a a 2b) obsahujú informácie zo sledovania, ktorá vám pomôžu určiť, čo sa deje s úlohou klastra. Táto časť obsahuje detaily o zásobníku volaní, napríklad názov SPI (2c), dokončené potvrdenie (ACK) (2d), identifikátor požiadavky pre priradené rozhrania API (2e) alebo identifikátor poslednej dokončenej požiadavky (2f), aktuálny bod zlyhania (POF) (2g a 2h), aktuálny cyklus potvrdzovania (ACK) (2i a 2j) a uzly, ktoré vykonalí potvrdenie (ACK) (2k a 2l).

Aktuálny bod zlyhania (POF) je interná hodnota, ktorá reprezentuje miesto, kde sa nachádza kód aktuálneho protokolu a nemusí nutne znamenať, že došlo k zlyhaniu. Hodnota *Ack* znamená, že uzol úspešne dokončil túto časť protokolu a čaká na potvrdenie (ACK) alebo záporné potvrdenie (Nack) od všetkých ostatných uzlov. Hodnota *Nack* znamená, že uzol nemôže úspešne dokončiť túto časť protokolu a čaká na odpoveď od všetkých ostatných uzlov. ID správy pre Nack je uvedené v ďalšom stĺpci (2bb). Toto je rovnaká správa, ktorá sa odosiela do RIQ pôvodcu. Ak uzol počas protokolu zlyhá, jeho stav bude "Fail" a v závislosti od protokolu a od uzla sa môže alebo nemusí považovať za Nack. Stav Ack "Inactive" znamená, že sa uzol nezúčastnil protokolu. Hodnota "Ready" znamená, že uzol ešte neodpovedal. Keď je vlákno v stave getNextWorkMsg (2m), znamená to, že čaká na prácu, ktorú bude vykonávať.

Názvy procedúr čítajte odspodu, smerom navrch zásobníka volaní. Tento ukážkový súbor obsahuje deq (2dd) so stavom waitForSpecialMsg, waitForJobEnd a doExitPgmPhase. To znamená, že protokol pred pokračovaním v spracovaní čaká na dokončenie ukončovacieho programu. Z hodnoty Ack Status (2k) je možné určiť, na ktorý uzol čaká protokol. V tomto príklade sa čaká na uzol SYSTEM1 (2n). Úplný názov úlohy (2e) označuje úlohu, na ktorú čaká systém. Keď určíte názov úlohy, môžete s ňou pracovať a odstrániť príčinu čakania. Možnou príčinou môže byť napríklad, že úloha stále čaká vo fronte, úloha sa vykonáva ale jej spracovanie trvá dlhší čas, alebo že úloha čaká na zamknutý objekt.

V tomto príklade protokol čaká na dokončenie ukončovacieho programu. Jednoduchší spôsob určenia, či protokol čaká na dokončenie ukončovacieho programu alebo úlohy zapnutia, je skontrolovať prvú časť a zistiť, či je stav waitForJobEnd (2o). Názov úlohy, na ktorú sa čaká, je uvedený na rovnakom riadku. To eliminuje potrebu kontroly zásobníkov.

### 3. časť výsledkov príkazu DMPCLUTRC

```
5722SS1 V5R4M0 060210 AS/400 DUMP 073586/QSYS/MOJECRG 08/22/05 16:48:18 PAGE 1
DMPYSOBB PARAMETERS
(3a)OBJ- MOJECRGAIX CONTEXT- QTEMP
TYPE- *ALL SUBTYPE-*ALL
OBJECT TYPE- INDEX *CRGM
NAME- MOJECRGAIX TYPE- 0E SUBTYPE- A5
LIBRARY- QTEMP 006B8A19B00C9E807000 TYPE- 04 SUBTYPE- C1
CREATION- 08/22/05 16:43:32 SIZE- 0000007000
ATTRIBUTES- 0000 ADDRESS- C7FE286F04 000000
.
.
.
.POINTERS-
NONE
OIR DATA-
NONE
END OF DUMP

* * * * * E N D O F L I S T I N G * * * * *
```

Tretia zobrazená časť predstavuje interný objekt, ktorý obsahuje informácie o úlohe klastra. V tomto príklade je to interný index s názvom MOJECRGAIX (3a). Tieto informácie sú omnoho čitateľnejšie ako vyššie uvedená druhá časť.

### 4. časť výsledkov príkazu DMPCLUTRC

```
5722SS1 V5R4M0 060210 AS/400 DUMP 073586/QSYS/USER 08/22/05 16:48:18 PAGE 1
DMPYSOBB PARAMETERS
(4a)OBJ- MOJECRGTQ CONTEXT- QTEMP
TYPE- 0A SUBTYPE-EF
OBJECT TYPE- QUEUE *QTQ
NAME- MOJECRGTQ TYPE- 0A SUBTYPE- EF
```

```

| LIBRARY-      QTEMP      006B8A19B00C9E807000 TYPE-          04  SUBTYPE-          C1
| CREATION-     08/22/05  16:43:32          SIZE-          000002C000
| ATTRIBUTES-   0000                      ADDRESS-       CC6765CAA2 000000
| QUEUE ATTRIBUTES-
| .
| .
| .
| .POINTERS-
| 000010 SPP 1A EF MOJECRG QSYS      073586                      00002160 0000
| 000020 SPP 1A EF MOJECRG QSYS      073586                      00001540 8000
| 000030 SPP 1A EF MOJECRG QSYS      073586                      000016E0 0000
| 000040 SPP 1A EF MOJECRG QSYS      073586                      00001690 0000
| 000050 SPP 1A EF MOJECRG QSYS      073586                      000016A0 0000
| 000070 SPP 1A EF MOJECRG QSYS      073586                      00002160 0000
| OIR DATA-
| NONE
| END OF DUMP
|
| * * * * * E N D O F L I S T I N G * * * * *

```

Štvrtá zobrazená časť sa nazýva front sledovania (4a). V tomto príklade sa nazýva MOJECRGTO. Obsahuje informácie o tom, čo použil klaster na vykonanie tejto úlohy a ako každá úloha odpovedala na požiadavku.

**Poznámka:** Nie každá správa je tu úplne opísaná.

## 5. časť výsledkov príkazu DMPCLUTRC

```

| 5722SS1 V5R4M0 060210 AS/400 DUMP 073586/QSYS/MOJECRG 08/22/05 16:48:18 PAGE 1
| DMPYSO0BJ PARAMETERS
| (5a) OBJ- MOJECRG CONTEXT- QUSRSYS
| TYPE- 19 SUBTYPE-2C
| OBJECT TYPE- SPACE *CRG
| NAME- MOJECRG TYPE- 19 SUBTYPE- 2C
| LIBRARY- QUSRSYS TYPE- 04 SUBTYPE- 01
| CREATION- 08/17/05 07:16:40 SIZE- 0000002000
| OWNER- MOJKLASTER TYPE- 08 SUBTYPE- 01
| ATTRIBUTES- 0800 ADDRESS- 1EC687A1F3 000000
| SPACE ATTRIBUTES-
| .
| .
| .
| END OF DUMP
|
| * * * * * E N D O F L I S T I N G * * * * *

```

Piata časť obsahuje informácie o objekte CRG (5a).

## Zisťovanie problému pomocou makra CLUSTERINFO

Makro CLUSTERINFO zobrazuje informácie, ktoré obsahujú služby klastrových prostriedkov o uzloch, skupinách CRG a aktívnych adresách IP klastra.

Makro CLUSTERINFO vytvára snímku informácií o aktuálnom klastri. Príkaz prechádza objektmi klastrovania a vytvára v lokálnom uzle opis klastra. Makro CLUSTERINFO poskytuje letový záznamník pre rôzne objekty klastra a môže vám pomôcť pri určovaní zdroja problému v klastri. Ak chcete použiť makro CLUSTERINFO, vykonajte tieto kroky:

1. V znakovom rozhraní zadajte STRSST.
2. Prihláste sa pomocou vášho užívateľského profilu pre servisné nástroje.
3. Na obrazovke Spustenie servisného nástroja vyberte voľbu 1 (Spustiť servisný nástroj).
4. Vyberte voľbu 4 (Zobrazenie/Zmena/Výpis).
5. Vyberte voľbu 2 (Výpis na tlačiareň).
6. Vyberte voľbu 2 (Údaje licenčného interného kódu (LIC))
7. Vyberte voľbu 14 (Rozšírená analýza).

8. Zadáte hodnotu 1 pred voľbou makra CLUSTERINFO. Stlačte kláves Enter

Keď sa zobrazí makro CLUSTERINFO, pomocou voľby -H môžete zobraziť pomoc pre všetky dostupné voľby tohto makra. Tento diagram použitia opisuje každú dostupnú voľbu makra CLUSTERINFO:

Tabuľka 21. Voľby pre makro CLUSTERINFO

Voľba	Popis
-H	Zobrazí obrazovku s pomocou pre voľby
-A	Zobrazí všetky informácie
-FR	Zobrazí položky letového záznamníka
-HB	Zobrazí informácie o kontrolnom signáli
-PERF	Zobrazí počítadlá výkonu
-Q	Zobrazí stav frontu odchádzajúcich správ
-G	Zruší potlačenie zobrazovania všetkých skupín broadcast CC
-T	Zobrazí ladiace parametre
-M	Zobrazí matice pre všetky DA
-DP	Zobrazenie informácií o údajovom porte

## Interpretovane výsledkov makra CLUSTERINFO

V tomto príklade je zadaná voľba -A, aby sa vypísali všetky polia. Letové záznamníky predstavujú hlavné nástroje ladenia. Všimnite si, že pri ukončení alebo vymazaní uzla klastra sa tieto letové záznamníky vymažú. Ak chcete vykonať analýzu problému, makro CLUSTERINFO musíte spustiť pred ukončením alebo vymazaním klastra. V niektorých prípadoch sa môžu letové záznamníky pri vymazaní alebo ukončení klastra zapísať do protokolu vlog. Letové záznamníky uchovávajú rôzne udalosti, ktoré ovplyvňujú štruktúru a výkon klastra. Detailná interpretácia údajov letových záznamníkov presahuje rozsah týchto informácií.

! **Poznámka:** Vertikálne znaky vynechania označujú, že časť výsledkov bola odstránená a nie je vo výstupe zobrazená.

### 1. časť výsledkov makra CLUSTERINFO

```
DISPLAY/ALTER/DUMP CLUSTERINFO -NEW2 08/23/05 13:36:37 PAGE 1
Running macro: CLUSTERINFO -A
Use -H for command information
Cluster Name : MOJKLASTER
Local Node Name: SYSTEM1
CC/CTS Version : 5
Macro Timestamp: 08/23/05 13:36:37.079
```

Prvá časť obsahuje všeobecné informácie o klastri, napríklad názov klastra, verziu klastra a čas vytvorenia hlásenia. V tomto príklade je názov klastra MOJKLASTER a názov lokálneho uzla SYSTEM1.

### 2. časť výsledkov makra CLUSTERINFO

```
Cluster Object Addresses
CstcClusterServices Address: DBF08681C9161580
Cluster Address : FC5B04B0D4001000
Cluster Task Address : B00010000E932000
Cluster Task Q Address : DBF08681C9169A00
Clue Group Services Address: CDAB6D0339001000
CC Services Address : FC5B04B0D4008000
```

Druhá časť poskytuje ukazovatele na umiestnenia kľúčových objektov klastra.

### 3. časť výsledkov makra CLUSTERINFO

```
Message Statistics
Number of non-fragmented messages: 250
Number of fragmented messages   : 1
Number of fragments              : 7
Number of acks                   : 148
```

Tretia časť obsahuje štatistiku správ pre klaster, napríklad počet fragmentov a počet potvrdení (ack).

### 4. časť výsledkov makra CLUSTERINFO

```
Node Map
Node ID : SYSTEM1
GenesisSubnetId : 9.5.251.0
CCNode *      : FC5B04B0D4007000
CCSrvNode *   : FC5B04B0D404F000
Adapter 1    : 9.5.251.46 Primary
  Status     : 0x01 Reachable
  Line Type  : 0x09 Ethernet
Node ID : SYSTEM2
GenesisSubnetId : 9.5.251.0
CCNode *      : FC5B04B0D4060000
CCSrvNode *   : FC5B04B0D4061000
Adapter 1    : 9.5.251.47 Primary
  Status     : 0x01 Reachable
  Line Type  : 0x09 Ethernet
```

Štvrtá časť obsahuje všetky aktívne uzly klastra v mape uzlov. V tomto príklade sú uvedené dva uzly, SYSTEM1 a SYSTEM2.

### 5. časť výsledkov makra CLUSTERINFO

```
Subnet Map
Subnet ID: 127.0.0.0
CCSubnet *      : FC5B04B0D4006000
CCSrvSubnet*   : FC5B04B0D400C000
Retries        : 0
Msg Timeouts   : 0
Bad Msg Counter : 0
Failed Default Address: 0
Subnet ID: 226.5.5.5
CCSubnet *      : FC5B04B0D405B100
CCSrvSubnet*   : FC5B04B0D405C000
Retries        : 0
Msg Timeouts   : 0
Bad Msg Counter : 0
Failed Default Address: 0
```

Piata časť obsahuje zoznam všetkých objektov podsietí, ktoré sa nachádzajú v klasteri.

### 6. časť výsledkov makra CLUSTERINFO

```
Group Map
Group ID: 0x0000000000000001
Name : CTS
CCGroup *      : FC5B04B0D405FF00
CCSrvGroup *   : FC5B04B0D4064B00
Member Nodes
  SYSTEM1
  SYSTEM2
Group ID: 0x0000000000000002
Name : CTS
CCGroup *      : FC5B04B0D4055100
CCSrvGroup *   : FC5B04B0D4055200
Member Nodes
  SYSTEM1
```

SYSTEM2

.  
. .  
.

Šiesta časť obsahuje zoznam všetkých aktuálnych skupín v klastrí. Každá skupina sa nazýva skupina distribuovanej aktivity. Tieto skupiny sa používajú na komunikáciu medzi skupinami v každom aktívnom uzle klastra. Väčšina z týchto skupín je pre licenčný interný kód (LIC). Tie je možné identifikovať podľa názvu skupiny CTS a BADA. Zobrazí sa tiež skupina pre CCTL (úloha QCSTCTL v operačnom systéme), CRGM (úloha QCSTCRGM v operačnom systéme) a pre úlohu každej skupiny prostriedkov klastra (CRG). Skupiny pre úlohy CRG nebudú mať názov. Každá skupina má členské uzly. Členské uzly sú uzly, ktoré navzájom komunikujú v tejto skupine.

## 7. časť výsledkov makra CLUSTERINFO

Partition Map  
Partition Map is empty

Siedma časť obsahuje zoznam všetkých uzlov v zozname oddielov SLIC.

**Poznámka:** Toto nie je rovnaký koncept ako pre uzly s oddielmi XPF.

## 8. časť výsledkov makra CLUSTERINFO

CTS Client List  
CTS Client List is empty

Ôsma časť obsahuje zoznam všetkých registrovaných klientov klastra, napríklad údajových portov.

## 9. časť výsledkov makra CLUSTERINFO

Flight Recorder : CSTCSVFR  
Flight Recorder Address: DBF08681C9161620

Deviata časť obsahuje letový záznamník klastrových služieb (CSTCSVFR), ktorý sa uchováva v systéme až do IPL.

## 10. časť výsledkov makra CLUSTERINFO

Message Statistics  
Number of non-fragmented messages: 250  
Number of fragmented messages : 1  
Number of fragments : 7  
Number of acks : 148  
Time Stamp: 08/18/05 14:00:15.329  
Trace Point: 0x0010 CstcClusterServicesTracePtCreatedFlightRecorder  
C3D9C5C1E3C5C6D9 <CREATEFR>  
Time Stamp: 08/22/05 16:43:28.912  
Trace Point: 0x0020 CstcClusterServicesTracePtCreatedClusterObject  
D4D6D9C5E8404040 4040C5F8D3770500 <MOJKLASTER E8L...>  
1000 <.. >  
Time Stamp: 08/23/05 13:33:40.935  
Trace Point: 0x0030 CstcClusterServicesTracePtDeletedClusterObject  
D4D6D9C5E8404040 404040E2E3 <MOJKLASTER ST >  
Time Stamp: 08/23/05 13:33:41.204  
Trace Point: 0x0030 CstcClusterServicesTracePtDeletedClusterObject  
C3D4D7E3 <CMPT >  
Time Stamp: 08/23/05 13:33:55.122  
Trace Point: 0x0020 CstcClusterServicesTracePtCreatedClusterObject  
D4D6D9C5E8404040 4040FC5B04B0D400 <MOJKLASTER ...M.>  
1000 <.. >

Desiata časť obsahuje letový záznamník pre CSTCCCFR. Tento letový záznamník klastra sa uchováva v systéme, kým sa v tomto uzle neukončí klastrovanie.

## 11. časť výsledkov makra CLUSTERINFO

Flight Recorder : CSTCCLFR  
Flight Recorder Address: FC5B04B0D4001E80

```
-----  
Time Stamp: 08/23/05 13:33:54.944  
Trace Point: 0x1010 CstcClusterTracePtCreatedSubnetObject  
7F000000FC5B04B0 D4006000 <.....M.-. >  
Time Stamp: 08/23/05 13:33:55.062  
Trace Point: 0x1000 CstcClusterTracePtCreatedNodeObject  
C3E2E3D9D9C3C8C3 E2E3F5F2FC5B04B0 <CSTRSYSTEM1....>  
D4007000 <M... >  
Time Stamp: 08/23/05 13:33:55.122  
Trace Point: 0x1020 CstcClusterTracePtCreatedMCGroupObject  
0000000000000001 00000000D9C3C8C3 <.....RCHC>  
E2E3F5F2 <ST52 >  
. . .
```

Jedenásta časť obsahuje letový záznamník pre komunikáciu klastra (CSTCCLFR). Tento letový záznamník klastra sa uchováva v systéme, kým sa v tomto uzle neukončí klastrovanie.

## 12. časť výsledkov makra CLUSTERINFO

Flight Recorder : CSTCCCFR  
Flight Recorder Address: FC5B04B0D4006380

```
-----  
Time Stamp: 08/23/05 13:33:55.080  
Trace Point: 0x3000 CstcCCScamTracePtScamOpen  
FC5B04B0D400E480 0000000000000000 <....M.U.....>  
Time Stamp: 08/23/05 13:33:55.097  
Trace Point: 0x3010 CstcCCScamTracePtScamBind  
FC5B04B0D400E480 0000000000000000 <....M.U.....>  
Time Stamp: 08/23/05 13:33:55.100  
Trace Point: 0x3000 CstcCCScamTracePtScamOpen  
FC5B04B0D400E480 0000000000000000 <....M.U.....>  
D6E4E3 <OUT >  
Time Stamp: 08/23/05 13:33:55.100  
Trace Point: 0x3010 CstcCCScamTracePtScamBind  
FC5B04B0D400E480 0000000000000000 <....M.U.....>  
. . .
```

Dvanásta časť obsahuje letový záznamník kľúčových udalostí (CSTCCCFR), ktorý sa uchováva v systéme, kým sa v tomto uzle neukončí klastrovanie.

## 13. časť výsledkov makra CLUSTERINFO

```
Time Stamp: 08/23/05 13:33:55.201  
C3A2A385C7E27A7A C3A2A385C7E24082 <CsteGS::CsteGS b>  
85878995A2 <egins >  
Time Stamp: 08/23/05 13:33:55.201  
C3A2A385C4C14083 9695A2A399A483A3 <CsteDA construct>  
85847A40C2C1C4C1 404040404040 <ed: BADA >  
Time Stamp: 08/23/05 13:33:55.201  
C3A2A385C7E27A7A C3A2A385C7E24081 <CsteGS::CsteGS a>  
8484408281848140 A39640C4C16D9389 <dd bada to DA_li>  
A2A3 <st >  
. . .
```

Trinásta časť zobrazuje obsah frontov pre odosielanie a aktívnych frontov správ. Ak táto časť nie je dlhšiu dobu prázdna, indikuje to problém v klastri.



## 14. časť výsledkov makra CLUSTERINFO

Flight Recorder : CSTECLF2  
Flight Recorder Address: CDAB6D0339001300

```
-----  
Time Stamp: 08/23/05 13:33:55.201  
C3A2A385C4C17A7A C3A2A385C4C16B40 <CsteDA::CsteDA, >  
83998581A385D4C3 C79996A49740C9C4 <createMCGroup ID>  
407E40F0A7F1F5 < = 0x15 >  
Time Stamp: 08/23/05 13:33:55.209  
C3A2A385E2C3D985 977A7A84859389A5 <CsteSCRep::deliv>  
85994094A287E3A8 97857EF0A7F140A2 <er msgType=0x1 s>  
A482E3A897857EF0 A7F240C4C17EC2C1 <ubType=0x2 DA=BA>  
C4C1404040404040 <DA >  
Time Stamp: 08/23/05 13:33:55.209  
C3A2A385C4C17A7A A58985A66B409985 <CsteDA::view, re>  
9496A585D4C3C799 96A497D485948285 <moveMCGroupMembe>  
99A240C9C4407E40 F0A7F1F540969384 <rs ID = 0x15 old>  
6D959684856D9389 A2A340A289A98540 <_node_list size >  
7E40F0A7F0 <= 0x0 >  
. . .
```

Štrnásť časť obsahuje informácie o letovom záznamníku.

## 15. časť výsledkov makra CLUSTERINFO

Message Queues

Send Queues:

```
Send Queue: 00 Messages: 00 MessageQueue*: FC5B04B0D400BF80  
Send Queue: 01 Messages: 00 MessageQueue*: FC5B04B0D400DF80  
Send Queue: 02 Messages: 00 MessageQueue*: FC5B04B0D400E600  
Send Queue: 03 Messages: 00 MessageQueue*: FC5B04B0D400E680  
Send Queue: 04 Messages: 00 MessageQueue*: FC5B04B0D400E700  
Send Queue: 05 Messages: 00 MessageQueue*: FC5B04B0D400E780  
Send Queue: 06 Messages: 00 MessageQueue*: FC5B04B0D400E800  
Send Queue: 07 Messages: 00 MessageQueue*: FC5B04B0D400E880  
Send Queue: 08 Messages: 00 MessageQueue*: FC5B04B0D400E900  
Send Queue: 09 Messages: 00 MessageQueue*: FC5B04B0D400E980  
Send Queue: 10 Messages: 00 MessageQueue*: FC5B04B0D400EA00  
Send Queue: 11 Messages: 00 MessageQueue*: FC5B04B0D400EA80  
Send Queue: 12 Messages: 00 MessageQueue*: FC5B04B0D400EB00  
Send Queue: 13 Messages: 00 MessageQueue*: FC5B04B0D400EB80  
Send Queue: 14 Messages: 00 MessageQueue*: FC5B04B0D400EC00  
Send Queue: 15 Messages: 00 MessageQueue*: FC5B04B0D400EC80  
Send Queue: 16 Messages: 00 MessageQueue*: FC5B04B0D400ED00  
Send Queue: 17 Messages: 00 MessageQueue*: FC5B04B0D400ED80  
Send Queue: 18 Messages: 00 MessageQueue*: FC5B04B0D400EE00  
Send Queue: 19 Messages: 00 MessageQueue*: FC5B04B0D400EE80
```

Active Message Queues:

```
Active Message Queue: 00 Messages: 00 MessageQueue*: FC5B04B0D4008570  
Active Message Queue: 01 Messages: 00 MessageQueue*: FC5B04B0D4008640  
Active Message Queue: 02 Messages: 00 MessageQueue*: FC5B04B0D4008710  
Active Message Queue: 03 Messages: 00 MessageQueue*: FC5B04B0D40087E0  
Active Message Queue: 04 Messages: 00 MessageQueue*: FC5B04B0D40088B0  
Active Message Queue: 05 Messages: 00 MessageQueue*: FC5B04B0D4008980  
Active Message Queue: 06 Messages: 00 MessageQueue*: FC5B04B0D4008A50  
Active Message Queue: 07 Messages: 00 MessageQueue*: FC5B04B0D4008B20  
Active Message Queue: 08 Messages: 00 MessageQueue*: FC5B04B0D4008BF0  
Active Message Queue: 09 Messages: 00 MessageQueue*: FC5B04B0D4008CC0  
Active Message Queue: 10 Messages: 00 MessageQueue*: FC5B04B0D4008D90  
Active Message Queue: 11 Messages: 00 MessageQueue*: FC5B04B0D4008E60  
Active Message Queue: 12 Messages: 00 MessageQueue*: FC5B04B0D4008F30  
Active Message Queue: 13 Messages: 00 MessageQueue*: FC5B04B0D4009000  
Active Message Queue: 14 Messages: 00 MessageQueue*: FC5B04B0D40090D0  
Active Message Queue: 15 Messages: 00 MessageQueue*: FC5B04B0D40091A0
```

```
Active Message Queue: 16 Messages: 00 MessageQueue*: FC5B04B0D4009270
Active Message Queue: 17 Messages: 00 MessageQueue*: FC5B04B0D4009340
Active Message Queue: 18 Messages: 00 MessageQueue*: FC5B04B0D4009410
Active Message Queue: 19 Messages: 00 MessageQueue*: FC5B04B0D40094E0
```

#### Tuning Values

```
cstcRcvSendTimerRatio      : 2 Default: 2
cstcMcastRelayTimerRatio   : 8 Default: 8
cstcMcastRelayHBTTimerRatio : 4 Default: 4
cstcHeartbeatBaseTimer     : 12288000000 Default: 12288000000
cstcHeartbeatBasePrecision : 4096000000 Default: 4096000000
cstcRetryPrecision         : 4096000000 Default: 4096000000
cstcRetryTimerVal         : 4096000000 Default: 4096000000
cstcCDATBaseTimer          : 491520000000 Default: 491520000000
cstcCDATBasePrecision      : 40960000000 Default: 40960000000
cstcRecoveryBaseTimer      : 3686400000000 Default: 3686400000000
cstcRecoveryBasePrecision  : 491520000000 Default: 491520000000
cstcMaxRetryTime           : 32768000000 Default: 32768000000
cstcCCFragmentSize        : 1464 Default: 1464
cstcCCSendQOverflow        : 1024 Default: 1024
cstcBadMsgCtrThreshold     : 3 Default: 3
cstcUnreachableHBBackThreshold : 1 Default: 1
cstcReachableHBBackThreshold : 3 Default: 3
cstcUnreachableHBThreshold : 4 Default: 4
cstcReachableHBThreshold   : 4 Default: 4
cstcMaxHBThreshold         : 16 Default: 16
cstcDisableMsgTimer        : 0 Default: 0
cstcRepeatAckThreshold     : 10 Default: 10
DISPLAY/ALTER/DUMP        CLUSTERINFO -NEW2 08/23/05 13:36:37 PAGE 87
cstcDelayedAckTimer        : 4096000000 Default: 4096000000
cstcDelayedAckPrecision    : 40960000 Default: 40960000
cstcCCSendWindow           : 2 Default: 2
cstcCCEnableMcast          : 1 Default: 1
cstcCCPerfClass             : 2 Default: 2
```

\*\*\*\*\* END OF DUMP \*\*\*\*\*

Pätْنَا čast obsahuje ladiace hodnoty. Ladiace hodnoty sú ekvivalentné informáciám o výkone a konfigurácii klastra, opísaným v téme API QcstRetrieveCRSInfo (Retrieve Cluster Resource Services Information). Štrnásť čast obsahuje aktuálne a predvolené hodnoty týchto polí.

## Bežné problémy s klastrom

Táto téma uvádza niektoré bežné problémy, ku ktorým môže dôjsť v klastri, ako aj spôsoby, ako sa im vyhnúť alebo vykonať obnovu.

Nižšie uvedeným problémom sa dá jednoducho vyhnúť alebo môžu byť ľahko napravené.

### Nemôžete spustiť alebo reštartovať uzol klastra

Táto situácia obvyčajne nastane pri niektorých problémoch s vaším komunikačným prostredím. Aby ste jej zabránili, uistite sa, či sú vaše sieťové atribúty správne nastavené, vrátane adresy spätnej slučky, nastavení INETD, atribútu ALWADDCLU a IP adres pre klastrovú komunikáciu.

- Ak sa pokúšate spustiť vzdialený uzol, musia byť na cieľovom uzle vhodne nastavené sieťové atribúty ALWADDCLU. Atribút môže byť nastavený buď na \*ANY alebo \*RQSAUT v závislosti od vášho prostredia.
- Vybraté adresy IP pre klastrovanie v lokálnom a cieľovom uzle musia mať stav *Aktívna*.
- Takisto musí byť lokálne a na cieľovom uzle aktívna adresa spätnej slučky LOOPBACK (127.0.0.1).
- Lokálne a akékoľvek vzdialené uzly musia byť kvôli zabezpečeniu aktívneho smerovania siete testované na odozvu za pomoci adresy IP, použitých na klastrovanie.
- INETD musí byť na cieľovom uzle aktívne. Keď je INETD aktívne, port 5550 v cieľovom uzle by mal mať stav *Počúvanie*. Informácie o spúšťaní servera INETD nájdete v téme Server INETD.

- Hlavne kvôli pokusu o spustenie uzla nesmie byť port 5551 otvorený. V opačnom prípade by sa nepodarilo úspešne spustiť klastrovanie.

## Máte niekoľko rozpojených klastrov obsahujúcich jeden uzol

Táto situácia môže nastať, ak spustený uzol nemôže komunikovať s ostatnými klastrovými uzlami. Vtedy skontrolujte komunikačné cesty.

## Odpoveď užívateľských programov je pomalá

Bežnou príčinou tejto situácie je nesprávne nastavenie pre popis úlohy, ktorý používa ukončovaci program. Parameter MAXACT je možno nastavený na priveľmi nízku hodnotu, takže napríklad môže byť kedykoľvek aktívna inštalácia ukončovacieho programu. Odporúča sa nastavenie na \*NOMAX.

## Celkový výkon sa zdá byť nízky

Existuje niekoľko typických príčin.

- Najpravdepodobnejšou príčinou je silné zaťaženie zdieľaných komunikačných liniek.
- Ďalšou príčinou môže byť nekonzistencia medzi komunikačným prostredím a laditeľnými parametrami klastrového posielania správ. Na zobrazenie aktuálnych nastavení ladiacich parametrov môžete použiť API QcstRetrieveCRSInfo (Retrieve Cluster Resource Services Information) a na ich zmenu môžete použiť API QcstChgClusterResourceServices (Change Cluster Resource Services). Výkon klastra môže byť znížený kvôli nastaveniam predvolených laditeľných parametrov klastra, ak je použitý starý adaptérový hardvér. Ako *staré* sú definované tieto typy hardvéru adaptérov: 2617, 2618, 2619, 2626 a 2665. V tomto prípade je vhodné nastaviť ladiaci parameter *Trieda výkonu* na hodnotu *Normálna*.
- Ďalšou bežnou príčinou tohto stavu je problém so skupinami podpory rozosielania paketov IP. Ak sa pre niekoľko uzlov premiestnia primárne klastrové adresy na bežný LAN (prvé adresy zadané pre daný uzol pri vytváraní klastra alebo pri pridávaní uzla), klaster použije spôsobilosť podpory rozosielania paketov IP. Pomocou príkazu NETSTAT skontrolujte, že primárna adresa klastra je nastavená na skupinu hostiteľov multicast 226.5.5.5. Môžete to zobraziť pomocou voľby 14 - *Zobrazíť skupinu multicast* pre požadovanú adresu. Ak skupina multicast neexistuje, pomocou API QcstRetrieveCRSInfo (Retrieve Cluster Resource Services Information) skontrolujte, že je ladiaci parameter klastra *Povolíť multicast* stále nastavený na hodnotu TRUE.
- Ak sú všetky uzly klastra v lokálnej sieti alebo majú schopnosť smerovania, ktoré môžu v sieťových trasách spracúvať pakety s maximálnou prenosovou veľkosťou (MTU) väčšou než 1464 bajtov, prenosi veľkých správ klastra (väčších než 1536 kilobajtov) môžete výrazne zrýchliť, ak nastavíte ladiaci parameter klastra *Veľkosť fragmentov správ* tak, aby lepšie zodpovedal hodnotám MTU trasy.

## Nemôžete používať žiadnu z funkcií nového vydania

Ak sa pri pokuse o použitie funkcie nového vydania zobrazí chybová správa CPFBB70, vaša aktuálna verzia klastra je stále nastavená na predošlú úroveň verzie. Musíte aktualizovať všetky klastrové uzly na úroveň nového vydania a potom použiť rozhranie prispôsobenia klastrovej verzie pre nastavenie aktuálnej klastrovej verzie na novú úroveň. Viac informácií nájdete v téme Nastavenie verzie klastra.

## Nemôžete pridať uzol do domény zariadení alebo nemôžete používať rozhranie Navigátora iSeries na manažovanie klastrov

Ak chcete používať rozhranie Navigátora iSeries na manažovanie klastrov alebo ak chcete používať prepínateľné zariadenia, musíte mať vo vašom systéme nainštalovanú voľbu 41 systému i5/OS, Vysokodostupné prepínateľné zariadenia. Takisto musíte mať pre túto voľbu platný licenčný kľúč.

## Aplikovali ste PTF klastra, ale nezdá sa, že funguje

1. Po aplikovaní PTF musíte vykonať tieto úlohy:
  1. Ukončiť klaster

## 2. Odhlásiť sa a znova sa prihlásiť

Starý program je v aktivačnej skupine aktívny dovtedy, kým nie je táto skupina zrušená. Všetky klastrové kódy (aj klastrové API) sú spustené v predvolenej aktivačnej skupine.

## 3. Spustíť klaster

Väčšina opráv PTF pre klaster vyžaduje na aktiváciu ukončenie a opakované spustenie klastrovania v uzle.

## V protokole úlohy pre ukončovací program sa nachádza správa CEE0200

V tomto hlásení o chybe je formovací modul QLEPM a formovacia procedúra Q\_LE\_leBdyPeilog. Každý program, ktorý je vyvolaný ukončovacím programom, musí byť spustený buď v \*CALLER alebo v pomenovanej aktivačnej skupine. Musíte zmeniť svoj aktivačný program alebo program s chybou, aby ste tento stav napravili.

## V protokole úlohy pre služby klastrových prostriedkov sa nachádza správa CPD000D a po nej správa CPF0001

Ak sa zobrazí táto chybová správa, skontrolujte, že je systémová hodnota QMLTTHDACN nastavená na 1 alebo 2.

## Zdá sa, že sa klaster zasekol

Uistite sa, že ukončovacie programy skupiny klastrových prostriedkov sú nevybavené. Ak chcete skontrolovať ukončovací program, použite príkaz WRKACTJOB (Work with Active Jobs) a overte, či je v stĺpci Funkcia uvedené PGM-QCSTCRGEXT.

### Súvisiace koncepty

“Povolenie pridania uzla do klastra” na strane 88

Skôr ako budete môcť pridať uzol do klastra, musíte nastaviť hodnotu pre sieťový atribút ALWADDCLU (Allow add to cluster).

“Výkon klastra” na strane 110

Keď sú vykonané zmeny na klastri, môže to ovplyvniť réžiu spravovania klastra.

“Verzia klastra” na strane 13

*Verzia klastra* reprezentuje úroveň funkcií dostupných v klastri.

“Manažovanie klastrov v Navigátore iSeries” na strane 72

IBM ponúka rozhranie na manažovanie klastrov, ktoré je dostupné v Navigátore iSeries a ktoré poskytuje voľba 41 (i5/OS - Vysokodostupné prepínateľné prostriedky).

### Súvisiace úlohy

“Prispôsobenie klastrovej verzie pre klaster” na strane 101

Verzia klastra definuje úroveň, na ktorej navzájom aktívne komunikujú všetky uzly klastra.

## Chyby oddielov

Určité klastrové stavy je možné jednoducho opraviť. Ak došlo k rozkladu klastra, dozviete sa tu, ako vykonať obnovu. Tiež sa dozviete, ako môžete zabrániť deleniu klastra a uvedie príklady, ako oddiely spojiť naspäť.

K rozkladu klastra dochádza v vždy, keď sa preruší kontakt s jedným alebo viacerými uzlami klastra a zlyhanie stratených uzlov nie je možné potvrdiť. Nemali by ste si to zamieňať s delením v prostredí logických oddielov (LPAR).

Ak do protokolu histórie (QHST), alebo protokolu úlohy QCSTCTL dostanete chybovú správu CPFBB20, objavilo sa rozdelenie klastra a vy by ste mali vedieť, ako ho obnoviť. Nasledujúci príklad popisuje rozdelenie klastra, ktorý obsahuje vytvorený zo štyroch uzlov: A, B, C a D. Príklad ukazuje stratu spojenia, ktorá sa objavila medzi uzlami B a C, čo spôsobilo, že sa klaster rozdelil na dva oddiely. Pred rozkladom klastra existovali štyri skupiny prostriedkov klastra ľubovoľného typu s názvom CRG A, CRG B, CRG C a CRG D. Príklad ukazuje doménu obnovy každej skupiny prostriedkov klastra.

Tabuľka 22. Príklad domény obnovy počas rozkladu klastra

Uzol A	Uzol B	x	Uzol C	Uzol D
CRG A (záloha1)	CRG A (primárna)			
	CRG B (primárna)		CRG B (záloha1)	
	CRG C (primárna)		CRG C (záloha1)	CRG C (záloha2)
CRG D (záloha2)	CRG D (primárna)		CRG D (záloha1)	
<b>Oddiel 1</b>			<b>Oddiel 2</b>	

Klaster sa môže rozdeliť, ak je maximálna jednotka prenosu (MTU) v akomkoľvek bode komunikačnej cesty nižšia, než parameter ladiaci komunikáciu klastra, veľkosť fragmentu správy. MTU pre adresu IP klastra môžete v danom uzle skontrolovať pomocou príkazu WRKTCPS (Work with TCP/IP Network Status). MTU musí byť tiež overené na každom kroku pozdĺž celej komunikačnej cesty. Ak je MTU nižšia, než veľkosť fragmentu správy, zvýšte MTU cesty, alebo veľkosť fragmentu správy znížte. Na zobrazenie aktuálnych nastavení ladiacich parametrov môžete použiť API QcstRetrieveCRSInfo (Retrieve Cluster Resource Services Information) a na ich zmenu môžete použiť API QcstChgClusterResourceServices (Change Cluster Resource Services).

Ak bola odstránená príčina rozdelenia klastra, nájde klaster znovuvytvorené spojenie a odošle do protokolu histórie (QHST), alebo do protokolu úlohy QCSTCTL, správu CPFBB21. Tá informuje operátora, že bol klaster obnovený. Nezabúdajte, že akonáhle je dôvod rozdelenia klastra opravený, môže trvať niekoľko minút, kým sa klaster zasa spojí.

#### Súvisiace koncepty

“Oddiel klastra” na strane 27

*Oddiel klastra* je podskupina aktívnych uzlov klastra, ktorá vzniká následkom zlyhania spojenia. Členovia podskupiny si medzi sebou udržiavajú vzájomnú pripojiteľnosť.

“Zabránenie rozdeleniu klastra” na strane 83

Typickému rozkladu klastra súvisiacemu so sieťou môžete najlepšie predísť tak, že medzi všetkými uzlami klastra nakonfigurujete redundantné komunikačné cesty.

“Spájanie” na strane 22

Operácia *zlúčenia* je podobná operácii opakovaného pripojenia s tým rozdielom, že k nej dochádza, keď rozdelené uzly začnú znova komunikovať.

“Príklad: Zlyhanie” na strane 18

Zotavenie po zlyhaní zvyčajne vyplýva zo zlyhania uzla, ale sú aj iné príčiny, ktoré ho spôsobujú.

#### Určenie primárneho a sekundárneho oddielu

| Ak chcete zistiť, ktoré typy akcií pre skupiny prostriedkov klastra môžete vykonať v oddiele klastra, musíte vedieť, či sa jedná o primárny alebo sekundárny oddiel klastra. Keď sa zistí rozklad, pre každú skupinu prostriedkov klastra definovanú v klastri sa každý oddiel nastaví ako primárny alebo sekundárny.

| V prípade modelu s primárnymi/záložnými uzlami obsahuje primárny oddiel uzol, ktorý má aktuálne rolu primárneho uzla. Všetky ostatné oddiely sú sekundárne. Primárny oddiel nemusí byť pre všetky skupiny prostriedkov klastra rovnaký.

| V modeli s partnerskými uzlami platia pre oddiely tieto pravidlá:

- | • Ak sa všetky uzly domény obnovy nachádzajú v jednom oddiele, ten bude primárnym oddielom.
- | • Ak uzly domény obnovy presahujú jeden oddiel, žiaden oddiel nebude primárny. Oba oddiely budú sekundárne.
- | • Ak je skupina prostriedkov klastra aktívna a v danom oddiele sa nenachádzajú žiadne partnerské uzly, skupina prostriedkov klastra sa v tomto oddiele ukončí.
- | • V sekundárnom oddiele sú povolené prevádzkové zmeny za predpokladu, že sú splnené obmedzenia pre prevádzkové zmeny.
- | • V sekundárnom oddiele nie sú povolené žiadne zmeny konfigurácie.

Toto sú obmedzenia pre každé API Skupiny klastrových prostriedkov:

Tabuľka 23. Obmedzenia oddielu pre API skupiny prostriedkov klastra

Skupina klastrových prostriedkov API	Povolené v primárnom oddieli	Povolené v sekundárnom oddieli
Pridať uzol do obnovovacej domény	X	
Pridanie položky zariadenia CRG		
Zmeniť skupinu klastrových prostriedkov	X	
Zmena položky zariadenia CRG	X	X
Vytvoriť skupinu klastrových prostriedkov		
Vymazať skupinu klastrových prostriedkov	X	X
Rozmiestniť informácie	X	X
Ukončenie skupiny prostriedkov klastra <sup>1</sup>	X	
Inicializácia prepnutia	X	
Vypísať skupiny klastrových prostriedkov	X	X
Vypísanie informácií o skupine prostriedkov klastra	X	X
Odstránenie uzla z domény obnovy	X	
Odstránenie položky zariadenia CRG	X	
Spustenie skupiny prostriedkov klastra <sup>1</sup>	X	
<b>Poznámka:</b>		
1. Povolené vo všetkých oddieloch pre skupiny prostriedkov klastra s partnerskými uzlami, ale ovplyvňuje len oddiel, kde sa vykonáva API.		

Ak sú splnené tieto obmedzenia, po ukončení rozkladu klastra je možné zosynchronizovať skupiny prostriedkov klastra. Ako sa uzly zo stavu rozdelenia znova pripájajú ku klastru, je verzia skupiny klastrových prostriedkov okopírovaná z primárneho oddielu na uzly zo sekundárnych oddielov.

- | Keď zlučujete dva sekundárne oddiely pre model s partnerskými uzlami, za víťaza sa prehlási oddiel, ktorý obsahuje skupinu prostriedkov klastra v aktívnom stave. Ak majú oba oddiely rovnaký stav pre skupinu prostriedkov klastra, za víťaza sa prehlási oddiel, ktorý obsahuje prvý uzol uvedený v doméne obnovy skupiny prostriedkov klastra. Verzia skupiny prostriedkov klastra z víťazného oddielu sa skopíruje do uzlov v druhom oddiele.

Keď sa zistí rozklad, v žiadnom z oddielov nie možné vykonať API Add Cluster Node Entry, Adjust Cluster Version a Create Cluster. API Add Device Domain Entry je možné vykonať, len keď nie sú rozdelené žiadne z uzlov domény obnovy. Všetky ostatné API na riadenie klastrov môžu byť spustené v ktoromkoľvek oddieli. Výsledok spustenia API sa však prejaví len v oddieli, na ktorom bolo spustené.

## Zmena oddielových uzlov na zlyhané

V niektorých prípadoch je zobrazená podmienka rozdelenia pri skutočnom výpadku uzla. Táto situácia nastane vtedy, keď služby klastrových prostriedkov stratia komunikáciu v rámci jedného alebo medzi viacerými uzlami a nedá sa zistiť ich funkčnosť. V takomto prípade je tu pre vás jednoduchý mechanizmus zobrazenia, či uzol zlyhal.

**Upozornenie:** Ak službe klastrových prostriedkov oznámite zlyhanie uzla, obnovenie zo stavu oddelenia bude jednoduchšie oddelenia. Na druhej strane, ak sa vyskytlo toto oddelenie a daný uzol je ešte stále aktívny, nemal by sa jeho stav pri oddelení zmeniť na zlyhaný. Ak to vykonáte, rolu primárneho uzla skupiny prostriedkov klastra môže prijať uzol vo viac než jednom oddiele. Keď si dva uzly myslia, že sú primárne, údaje ako súbory alebo databázy sa môžu rozpojiť alebo poškodiť, ak bude viacero uzlov nezávisle vykonávať zmeny svojich kópií súborov. Okrem toho, ak mal uzol v každom oddiele priradenú rolu primárneho uzla, dva oddiely nie je možné znova zlúčiť.



Úloha uzlov v obnovovacej doméne pre každú skupinu klastrových prostriedkov v oddiele môže byť znova usporiadaná pri zmene stavu uzla na Zlyhaný. Uzol nastavený ako Zlyhaný bude priradený ako posledná záloha. Pri zlyhaní viacnásobných uzlov a zmene ich stavu ovplyvní poradie zmien týchto uzlov konečné poradie zálohových uzlov v obnovovacej doméne. Ak zlyhaný uzol bol predtým primárnym uzlom pre CRG, potom prvý aktívny zálohový uzol bude prepísaný na nový primárny uzol.

#### Súvisiace koncepty

“Spájanie” na strane 22

Operácia *zlúčenia* je podobná operácii opakovaného pripojenia s tým rozdielom, že k nej dochádza, keď rozdelené uzly začnú znova komunikovať.

“Opätovné pripojenie” na strane 20

*Opätovné pripojenie* znamená, že uzol sa stane aktívnym členom klastra po tom, čo bol nezúčastneným členom.

#### Súvisiace úlohy

“Tipy: Klastrové oddiely”

Tieto tipy použijete v prípade rozkladu klastra.

#### Súvisiaci odkaz

Príkaz CHGCLUNODE

API QcstChangeClusterNodeEntry (Change Cluster Node Entry)

Príkaz STRCLUNOD

API QcstStartClusterNode (Start Cluster Node)

#### Použitie Navigátora iSeries:

Toto vyžaduje nainštalovanú a licencovanú voľbu 41 (i5/OS - Vysokodostupné prepínateľné prostriedky).

Keď sa preruší komunikácia služieb klastrových prostriedkov s uzlom a služby nemôžu zistiť, či je uzol stále funkčný, uzol klastra bude mať v kontajneri Uzly v Navigátore iSeries stav **Nekomunikuje**. Možno bude potrebné zmeniť stav uzla z **Nekomunikuje** na **Zlyhaný**. Až potom môžete uzol reštartovať.

Ak chcete zmeniť stav uzla z **Nekomunikuje** na **Zlyhaný**, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Rozviňte klaster obsahujúci uzol, ktorého stav chcete zmeniť.
4. Kliknite na **Uzly**.
5. Pravým tlačidlom myši kliknite na uzol, ktorého stav chcete zmeniť a vyberte voľbu **Klaster** → **Zmeniť stav**.

a vyberte voľbu Klaster Zmeniť stav.

Na reštartovanie uzla použijete nasledovné kroky:

1. Pravým tlačidlom kliknite na uzol a vyberte voľbu **Klaster** → **Spustiť**.

#### Použitie príkazov CL a rozhraní API:

Ak chcete zmeniť stav uzla z **Nekomunikuje** na **Zlyhaný**, vykonajte tieto kroky:

1. Na zmenu stavu uzla z Rozdelený na Zlyhaný môžete použiť príkaz CHGCLUNODE alebo API QcstChangeClusterNodeEntry (Change Cluster Node Entry). Takto môžete zmeniť stav všetkých zlyhaných uzlov.
2. Na spustenie uzla klastra, čím sa umožní jeho opakované pripojenie ku klastru, môžete použiť príkaz STRCLUNOD alebo API QcstStartClusterNode (Start Cluster Node).

#### Tipy: Klastrové oddiely

Tieto tipy použijete v prípade rozkladu klastra.



1. Pravidlá pre obmedzenie operácií v rámci oddielu sú vytvorené tak, aby vytvorili pevné spojenie oddielov. Bez týchto obmedzení by opakované zostavenie klastra vyžadovalo množstvo práce.
2. V prípade, že sú zničené uzly v primárnom oddiele, môže sa stať, že bude nevyhnutné ich špeciálne spracovanie. Najbežnejší scenár, ktorý spôsobuje tento stav, je poškodenie lokality, ktorá tvorila primárny oddiel. Použite príklad z časti o obnove po chybách oddielu a predpokladajte zničenie Oddielu 1. V tomto prípade musí byť primárny uzol pre skupiny klastrových prostriedkov B, C, a D umiestnený v Oddiele 2. Najjednoduchším spôsobom obnovy je zmena zadanej hodnoty klastrového uzla, ktorá nastaví obidva uzly A a B na zlyhané. Viac informácií o tejto operácii nájdete v časti Zmena rozložených uzlov na zlyhané. Obnoviť uzol môžete aj manuálne, týmto spôsobom:
  - a. Z klastra odstránite uzly A a B v Oddiele 2. Tento oddiel teraz plní funkciu klastra.
  - b. Vytvorte všetky prostredia logickej replikácie, potrebné v novom klastru. Napr. príkazom CL alebo API Start Cluster Resource Group a podobne.

V takomto prípade zlyhá pokus o spojenie Oddielu 1 a 2, pretože boli z klastrovej definície v Oddiele 2 odstránené uzly. Ak chcete opraviť nezhodu medzi definíciami klastra, v každom uzle Oddielu 1 vykonajte API QcstDeleteCluster (Delete Cluster). Potom pridajte uzly z Oddielu 1 do klastra a znova vytvorte všetky definície skupín prostriedkov klastra, domény obnovy a logickú replikáciu. Táto úprava vyžaduje obrovské množstvo práce a navyše je náchylná na chyby. Veľmi dôležité je robiť úpravu len počas straty lokality.

3. Spustenie uzla závisí od jeho stavu:
 

Uzol buď zlyhal sám, alebo ho ukončila operácia Ukončiť uzol:

  - a. Služby klastrových prostriedkov sú spustené na pridávanom uzle
  - b. Klastrová definícia sa z aktívneho uzla v klastru skopíruje do spúšťaného uzla
  - c. Všetky skupiny klastrových prostriedkov, obsahujúce spúšťaný uzol vo svojej obnovovacej doméne, sú z aktívneho uzla skopírované do spúšťaného uzla. Žiadne skupiny klastrových prostriedkov nie sú v rámci klastra skopírované zo spúšťaného uzla do aktívneho uzla.

Ak ide o oddelený uzol:

- a. Klastrová definícia aktívneho uzla sa porovnáva s klastrovou definíciou spúšťaného uzla. Ak sú definície rovnaké, spustenie bude pokračovať ako spájacia operácia. Ak sa definície nezhodujú, spojenie sa zastaví a bude musieť zasiahnuť užívateľ.
- b. Ak spojenie pokračuje, spúšťaný uzol sa nastaví na aktívny.
- c. Akákoľvek skupina klastrových prostriedkov, obsahujúca spúšťaný uzol v obnovovacej doméne, je z primárneho oddielu skupiny kopírovaná do sekundárneho oddielu tej istej skupiny. Skupiny klastrových prostriedkov sa môžu kopírovať zo spúšťaného uzla do už aktívnych uzlov v klastru.

#### Súvisiace úlohy

“Zmena oddielových uzlov na zlyhané” na strane 136

V niektorých prípadoch je zobrazená podmienka rozdelenia pri skutočnom výpadku uzla. Táto situácia nastane vtedy, keď služby klastrových prostriedkov stratia komunikáciu v rámci jedného alebo medzi viacerými uzlami a nedá sa zistiť ich funkčnosť. V takomto prípade je tu pre vás jednoduchý mechanizmus zobrazenia, či uzol zlyhal.

#### Súvisiaci odkaz

API QcstDeleteCluster (Delete Cluster)

## Obnova klastra

Dozviete sa, čo podniknúť v prípade ďalších klastrových zlyhaní, ktoré sa môžu vyskytnúť.

### Obnova po zlyhaní úloh klastra

Zlyhanie úlohy služby klastrových prostriedkov zvyčajne naznačuje iný problém.

Mali by ste si prezrieť protokol úlohy, ktorá zlyhala a hľadať správy ktoré by popisovali, prečo došlo k zlyhaniu. Opravte akékoľvek chybové situácie.

- | Ak chcete opakovane spustiť ukončenú úlohu skupiny prostriedkov klastra bez toho, aby ste ukončili a znova spustili klastrovanie v uzle, môžete použiť príkaz CHGCLURCY (Change Cluster Recovery).

1. CHGCLURCY CLUSTER(PRIKLAD)CRG(CRG1)NODE(UZOL1)ACTION(\*STRCRGJOB) Tento príklad predloží úlohu skupiny prostriedkov klastra CRG1 v uzle UZOL1. Ak chcete v uzle UZOL1 spustiť úlohu skupiny prostriedkov klastra, v tomto uzle musí byť aktívne klastrovanie.
2. Znova spustíte klastrovanie na tomto uzle.

Ak používate produkt na manažovanie klastrov od obchodného partnera IBM, pozrite si dokumentáciu dodávanú k tomuto produktu.

#### **Súvisiace koncepty**

“Úloha štruktúry a užívateľské fronty” na strane 112

Pri manažovaní klastra musíte poznať štruktúry úloh a užívateľské fronty.

#### **Súvisiace úlohy**

“Ukončenie klastrového uzla” na strane 101

Zastavenie, alebo ukončenie uzla, zastavuje na uzle služby klastrových prostriedkov.

“Spustenie klastrového uzla” na strane 100

Spustenie klastrového uzla spustí na tomto uzle služby klastrových prostriedkov. Od verzie 3 sa uzol môže spustiť sám a bude schopný opätovne sa pripojiť k aktívnemu klastru, ak v ňom nájde aktívny uzol.

## **Obnovenie poškodeného klastrového objektu**

Hoci je nepravdepodobné, že sa poškodí objekt, na druhej strane je možné, že sa poškodia objekty služieb klastrových prostriedkov.

Ak ide o aktívny uzol, systém sa ho pokúsi obnoviť z iného aktívneho uzla v klastru. Pri tom vykoná nasledovné obnovovacie kroky:

### **Pre poškodený interný objekt**

1. Uzol s poškodenými ukončeniami.
2. Ak existuje ešte aspoň jeden uzol v rámci klastra, poškodený uzol sa automaticky sám reštartuje a znova spojí klaster. Proces opätovného spojenia naprávi poškodenú situáciu.

### **Pre poškodenú skupinu prostriedkov klastra**

1. Uzol s poškodeným CRG zlyhá pri každej vykávajúcej sa operácii, ktorá je priradená k tomuto CRG. Systém sa potom pokúsi automaticky obnoviť CRG z ďalšieho aktívneho uzla.
2. Iba vtedy, ak existuje aspoň jeden aktívny člen v obnovovacej doméne, bude pracovať obnova CRG. V opačnom prípade sa úloha CRG ukončí.

Ak systém nemôže identifikovať alebo dosiahnuť žiadny iný aktívny uzol, musíte vykonať tieto kroky na obnovu.

### **Pre poškodený interný objekt**

Zobrazí sa interná chyba klastrovania (CPFBB46, CPFBB47 alebo CPFBB48).

1. Ukončíte klastrovanie pre zničený uzol.
2. Reštartujte klastrovanie pre zničený uzol. Urobte to z iného aktívneho uzla v klastru.
3. Ak kroky 1 a 2 nevyriešia daný problém, odstráňte zničený uzol z klastra.
4. Pridajte systém späť ku klastru a k obnovovacej doméne pre príslušné skupiny klastrových prostriedkov.

### **Pre poškodenú skupinu prostriedkov klastra**

Zobrazí sa chyba informujúca o poškodení objektu (CPF9804).

1. Ukončíte klastrovanie na uzle so zničenou skupinou klastrových prostriedkov.
2. Vymažte CRG (pomocou príkazu DLTCRG).
3. Ak v klastru neexistuje žiaden ďalší aktívny uzol s objektom CRG, potom ho obnovte z média.

4. Spustíte klastrovanie na uzle so zničenou skupinou klastrových prostriedkov. Môžete to urobiť akéhokoľvek aktívneho uzla.
5. Keď začnete klastrovanie, systém znova zosynchronizuje všetky skupiny klastrových prostriedkov. Možno budete musieť znova vytvoriť CRG, ak ho žiaden iný uzol v klastru neobsahuje.

## Obnova klastrov po úplnej strate systému

Tieto informácie môžete spolu s vhodným kontrolným zoznamom z manuálu Zálohovanie a obnova použiť na obnovu celého systému po úplnej strate systému, ak dôjde k neočakávanému prerušeniu napájania vášho servera.

### Scenár 1: Obnova do rovnakého systému

1. Aby ste zabránili nekonzistencii informácií domény zariadení medzi licenčným interným kódom a systémom i5/OS, odporúča sa nainštalovať licenčný interný kód pomocou voľby 3 (Nainštalovať licenčný interný kód a obnoviť konfiguráciu).

**Poznámka:** Ak má byť operácia Inštalovať licenčný interný kód a obnoviť konfiguráciu úspešná, musíte mať rovnaké diskové jednotky -- s výnimkou diskovej jednotky zdroja načítania, ak zlyhala. Musíte tiež obnovovať rovnaké vydanie.

2. Po nainštalovaní licenčného interného kódu postupujte podľa procedúry *Ako obnoviť konfiguráciu vašich diskov* z kapitoly 5 manuálu *Zálohovanie a Obnova*. Tieto kroky vám pomôžu vyhnúť sa rekonfigurácii ASP.
3. Po tom, čo ste obnovili svoje systémové informácie, pripravili sa na spustenie klastrovania na uzle, ktorý ste práve obnovili, musíte začať klastrovanie aktívneho uzla. Tým odovzdáte obnovenému uzlu najaktuálnejšie konfiguračné informácie.

### Scenár 2: Obnova do iného systému

Po tom, čo ste obnovili systémové informácie a kontrolou protokolu úloh sa uistili, že sú obnovené všetky objekty, musíte pri získavaní správnej konfigurácie domény zariadení klastra, vykonať nasledujúce kroky.

1. Zmažte klaster z práve obnoveného uzla.
2. Na aktívnom uzle vykonajte tieto kroky:
  - a. Odstráňte obnovený uzol z klastra.
  - b. Pridajte obnovený uzol naspäť do klastra.
  - c. Pridajte obnovený uzol do domény zariadení.
  - d. Vytvorte skupinu klastrových prostriedkov, alebo pridajte tento uzol do domény obnovy.

#### Súvisiace úlohy

“Zálohovanie a obnova klastrov” na strane 114

Aj ak vo vašich systémoch používate klastrovanie, stále je dôležité, aby ste vytvorili stratégiu zálohovania a obnovy na ochranu vašich údajov.

#### Súvisiace informácie

Zálohovanie a obnova

## Obnova klastrov po havárii

V prípade havárie, pri ktorej ste stratili všetky uzly, potrebujete svoj klaster rekonfigurovať.

Aby ste na to boli pripravení, odporúčame vám, aby ste si uložili konfiguračné informácie vášho klastra a aby ste si odložili aj ich vytlačenú verziu.

#### Súvisiace úlohy

“Zálohovanie a obnova klastrov” na strane 114

Aj ak vo vašich systémoch používate klastrovanie, stále je dôležité, aby ste vytvorili stratégiu zálohovania a obnovy na ochranu vašich údajov.

## Obnova klastrov zo záložných pásov

Počas normálnej prevádzky by nikdy nemalo byť nutné vykonať obnovu zo záložnej pásky.

Toto je potrebné len v prípade havárie a straty všetkých uzlov v klastrí. Ak dôjde k havárii, vykonajte obnovu podľa vašich bežných procedúr obnovy, ktoré ste ustanovili po vytvorení stratégie zálohovania a obnovy.

#### **Súvisiace úlohy**

“Zálohovanie a obnova klastrov” na strane 114

Aj ak vo vašich systémoch používate klastrovanie, stále je dôležité, aby ste vytvorili stratégiu zálohovania a obnovy na ochranu vašich údajov.

#### **Súvisiace informácie**

Zálohovanie a obnova

## **Často kladené otázky o manažovaní klastrov v Navigátore iSeries**

Otázky a odpovede týkajúce sa grafického užívateľského prostredia Navigátora iSeries na vytváranie a manažovanie klastrov.

Grafické užívateľské rozhranie IBM na vytváranie a manažovanie klastrov je dostupné v Navigátore iSeries a poskytuje ho voľba 41 (Vysokodostupné prepínateľné prostriedky). Detaily o rozhraní nájdete v téme Manažovanie klastrov v Navigátore iSeries.

Tu je uvedený zoznam otázok a odpovedí týkajúcich sa manažovania klastrov v Navigátore iSeries.

### **Všeobecná otázka**

1. Existuje formulár, ktorý zobrazuje požiadavky na tvorbu klastra?

### **Manažovanie klastrov v Navigátore iSeries**

1. Kde sa v rozhraní Navigátora iSeries nachádza funkcia Klastre?
2. Ako vytvorím klaster?
3. Aký je vzťah medzi zložkou klastra a systémovou skupinou riadiacej centrály?
4. Pre niektoré systémy iSeries v sieti už mám definovaný klaster. Ako ho môžem pridať, aby som ho mohol zobrazovať a manažovať pomocou Navigátora iSeries?
5. Žiaden z uzlov môjho klastra nie je v stave "Spustený". Ktorý uzol mám spustiť ako prvý?
6. Prečo záleží na tom, ktorý z uzlov bude spustený ako prvý?
7. Čo znamená stĺpec "Aktuálny primárny uzol" v zložkách prepínateľných zariadení a prepínateľných aplikácií?
8. Ako v Navigátore iSeries nájdem skupinu prostriedkov klastra (CRG) zariadenia?
9. Ako v Navigátore iSeries nájdem skupinu prostriedkov klastra (CRG) aplikácie?
10. Ako v Navigátore iSeries nájdem skupinu prostriedkov klastra (CRG) údajov?
11. Chcem zobrazíť stav prepínateľného zariadenia (CRG zariadenia) bez toho, aby som musel prejsť späť do zložky Prepínateľný hardvér. Ako to môžem urobiť?

### **Komunikácia**

1. Akú adresu IP používa funkcia Klastre v Navigátore iSeries na komunikáciu s uzlami klastrí? Nepoužíva IP adresy názvu tohto uzla?

### **Zabezpečenie**

1. Prečo je väčšina kontextových ponúk v zložke Klastre v Navigátore iSeries deaktivovaná alebo sa nezobrazuje?
2. Používa funkcia Klastre v Navigátore iSeries hodnoty správy aplikácií?
3. Prečo funkcia Klastre v Navigátore iSeries zobrazuje mojím uzlom v klastrí prihlasovacie okno?

### **Odstraňovanie porúch**

1. Prečo sa mi v riadiacej centrále nezobrazí zložka klastra?
2. Už mám vytvorený klaster, ale nezobrazuje sa v zložke klastra. Prečo?
3. Prečo sa v zložke klastra nezobrazí posledný stav?

4. Prečo sa nevykonalo núdzové prepnutie mojej skupiny prepínateľného hardvéru alebo prepínateľnej aplikácie?
5. Dostal som správu o poškodenom objekte. Čo s tým môžem urobiť?
6. V sprievodcovi pre uzly používam tlačidlo "Prehľadávať" na vyhľadávanie adries IP. Prečo sa v prehľadávacom okne nezobrazujú všetky očakávané adresy TCP/IP?
7. Prečo je väčšina kontextových ponúk v zložke Klastre v Navigátore iSeries deaktivovaná alebo sa nezobrazuje?
8. Používal som sprievodcu "Nový klaster" a zobrazil sa panel s nadpisom: "Nový klaster - Nenašiel sa žiadny prepínateľný softvér". Je to zlé?
9. Jeden z mojich uzlov vykazuje stav "Nekomunikuje". Ako to napravím?

## Všeobecná otázka

### Existuje formulár, ktorý zobrazuje požiadavky na tvorbu klastra?

Áno. Použite Kontrolný zoznam konfigurácie klastra a uistite sa, že ste pripravení vo svojom prostredí nakonfigurovať klaster.

Späť k otázkam

### Manažovanie klastrov v Navigátore iSeries: Kde sa v rozhraní Navigátora iSeries nachádza funkcia Klastre?

Rozhranie Navigátora iSeries na manažovanie klastrov je dostupné ako súčasť softvérového balíka IBM iSeries Access. Funkcia Klastre sa nachádza v zložke Riadiaca centrála v Navigátore iSeries. Detaily nájdete v téme Manažovanie klastrov v Navigátore iSeries.

Späť k otázkam

### Ako vytvorím klaster?

Ak chcete pomocou Sprievodcu Nový klaster v Navigátore iSeries vytvoriť jednoduchý klaster, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Pravým tlačidlom myši kliknite na **Klastre** a vyberte voľbu **Nový klaster**.
3. Postupujete podľa sprievodcu, a tak vytvorte nový klaster.

Viac informácií o vytváraní a konfigurácii klastra nájdete na adrese Konfigurovať klaster.

Späť k otázkam

### Aký je vzťah medzi zložkou klastra a systémovou skupinou riadiacej centrály?

Keď pomocou Navigátora iSeries vytvoríte klaster, vytvorí sa tiež skupina systémov v serveri Riadiacej centrály. Systémová skupina je pomenovaná tak isto ako klaster a koncové systémy v systémovej skupine sú uzlami v klastru. Skupina systémov má tiež svoj špeciálny typ, takže Navigátor iSeries vie, že sa jedná o špeciálnu skupinu systémov, reprezentujúcu klaster.

**Dôležité:** Systém Riadiacej centrály obsahuje skupiny systémov. Ak v Navigátore iSeries zmeníte váš aktuálny systém Riadiacej centrály, nový systém Riadiacej centrály nebude mať špeciálne skupiny systémov pre klastre a preto sa tieto klastre nezobrazia v zložke Klastre.

Späť k otázkam

### Pre niektoré systémy iSeries v sieti už mám definovaný klaster. Ako ho môžem pridať, aby som ho mohol zobrazovať a manažovať pomocou Navigátora iSeries?

Ak chcete pridať existujúci klaster, aby sa zobrazil v Navigátore iSeries, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Pravým tlačidlom myši kliknite na **Klastre** a vyberte **Pridať nový klaster**
3. V okne **Pridať existujúci klaster** špecifikujte jeden zo serverov v klastri.
4. Kliknite na OK.

Späť k otázkam

### **Žiaden z uzlov môjho klastra nie je v stave "Spustený". Ktorý uzol mám spustiť ako prvý?**

Mali by ste spustiť ten uzol, ktorý bol ako posledný v stave "Spustený". Povedzme napríklad, že máte dva uzly vo svojom klastri: uzol A a B. Uzol A ani uzol B práve nie sú spustené. Avšak uzol B bol posledným spusteným uzlom so stavom "Spustený". Ako prvý by ste mali spustiť uzol B, pretože bude obsahovať najnovšie informácie o klastri.

Späť k otázkam

### **Prečo záleží na tom, ktorý z uzlov bude spustený ako prvý?**

Záleží na tom preto, lebo uzol, ktorý bol ako posledný v stave "Spustený", obsahuje najnovšie informácie o klastri. Je to dôležité, pretože ak spustíte uzol, ktorý bol najdlhšie odpojený, tento môže obsahovať zastarané informácie. Nebezpečné je to, že zastarané informácie sa môžu rozšíriť do ostatných uzlov v klastri, keď tieto uzly nie sú spustené. Povedzme napríklad, že máte dvojuzlový klaster s uzlami A a B. Ak uzol B bol posledným aktívnym uzlom v stave "Spustený", potom bude obsahovať posledné informácie o klastri. Ak sa rozhodnete spustiť ako prvý uzol A, potom tento môže obsahovať niektoré zastarané informácie, ale aj napriek tomu bude spustený. Keď neskôr spustíte uzol B, spojí sa s práve aktívnym uzlom v klastri (spojí sa s uzlom A). Zastaraná klastrová informácia z uzla A sa preniesie do uzla B, čo má za následok, že obidva uzly budú obsahovať zastarané informácie o klastri. Preto je dôležité spustiť ako prvý uzol B. Zastarané informácie o klastri môžu mať dopad na konfiguráciu prepínateľných zariadení. Ak zistíte problémy pri spúšťaní prepínateľných zariadení, pretože sa diskové jednotky ohlasujú v záložnom uzle, keď skupina prepínateľného hardvéru zobrazuje iný aktuálny uzol, budete musieť zmeniť rolu uzlov v doméne obnovy a uzol vlastníaci diskové jednotky nastaviť ako primárny uzol.

Späť k otázkam

### **Čo znamená stĺpec "Aktuálny primárny uzol" v zložkách Prepínateľný hardvér, Prepínateľný softvér a Prepínateľné údaje?**

Stĺpec "Aktuálny primárny uzol" označuje, že uzol aktuálne slúži ako primárny uzol pre prepínateľné zariadenie alebo prepínateľný softvérový produkt. Alebo v terminológii API klastra to znamená, že je to uzol s aktuálnou rolou v primárnej obnovovacej doméne.

Späť k otázkam

### **Ako v Navigátore iSeries nájdem skupinu prostriedkov klastra (CRG) zariadenia?**

Zariadenia skupín klastrových prostriedkov CRG sú označované ako prepínateľné hardvérové skupiny a nachádzajú sa v zložke **Prepínateľný hardvér** v zložke klastrov.

Späť k otázkam

### **Ako v Navigátore iSeries nájdem skupinu prostriedkov klastra (CRG) aplikácie?**

Aplikácie skupín klastrových prostriedkov CRG sú označované ako prepínateľné softvérové produkty a nachádzajú sa v zložke **Prepínateľný softvér** v zložke klastrov.

Späť k otázkam

## Ako v Navigátore iSeries nájdem skupinu prostriedkov klastra (CRG) údajov?

Údaje skupín klastrových prostriedkov CRG sú označované ako prepínateľné skupiny úloh a nachádzajú sa v zložke **Prepínateľné údaje** v zložke klastrov.

Späť k otázkam

## Chcem vidieť stav skupiny prepínateľného hardvéru (zariadenie CRG) bez toho, aby som musel ísť naspäť na zložku Prepínateľný hardvér. Ako to môžem urobiť?

Ako alternatívu pre navigáciu do zložky Prepínateľný hardvér môžete použiť otvorenie nového okna so zobrazením prepínateľného hardvéru kliknutím pravého tlačítka myši na zložku **Prepínateľný hardvér** a vybrať **Otvoriť**. Osobitné okno zobrazí prepínateľné hardvérové skupiny (zariadenia CRG) a ich priradené informácie o stave. Toto sa dá takisto použiť aj pre **Prepínateľný softvér** a **Prepínateľné údaje**.

Späť k otázkam

## Komunikácia: Akú adresu IP používa funkcia Klastre v Navigátore iSeries na komunikáciu s uzlami klastrí? Nepoužíva adresu IP pre názov uzla?

V hlavnej klastrovej zložke sa nachádza stĺpec "Server", ktorý zobrazuje informácie o vašich nakonfigurovaných klastroch. Názov servera sa nachádza aj na paneli vlastností pre každý klaster. Server uvedený v stĺpci "Server" predstavuje uzol klastra, ktorý používa rozhranie Navigátora iSeries na komunikáciu s klastrom. Týka sa to len spôsobu, akým Navigátor iSeries komunikuje s objektom klastra v serveri, nie spôsobu, akým navzájom komunikujú uzly klastra. Server, ktorý používa manažovanie klastrov v Navigátore iSeries, nemá nič spoločné s aktuálnym serverom Riadiacej centrály.

Ak sa vypne uzol, ktorý používa Navigátor iSeries na komunikáciu s klastrom, môžete komunikačný prostriedok, pomocou ktorého sa budú vykonávať klastrové funkcie, zmeniť na iný uzol klastra.

Ak chcete zmeniť server, ktorý bude používať rozhranie Navigátora iSeries na komunikáciu s klastrom, vykonajte tieto kroky:

1. V Navigátore iSeries rozviňte položku **Riadiaca centrála**.
2. Rozviňte **Klastre**.
3. Pravým tlačidlom myši kliknite na klaster a vyberte voľbu **Zmeniť server**.

Späť k otázkam

## Bezpečnosť: Prečo je väčšina kontextových ponúk v zložke Klastre v Navigátore iSeries deaktivovaná alebo sa nezobrazuje?

Niektoré operácie sú k dispozícii len v závislosti na stave aktuálnej konfigurácie vášho klastra. Napríklad nemôžete zastaviť uzol, ktorý už je zastavený, nemôžete pridať uzol ku klastru, ktorý už obsahuje maximálne množstvo nakonfigurovaných uzlov, čiže štyri. Online pomoc pre jednotlivé úlohy obsahuje vysvetlenie, prečo sú niektoré z týchto položiek deaktivované alebo nedostupné.

Niektoré informácie nie sú k dispozícii, ak nemáte dostatočné oprávnenie. Ak používate Navigátor iSeries a máte oprávnenie triedy užívateľov \*SECOFR, budete mať prístup k všetkým operáciám a správe klastra. Navigátor iSeries určí pomocou oprávnenia správy aplikácií z aktuálneho systému Riadiacej centrály, či máte oprávnenie správy aplikácií na rôzne operácie manažovania klastrov v Navigátore iSeries.

Ďalšie informácie o práci s aplikačnou administratívou nájdete v téme Aplikačná administratíva.

Späť k otázkam



## **Používa funkcia Klastre v Navigátore iSeries hodnoty správy aplikácií?**

Áno. Manažovanie klastrov v Navigátore iSeries určí pomocou hodnôt oprávnení správy aplikácií z aktuálneho systému Riadiacej centrály, či máte oprávnenie správy aplikácií na rôzne klastrové operácie.

Navigátor iSeries má dva typy nastavení oprávnenia pre prístup: **Klastrová operácia** a **Správa klastra**

Pomocou oprávnenia **klastrovej operácie** môžete:

- zobrazí stav klastra
- Spustiť a zastaviť uzly
- Spustiť a zastaviť prepínateľný hardvér a prepínateľný softvér
- vykonať manuálne prepnutie prepínateľného hardvéru a prepínateľného softvéru

Pomocou oprávnenia **klastrovej administrácie** môžete:

- vytvoriť/vymazať klastre
- Pridať vzdialené uzly
- Pridať a vymazať prepínateľný hardvér, prepínateľný softvér a diskové oblasti
- zmeniť vlastnosti prepínateľného hardvéru a prepínateľného softvéru

Späť k otázkam

## **Prečo funkcia Klastre v Navigátore iSeries zobrazuje mojim uzlom v klastri prihlasovacie okno?**

V niektorých prípadoch sa bude Navigátor iSeries pokúšať komunikovať so všetkými uzlami v klastri. Závisí to od stavu vášho klastra. Keď Navigátor iSeries potrebuje komunikovať s uzlom, najprv prehľadá vyrovnávaciu pamäť existujúcich prihlásení v Navigátore iSeries a pokúsi sa nájsť existujúce otvorené pripojenie. Ak nenájde existujúce pripojenie, potom požiada užívateľa o prihlásenie. Ak zrušíte prihlasovacie okno, Navigátor iSeries sa pokúsi povoliť užívateľovi vykonávanie klastrových operácií. Ak Navigátor iSeries nemôže komunikovať s uzlami, nemusí byť možné vykonávať všetky operácie.

Späť k otázkam

## **Odstraňovanie problémov: Prečo sa v Riadiacej centrále nezobrazuje zložka Klastre?**

Je možné, že ste vo vašom PC nevykonali úplnú inštaláciu produktu iSeries Access for Windows. Možno ste vykonali základnú inštaláciu alebo vybrali niektoré vlastné voľby. Detaily o inštalácii nájdete v téme iSeries Access.

Späť k otázkam

## **Už mám vytvorený klaster, ale nezobrazuje sa v zložke klastra. Prečo?**

Na túto otázku je krátka odpoveď: klaster sa nezobrazuje, pretože vo vašom systéme riadiacej centrály neexistuje systémová skupina, ktorá reprezentuje tento klaster. Túto skupinu systémov reprezentujúcu klaster vytvára manažment klastrov v Navigátore iSeries pri vytvorení klastra alebo pri pridaní klastra do zložky Klastre pomocou akcie "Pridať existujúci klaster". Ak chcete zobrazíť systémové skupiny, v riadiacej centrále rozviňte zložku **Systémové skupiny**. Klastrové systémové skupiny sa zobrazia ako systémové skupiny tretích strán, ale nepokladajte všetky tieto systémové skupiny tretích strán za klastre.

Späť k otázkam

## **Prečo sa v zložke klastra nezobrazí posledný stav?**

Navigátor iSeries zobrazuje informácie o nakonfigurovaných klastroch pre daný moment tak, že prejde do uzlov klastra, získa najnovšie informácie o klastri a potom ich zobrazí v okne Navigátora iSeries. Tým sa však automaticky

nevykonáva pravidelná aktualizácia informácií. Najlepší spôsob, ako získať najnovšiu snímku informácií, je urobiť manuálnu obnovu. Môžete použiť ponuku **Zobrazíť** v Navigátore iSeries a potom zvolíť voľbu **Obnoviť**. Prípadne môžete nastaviť Navigátor iSeries, aby vykonával automatickú obnovu.

Späť k otázkam

### **Prečo sa nevykonalo núdzové prepnutie môjho prepínateľného zariadenia, prepínateľnej aplikácie alebo prepínateľnej skupiny údajov?**

Najpravdepodobnejšie je, že ste v klastri nemali spustený prepínateľný prostriedok (skupinu klastrových prostriedkov). Inými slovami, stav prepínateľných prostriedkov pred tým, ako nastalo automatické prevzatie služby, bol "Spustené". Vaše prepínateľné prostriedky musia byť spustené pre prípad, že nastane prevzatie služby.

Späť k otázkam

### **Dostal som správu o poškodenom objekte. Čo s tým môžem urobiť?**

Možno ste prijali podobnú správu: CPF811C užívateľský front QUGCLUSRQ v QCLUMGT je poškodený

**1. možnosť:** Jednou z možností je vymazať objekt a obnoviť ho. Je to možné iba vtedy, ak ste predtým daný objekt uložili.

**2. možnosť:** Vymažte poškodený objekt. Napríklad, ak je poškodený príkaz QUGCLUSRQ v knižnici QCLUMGT, vymažte tento objekt. Potom existujúci klastor pridajte do Navigátora iSeries. Po pridaní klastra GUI klastra skontroluje, či objekty existujú a znova ich vytvorí, ak už neexistujú. Pozrite si časť Ako môžem pridať existujúci klastor, aby som ho mohol zobrazovať a manažovať pomocou Navigátora iSeries?

Späť k otázkam

### **V sprievodcovi pre uzly používam tlačidlo "Prehľadávať" na vyhľadávanie adries IP. Prečo sa v prehľadávacom okne nezobrazujú všetky očakávané adresy TCP/IP?**

Je to len zoznam kandidátov možných adries IP. Nemusíte sa obmedziť na zoznam možných adries, zobrazených v okne. Môžete zadať akúkoľvek adresu klastrového rozhrania. Berte však na vedomie, že neskôr sa zobrazia chyby, ak sa Navigátor iSeries nebude môcť pripojiť pomocou adresy IP, ktorú zadáte ako primárnu. Navigátor iSeries používa primárnu adresu IP na pripojenie k uzlu v klastri.

Späť k otázkam

### **Používal som sprievodcu "Nový klastor" a zobrazil sa panel s nadpisom: "Nový klastor - Nenašiel sa žiadny prepínateľný softvér". Je to zlé?**

Nie, nie je to zlé a nie je to ani chyba. Znamená to presne to, čo je uvedené; rozhranie Navigátora iSeries nemôže nájsť žiadny prepínateľný softvér, ktorý by mohol sprievodca automaticky nainštalovať. Navigátor iSeries vyžaduje, aby všetok automaticky nainštalovateľný prepínateľný softvér zodpovedal architektúre i5/OS pre aplikácie podporujúce klastre. Okrem toho Navigátor iSeries podporuje len podmnožinu tejto architektúry, nie celú.

Späť k otázkam

### **Jeden z mojich uzlov vykazuje stav "Nekomunikuje". Ako to napravím?**

Rozdelenie klastra nastáva vtedy, ak stratíte kontakt medzi jedným alebo viacerými uzlami v klastri a nemôže byť potvrdené zlyhanie stratených uzlov. Viac informácií nájdete v téme Chyby rozdelenia.

Niekedy je oznámená podmienka rozdelenia, keď naozaj nastal výpadok uzla. Táto situácia nastane vtedy, keď služby klastrových prostriedkov stratia komunikáciu v rámci jedného alebo medzi viacerými uzlami a nedá sa zistiť ich funkčnosť. V takomto prípade je dostupný jednoduchý mechanizmus na indikáciu zlyhania uzla. Ďalšie informácie nájdete v téme Zmeniť rozdelené uzly na zlyhané.

Späť k otázkam

## Kam volať pre podporu klastrov

Túto tému si pozrite, ak potrebujete kontaktovať IBM ohľadom otázok týkajúcich sa klastrov.

Ak potrebujete pomoc pri rozhodovaní, či sú pre váš obchod výhodné klastre, alebo ak ste sa dostali do problémov po implementovaní klastrov, môžete kontaktovať tento servis:

- Ak potrebujete technickú a obchodnú asistenciu alebo ak chcete využiť konzultačné služby IBM, kontaktujte Centrum pre nepretržitú dostupnosť v Technologickom centre iSeries pomocou e-mailovej adresy [rchelst@us.ibm.com](mailto:rchelst@us.ibm.com).
- V prípade ostatných problémov kontaktujete obchodného partnera IBM, od ktorého ste získali softvérový balík pre klastrovanie, alebo volajte telefónne číslo 1-800-IBM-4YOU (1-800-426-4968).

### Súvisiace úlohy

“Konfigurácia klastrov” na strane 97






Pochopte postup pri vytváraní klastra.

---


## Súvisiace informácie pre klastre

Nájdete tu informácie súvisiace s klastrami.

### Dokumenty Redbook

- Data Resilience Solutions for IBM i5/OS High Availability Clusters 
- Clustering and IASPs for Higher Availability 
- High Availability on the AS/400 System: A System Manager's Guide 
- IBM eServer iSeries Independent ASPs: A Guide to Moving Applications to IASPs 
- The System Administrator's Companion to AS/400 Availability and Recovery 

### Webové lokality


- High Availability and Clusters  ([www.ibm.com/servers/eserver/series/ha](http://www.ibm.com/servers/eserver/series/ha))
- Lokalita IBM pre vysokú dostupnosť a klastre

### Uloženie súborov PDF

Ak chcete uložiť PDF vo vašej pracovnej stanici za účelom zobrazenia alebo tlače:

1. Pravým tlačidlom myši kliknite na PDF vo vašom prehliadači (kliknite pravým tlačidlom myši na odkaz hore)
2. Ak používate prehliadač Internet Explorer, kliknite na **Save Target As**. Ak používate prehliadač Netscape Communicator, kliknite na **Save Link As**.
3. Prejdite do adresára, do ktorého chcete uložiť PDF.
4. Kliknite na **Save**.

## | **Prevzatie programu Adobe Acrobat Reader**

- | Na zobrazenie alebo tlač týchto dokumentov PDF potrebujete Adobe Acrobat Reader. Kópiu tohto programu môžete
- | prevziať z webovej lokality spoločnosti Adobe ([www.adobe.com/products/acrobat/readstep.html](http://www.adobe.com/products/acrobat/readstep.html))  .

---

## **Informácie o licencií na kód a právne vyhlásenia**

IBM vám udeľuje neexkluzívne právo na používanie všetkých príkladov programovania, z ktorých môžete vygenerovať podobnú funkciu prispôbenú pre vaše vlastné špecifické potreby.

- | S VÝNIMKOU ZÁRUK VYPLÝVAJÚCICH ZO ZÁKONA, KTORÉ NEMOŽNO ODOPRIEŤ, IBM, JEJ
- | VÝVOJÁRI PROGRAMOV A DODÁVATELIA NEDÁVAJÚ ŽIADNE VYJADRENÉ ANI PREDPOKLADANÉ
- | ZÁRUKY ALEBO PODMIENKY, VRÁTANE ALEBO BEZ OBMEDZENIA LEN NA PREDPOKLADANÉ
- | ZÁRUKY ALEBO PODMIENKY PREDAJNOSTI, VHODNOSTI NA URČITÝ ÚČEL A DODRŽIAVANIA
- | AUTORSKÝCH PRÁV TÝKAJÚCICH SA PROGRAMU ALEBO TECHNICKEJ PODPORY, AK JE NEJAKÁ.
  
- | V ŽIADNOM PRÍPADE NIE SÚ IBM, JEJ VÝVOJÁRI PROGRAMOV ALEBO DODÁVATELIA ZODPOVEDNÍ
- | ZA NIČ Z NASLEDUJÚCEHO, AJ KEĎ BOLI O TEJTO MOŽNOSTI INFORMOVANÍ:
- | 1. STRATA ALEBO POŠKODENIE DÁT;
- | 2. PRIAME, ŠPECIÁLNE, NÁHODNÉ ALEBO NEPRIAME ŠKODY ANI ŽIADNE NÁSLEDNÉ EKONOMICKÉ
- | ŠKODY; ALEBO
- | 3. UŠLÝ ZISK, STRATU OBCHODU, ZISKU, DOBRÉHO MENA ALEBO OČAKÁVANÝCH ÚSPOR.
  
- | NIEKTORÉ JURISDIKCIE NEPOVOĽUJÚ VÝNIMKY ALEBO OBMEDZENIA PRIAMYCH, NÁHODNÝCH
- | ALEBO NÁSLEDNÝCH ŠKÔD, PRETO SA VÁS NEMUSIA TÝKAŤ NIEKTORÉ ALEBO VŠETKY Z
- | HOREUVEDENÝCH OBMEDZENÍ ALEBO VÝNIMIEK.

---

## Príloha. Právne informácie

Tieto informácie boli vyvinuté pre produkty a služby ponúkané v USA.

IBM nemusí ponúkať produkty, služby alebo vlastnosti opisované v tomto dokumente v iných krajinách. Informácie o aktuálne dostupných produktoch a službách vo vašej krajine získate od predstaviteľa lokálnej pobočky IBM. Žiadny odkaz na produkt, program alebo službu IBM nie je myslený tak a ani neimplikuje, že sa môže používať len tento produkt, program alebo služba od IBM. Namiesto nich sa môže použiť ľubovoľný funkčne ekvivalentný produkt, program alebo služba, ktorá neporušuje intelektuálne vlastnícke právo IBM. Vyhodnotenie a kontrola činnosti produktu, programu alebo služby inej ako od IBM je však na zodpovednosti užívateľa.

IBM môže mať patenty alebo podané prihlášky patentov týkajúcich sa predmetu opísanom v tomto dokumente. Opatrenie tohto dokumentu vám neudeľuje žiadnu licenciu na tieto patenty. Požiadavky na licencie môžete zasielať písomne na:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Otázky na licencie týkajúce sa dvojbajtových (DBCS) informácií adresujte Oddeleniu intelektuálneho vlastníctva IBM vo vašej krajine alebo ich pošlite písomne na adresu:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106-0032, Japan

**Nasledujúci odsek sa netýka Veľkej Británie alebo akejkoľvek inej krajiny, v ktorej sú takéto ustanovenia nezlučiteľné s miestnym zákonom:** INTERNATIONAL BUSINESS MACHINES CORPORATION POSKYTUJE TÚTO PUBLIKÁCIU "TAK AKO JE" BEZ AKÝCHKOĽVEK GARANCIÍ, ČI UŽ VYJADRENÝCH ALEBO IMPLIKOVANÝCH, ALE NEOBMEDZENÝCH NA IMPLIKOVANÉ GARANCIE NEPORUŠENIA, SCHOPNOSTI UVEDENIA NA TRH ALEBO SPÔSOBILOSTI NA URČITÝ ÚČEL. Niektoré štáty nedovoľujú zriecť sa vyjadrených alebo implikovaných záruk v určitých transakciách, preto sa vás toto vyhlásenie nemusí týkať.

Tieto informácie môžu obsahovať technické nepresnosti alebo typografické chyby. Tieto informácie sa periodicky menia; tieto zmeny budú začlenené do nových vydaní publikácie. IBM môže kedykoľvek bez ohlásenia spraviť zmeny a/alebo vylepšenia v produkte(och) a/alebo programe(och) opísanom v tejto publikácii.

Všetky odkazy v týchto informáciách na webové lokality iné ako od IBM sú poskytnuté len pre pohodlie a v žiadnom prípade neslúžia ako potvrdenie obsahu týchto webových lokalít. Materiály na týchto webových lokalitách nie sú časťou produktov IBM a použitie týchto webových lokalít je na vaše vlastné riziko.

IBM môže použiť alebo distribuovať všetky informácie, ktoré jej poskytnete, ľubovoľným spôsobom, ktorý uzná za vhodný, bez toho, aby vznikol voči vám akýkoľvek záväzok.

Držitelia licencie na tento program, ktorí o ňom chcú získať viac informácií z dôvodu povolenia: (i) výmeny informácií medzi nezávisle vytvorenými programami a inými programami (vrátane tohto) a (ii) vzájomné používanie vymieňaných informácií, by mali kontaktovať:

IBM Corporation  
Software Interoperability Coordinator, Department YBWA  
3605 Highway 52 N

Rochester, MN 55901  
U.S.A.

Takéto informácie môžu byť dostupné, môžu byť predmetom príslušných pojmov a podmienok a v niektorých prípadoch sú dostupné za poplatok.

- | Licenčný program, opisovaný v týchto informáciách, a všetky preň dostupné licenčné materiály poskytuje IBM podľa
- | podmienok zmluvy IBM Customer Agreement, IBM International Program License Agreement, IBM License
- | Agreement for Machine Code alebo ľubovoľnej ekvivalentnej zmluvy medzi nami.

Všetky údaje o výkone, uvádzané v tomto dokumente boli získané v riadenom prostredí. Výsledky získané v iných prevádzkových prostrediach sa môžu podstatne odlišovať. Niektoré merania boli vykonané v systémoch vývojovej úrovne a nie je žiadna záruka, že tieto merania budú rovnaké vo všeobecne dostupných systémoch. Okrem toho, niektoré výsledky boli odhadnuté extrapoláciou. Skutočné výsledky sa môžu odlišovať. Užívatelia tohto dokumentu by si mali overiť použiteľnosť týchto údajov pre svoje špecifické prostredie.

Informácie o produktoch iných ako od IBM boli získané od poskytovateľov týchto produktov, z ich uverejnených oznámení alebo z iných, verejne dostupných zdrojov. IBM netestovala tieto produkty a nemôže potvrdiť presnosť ich výkonu, kompatibilitu ani žiadne iné tvrdenie týkajúce sa produktov iných ako od IBM. Otázky k schopnostiam produktov iných ako od IBM by ste mali adresovať poskytovateľom týchto produktov.

Všetky vyhlásenia týkajúce sa budúceho smerovania alebo úmyslov IBM sú predmetom zmeny alebo zrušenia bez ohlásenia a vyjadrujú len zámery a ciele.

Všetky ceny IBM sú navrhované predajné ceny stanovené spoločnosťou IBM, sú aktuálne a sú predmetom zmeny bez ohlásenia. Ceny dilerov môžu byť odlišné.

Tieto informácie slúžia len na plánovacie účely. Tu uvedené informácie sú predmetom zmeny pred sprístupnením opisovaných produktov.

Tieto informácie obsahujú príklady údajov a hlásení používaných v každodenných firemných operáciách. Kvôli ich čo najlepšej ilustrácii obsahujú tieto príklady mená osôb, názvy spoločností, pobočiek a produktov. Všetky tieto mená a názvy sú vymyslené a akákoľvek podobnosť s menami, názvami a adresami používanými skutočnými osobami a spoločnosťami je čisto náhodná.

#### LICENCIA NA AUTORSKÉ PRÁVA:

Tieto informácie obsahujú vzorové aplikačné programy v zdrojovom kóde, ktoré ilustrujú programovacie techniky v rôznych platformách. Tieto vzorové programy môžete kopírovať, upravovať a distribuovať v ľubovoľnej forme bez platenia poplatku spoločnosti IBM, za účelom vývoja, použitia, marketingu alebo distribúcie aplikačných programov vyhovujúcich aplikačnému programovému rozhraniu pre prevádzkovú platformu, pre ktorú sú napísané tieto vzorové programy. Tieto príklady neboli dôkladne otestované pri všetkých podmienkach. IBM preto nemôže garantovať alebo predpokladať spoľahlivosť, použiteľnosť ani funkciu týchto programov.

Každá kópia alebo časť týchto vzorových programov alebo odvodená práca musí obsahovať túto poznámku o autorských právach:

© (názov vašej spoločnosti) (rok). Časti tohto kódu sú odvodené od vzorových programov spoločnosti IBM. © Copyright IBM Corp. \_sem napíšte rok alebo roky\_. Všetky práva vyhradené.

Ak si prezeráte elektronickú kópiu týchto informácií, nemusia byť zobrazené fotografie ani farebné ilustrácie.

---

## Informácie o programovom rozhraní

Dokumenty tejto publikácie, Klastre, používali programové rozhrania, ktoré dovoľujú zákazníčkovi písať programy na získanie služieb systémov IBM i5/OS.

---

## Ochranné známky

Nasledujúce pojmy sú ochranné známky spoločnosti International Business Machines v USA, v iných krajinách alebo v oboch:

- | 400
- | i5/OS
- | IBM
- | iSeries
- | OS/400
- | Redbooks

- | Intel, Intel Inside (logá), MMX a Pentium sú ochranné známky spoločnosti Intel Corporation v Spojených štátoch alebo iných krajinách.

Microsoft, Windows, Windows NT a logo Windows sú ochranné známky spoločnosti Microsoft v USA, v iných krajinách alebo v oboch.

Java a všetky ochranné známky založené na Java sú ochranné známky spoločnosti Sun Microsystems v USA, v iných krajinách alebo v oboch.

- | Linux je ochranná známka Linusa Torvaldsa v Spojených štátoch alebo iných krajinách.

UNIX je registrovaná ochranná známka spoločnosti The Open Group v USA a iných krajinách.

Ostatné názvy spoločnosti, produktov alebo služieb môžu byť ochranné známky alebo značky služieb iných.

---

## Pojmy a podmienky

Oprávnenia na používanie týchto publikácií sú predmetom nasledujúcich pojmov a podmienok.

**Osobné použitie:** Tieto publikácie môžete reprodukovať pre svoje osobné, nekomerčné použitie za podmienky zachovania všetkých informácií o autorských právach. Bez výslovného povolenia autora ich nemôžete distribuovať, zobrazovať ani odvádzať práce z týchto publikácií ani žiadnej ich časti.

**Komerčné použitie:** Tieto publikácie môžete reprodukovať, distribuovať a zobrazovať výlučne vo vašej spoločnosti za podmienky zachovania všetkých informácií o autorských právach. Bez výslovného povolenia od autora nemôžete odvádzať práce z týchto publikácií ani reprodukovávať, distribuovať a zobrazovať tieto publikácie ani žiadne ich časti.

S výnimkou ako je uvedené v týchto podmienkach, na publikácie ľubovoľné informácie, údaje, softvér alebo iné tu obsiahnuté intelektuálne vlastníctvo nemáte žiadne oprávnenia, licencie ani práva, vyjadrené ani implikované.

Spoločnosť IBM si vyhradzuje právo odobrať tu uvedené oprávnenia vždy, podľa vlastného uváženia, keď použitie týchto publikácií škodí autorovi, alebo ak spoločnosť IBM, že pokyny hore nie sú striktne dodržiavané.

Tieto informácie nemôžete prevziať ani exportovať okrem prípadu, ak to dovoľujú všetky aplikovateľné zákony a regulácie, vrátane všetkých zákonov a regulácií USA pre export.

IBM SA NEZARUČUJE ZA OBSAH TÝCHTO PUBLIKÁCIÍ. PUBLIKÁCIE SÚ POSKYTNUTÉ "TAK AKO SÚ" BEZ ZÁRUKY AKÉHOKOĽVEK DRUHU, VYJADRENEJ ALEBO IMPLIKOVANEJ, VRÁTANE (ALE NEOBMEDZENÉ) IMPLIKOVANÝCH ZÁRUK PREDAJNOSTI, NEPOŠKODENIA A VHODNOSTI NA KONKRÉTNY ÚČEL.









Vytlačené v USA