

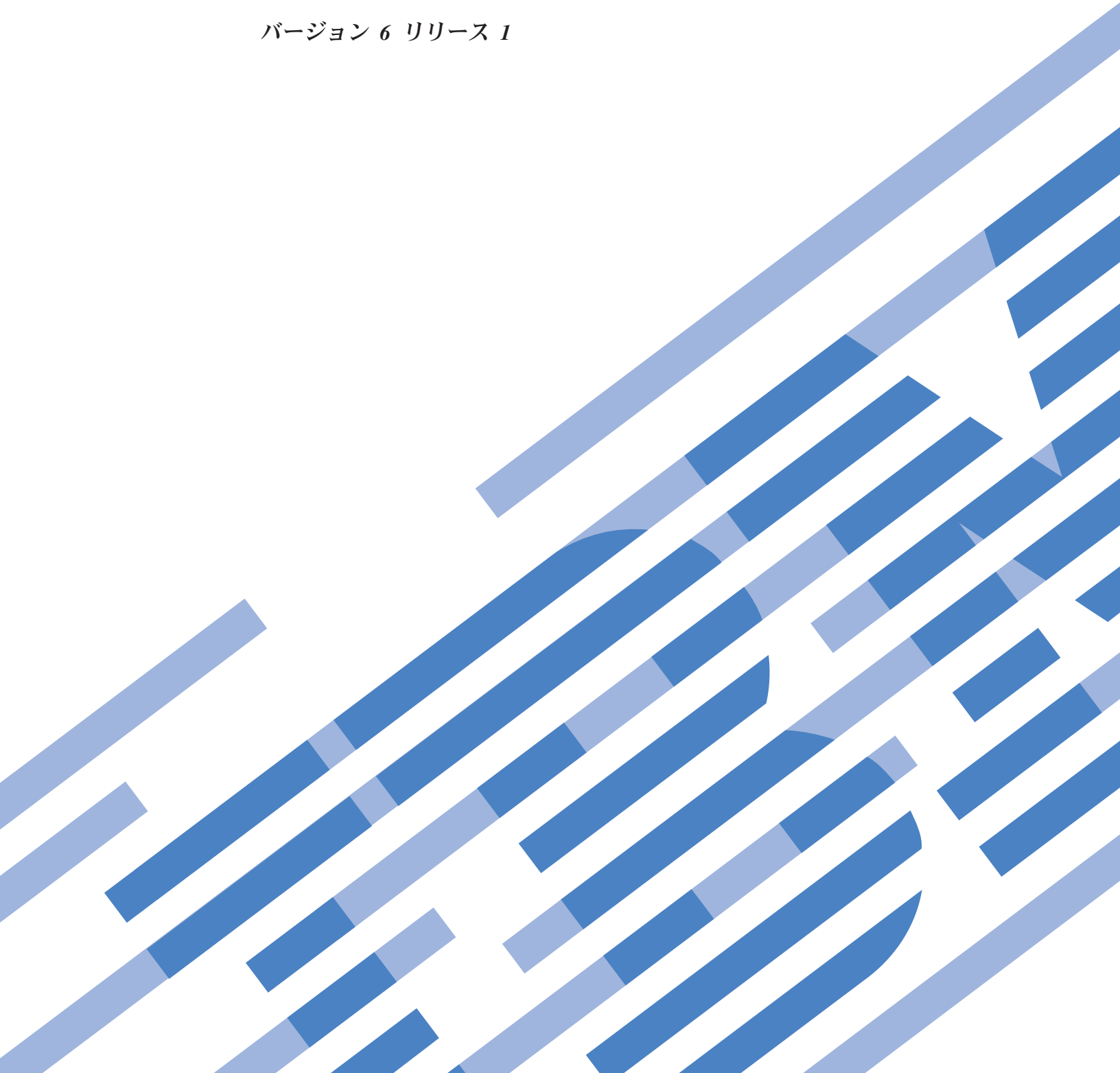


System i

可用性

タスク・ベース・アプローチによる
高可用性のインプリメント

バージョン 6 リリース 1





System i

可用性

タスク・ベース・アプローチによる
高可用性のインプリメント

バージョン 6 リリース 1

ご注意

本書および本書で紹介する製品をご使用になる前に、193 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM i5/OS (製品番号 5761-SS1) のバージョン 6、リリース 1、モディフィケーション 0 に適用されます。また、改訂版で断りがない限り、それ以降のすべてのリリースおよびモディフィケーションに適用されます。このバージョンは、すべての RISC モデルで稼働するとは限りません。また CISC モデルでは稼働しません。

IBM 発行のマニュアルに関する情報のページ

<http://www.ibm.com/jp/manuals/>

こちらから、日本語版および英語版のオンライン・ライブラリーをご利用いただけます。また、マニュアルに関するご意見やご感想を、上記ページよりお送りください。今後の参考にさせていただきます。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： System i
Implementing High Availability with the task-based
approach
Version 6 Release 1

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

目次

タスク・ベースのアプローチを使用した高

可用性のインプリメント 1

高可用性ソリューションの計画 2

アプリケーション回復力の計画 2

高回復力アプリケーションの識別 2

クラスタリング対応アプリケーションの i5/OS

アーキテクチャー 3

高可用性クラスター・アプリケーションの作成 3

アプリケーション・プログラムの回復力を高

める 4

可用性の高いクラスター・アプリケーション

の再始動 5

クラスター・リソース・グループ出口プログ

ラムの呼び出し 6

アプリケーション CRG の考慮事項 7

アプリケーション CRG による IP アドレス

のテークオーバーの管理 7

例: アプリケーション・クラスター・リソ

ース・グループのフェイルオーバー・アクシ

ョン 9

例: アプリケーション出口プログラム 10

データ回復力の計画 49

どのデータが回復されるべきかを判断する 49

切り替えディスクの計画 50

切り替えディスクのハードウェア要件 50

切り替えディスクのソフトウェア要件 51

切り替えディスクの通信要件 51

サイト間ミラーリングの計画 52

リモート・ミラーリングの計画 52

メトロ・ミラーの計画 57

グローバル・ミラーの計画 60

論理複製の計画 64

論理複製に使用するシステムの決定 64

クラスター・ミドルウェアの IBM ビジネ

ス・パートナーおよび使用可能なクラスタ

リング・プロダクト 65

論理複製のジャーナル処理計画 65

論理複製のバックアップ計画 65

論理複製のパフォーマンス計画 66

環境回復力の計画 66

クラスター管理可能ドメインの計画 66

モニター対象リソース項目 (MRE) の計画 67

クラスターの計画 67

クラスターのハードウェア要件 67

クラスターのソフトウェア要件 68

クラスターの通信要件 69

クラスターに使用するネットワークの専用

化 70

ヒント: クラスター通信 70

クラスターのパフォーマンス計画 71

複数リリース・クラスターの計画 73

クラスターのパフォーマンス計画 74

クラスターの計画チェックリスト 74

FlashCopy の計画 78

FlashCopy のハードウェア要件 78

FlashCopy のソフトウェア要件 78

FlashCopy の通信要件 79

高可用性のためのセキュリティ計画 79

クラスター全体の情報の配布 79

ファイアウォールを備えたクラスターの使用に

関する考慮事項 79

すべてのノード上のユーザー・プロファイルの

保守 80

高可用性の構成 81

シナリオ: 高可用性の構成 81

シナリオ: 論理区画間の切り替えディスク 81

シナリオ: システム間の切り替えディスク 82

シナリオ: リモート・ミラーリングによる切り

替えディスク 84

シナリオ: リモート・ミラーリングを使用する

サイト間ミラーリング 85

シナリオ: メトロ・ミラーリングを使用するサ

イト間ミラーリング 87

シナリオ: グローバル・ミラーを使用するサイ

ト間ミラーリング 88

高可用性のための TCP/IP のセットアップ 90

TCP/IP 構成属性の設定 91

INETD サーバーの開始 91

クラスターの構成 92

クラスターの作成 92

ノードをクラスターに追加できるようにす

る 93

ノードの追加 94

ノードの開始 94

デバイス・ドメインへのノードの追加 94

クラスター・リソース・グループ (CRG) の

作成 95

CRG の開始 99

メッセージ・キューの指定 100

切り替えの実行 101

ノードの構成 102

ノードの開始 103

ノードをクラスターに追加できるようにす

る 103

ノードの追加 104

デバイス・ドメインへのノードの追加 104

CRG の構成 105

CRG の開始 105

クラスター・リソース・グループ (CRG)

の作成 105

クラスター管理可能ドメインの構成 110

	クラスター管理可能ドメインの作成	110	モニター対象リソース項目の管理	147
	クラスター管理可能ドメインへのノードの		切り替えディスクの管理	163
	追加	111	ディスク・プールを使用不可にする	163
	クラスター管理可能ドメインの開始	112	ハードウェアを切り替え可能にする	164
	モニター対象リソースの同期	113	独立ディスク・プールの静止	166
	モニター対象リソース項目の追加	114	独立ディスク・プールの再開	166
	切り替えディスクの構成	114	サイト間ミラーリングの管理	167
	独立ディスク・プールの作成	115	リモート・ミラーリングの管理	167
	ミラー保護の開始	116	リモート・ミラーリングの中断	167
	ミラー保護の停止	116	リモート・ミラーリングの再開	168
	ディスク装置またはディスク・プールを追加		ミラー・コピーの切り離し	169
	する	117	ミラー・コピーの再接続	170
	現在の構成の評価	117	リモート・ミラーリングの構成解除	170
	ディスク・プールを使用可能にする	119	リモート・ミラーリング・プロパティの	
	サイト間ミラーリングの構成	120	変更	171
	リモート・ミラーリングの構成	120	メトロ・ミラー・セッションの管理	172
	メトロ・ミラー・セッションの構成	121	メトロ・ミラー・セッションの中断	172
	グローバル・ミラー・セッションの構成	122	メトロ・ミラー・セッションの再開	172
	高可用性の管理	122	メトロ・ミラー・セッションの削除	172
	シナリオ: 高可用性ソリューションの管理	122	メトロ・ミラー・プロパティの表示	173
	シナリオ: 高可用性環境でのバックアップの		グローバル・ミラーの管理	173
	実行	123	グローバル・ミラー・セッションの中断	173
	シナリオ: リモート・ミラーリング環境で		グローバル・ミラー・セッションの再開	173
	のバックアップの実行	123	グローバル・ミラー・セッションの削除	174
	シナリオ: FlashCopy の実行	124	グローバル・ミラー・セッション・プロパ	
	シナリオ: 高可用性環境でのオペレーティン		ティの変更	174
	グ・システムのアップグレード	124	FlashCopy の管理	174
	例: オペレーティング・システムのアップ		FlashCopy セッションの構成	175
	グレード	126	FlashCopy の更新	175
	シナリオ: 装置を高可用性にする	128	FlashCopy の再接続	175
	クラスターの管理	128	FlashCopy の再接続	176
	クラスターのクラスター・バージョンの調整	129	FlashCopy の切り離し	176
	クラスターの削除	130	FlashCopy の削除	176
	クラスター構成の表示	130	FlashCopy からのデータの復元	176
	クラスター構成の保管および復元	131	FlashCopy プロパティの変更	177
	クラスター状況のモニター	131	高可用性ソリューションのトラブルシューティング	177
	メッセージ・キューの指定	132	クラスターのトラブルシューティング	177
	クラスター構成解除チェックリスト	134	クラスターの問題が存在するかどうかの判別	177
	ノードの管理	135	クラスターに関する回復情報の収集	179
	ノード・プロパティの表示	135	クラスターの一般的な問題	180
	ノードの停止	135	区画エラー	182
	ノードの除去	135	1 次クラスター区画と 2 次クラスター区	
	デバイス・ドメインからのノードの除去	136	画の判別	183
	クラスター・リソース・グループ (CRG) の		区画ノードの失敗への変更	184
	管理	137	区画化されたクラスター管理可能ドメイン	185
	CRG 状況の表示	137	ヒント: クラスター区画	185
	CRG の停止	138	クラスターの回復	186
	CRG の削除	139	クラスター・ジョブ障害からの回復	186
	切り替え可能装置の作成	139	損傷を受けたクラスター・オブジェクトの	
	CRG のリカバリー・ドメインの変更	140	回復	187
	フェイルオーバー停止イベントの管理	141	完全なシステム消失後のクラスターの回復	188
	クラスター管理可能ドメインの管理	144	災害後のクラスターの回復	188
	クラスター管理可能ドメインの停止	145	バックアップ・テープからのクラスターの	
	クラスター管理可能ドメインの削除	146	復元	188
	クラスター管理可能ドメインのプロパティ		サイト間ミラーリングのトラブルシューティング	189
	ーの変更	146	リモート・ミラーリング・メッセージ	189

付録. 特記事項	193	商標	195
Ⅰ プログラミング・インターフェース情報	195	使用条件	195

タスク・ベースのアプローチを使用した高可用性のインプリメント

i5/OS® の高可用性を構成および管理するタスク・ベースのアプローチは、ビジネスのニーズに基づいてカスタマイズされた高可用性ソリューションの構成および管理を可能にします。高可用性ソリューションの構成および管理では、グラフィカルおよびコマンド行インターフェースを使用します。

ソリューション・ベースのアプローチでは、High Availability Solution Manager のグラフィカル・インターフェースを使用して、制限されたユーザー入力を元に定義済みのソリューションが自動的に構成されます。対照的に、タスク・ベースのアプローチでは、知識の豊富なユーザーが個別設定したソリューションをカスタマイズおよびインプリメントするための手段を提供します。ただし、このアプローチを使用して高可用性ソリューションを作成および管理する場合、ユーザーは、高可用性のニーズに精通していること、およびいくつかのインターフェースを使い慣れていることが要求されます。

クラスター・リソース・サービスのグラフィカル・インターフェース

クラスター・リソース・サービスのインターフェースで、高可用性ソリューションに不可欠なクラスター・テクノロジーの構成および管理を行うことができます。このインターフェースを使用するには、IBM® System i™ High Availability Solutions Manager (iHASM) ライセンス・プログラム5761-HAS がインストールされていることが必要です。このインターフェースを使用すると、以下の機能を実行することができます。

- クラスターの作成と管理
- ノードの作成と管理
- クラスター・リソース・グループの作成と管理
- クラスター管理可能ドメインの作成と管理
- モニター対象リソースの作成と管理
- クラスター区画やフェイルオーバーなど、クラスターに関連したイベントをクラスターでモニター
- システムに対する定期保守など、計画停止の際に手動切り替えを実行

ディスク管理インターフェース

ディスク管理インターフェースは、いくつかのデータ回復テクノロジーのインプリメント時に必要な独立ディスク・プールの構成および管理を可能にします。インプリメントされたデータ回復テクノロジーのタイプにより、以下の機能の一部を使用する場合にインストール要件が必要な場合があります。

- ディスク・プールの作成
- ディスク・プールを使用可能にする
- ディスク・プールを使用不可にする
- リモート・ミラーリングの構成
- メトロ・ミラーの構成
- グローバル・ミラーの構成

コマンド行インターフェース

コマンド行インターフェースは、CL コマンドを使用して多種多様な高可用性タスクの実行を可能にします。クラスターに関連した各タスクには、CL コマンドが対応しています。

関連情報

IBM System i High Availability Solutions Manager (iHASM) コマンド

高可用性ソリューションの計画

i5/OS 高可用性ソリューションを構成する前に、ソリューションが要求するすべての要件が満たされるよう、適切な計画を立てる必要があります。

各高可用性ソリューション・テクノロジーには、特定のソリューションを構成する前に満たすべき、最小必要要件があります。これらの要件に加えて、回復力をもたせるリソースをどれにするか決定することも重要です。アプリケーション、データ、およびデバイスなど、これらのリソースは、高可用性にするかどうかを判断するために評価を行う必要があります。それらのリソースで高可用性が要求される場合は、環境に対して必要な変更を加えてから高可用性ソリューションを構成することが重要です。例えば、高可用性にするべきデータが SYSBAS にあるとします。その場合は、ソリューションを構成する前に、そのデータを独立ディスク・プールに移動します。高可用性を使用可能にするには、アプリケーションの変更が必要な場合もあります。

アプリケーション回復力の計画

アプリケーション回復力とは、高可用性環境の重要な要素の 1 つです。可用性の高いアプリケーションを、クラスター内で構築して使用する場合、これらのアプリケーションには特定の可用性仕様がある点に留意しなければなりません。

回復力のあるアプリケーションをご使用の環境で利用することにより、クライアントを再構成しなくても、別のクラスター・ノードで再始動できます。さらに、アプリケーションに関連したデータが、切り替えまたはフェイルオーバー後も使用可能です。これは、アプリケーションのユーザーが、アプリケーションとそのデータがプライマリー・ノードからバックアップ・ノードに切り替わる間に経験する中断が最小になる、またはほとんど気付かれないことを意味しています。ユーザーは、アプリケーションとデータがバック・エンドで移動したことを意識する必要がありません。

クラスターでアプリケーションの回復を実現するには、特定の仕様に合致したアプリケーションを使用しなければなりません。アプリケーションが切り替え可能になり、それゆえクラスターのユーザーにとって常に使用可能であるためには、アプリケーションに特定の特性が備わっていなければなりません。これらの要件があるため、切り替え可能アプリケーションをクラスターに使用する際には、以下のようないくつかのオプションがあります。

1. クラスター使用可能ソフトウェア・アプリケーションを購入する

クラスター使用可能のソフトウェア・プロダクトは、特定の高可用性要件に合致しています。

2. 高可用性を持つように、ユーザーのアプリケーションを作成または変更する

- | 独立系ソフトウェア・ベンダーおよびアプリケーション・プログラマーは、i5/OS 高可用性環境で切り
- | 替え可能になるように、アプリケーションをカスタマイズすることができます。

回復アプリケーションを手に入れたなら、クラスター内で管理する必要があります。

関連情報

高可用性およびクラスター (英語)

高回復力アプリケーションの識別

すべてのアプリケーションが、クラスターリングを使用できることを有効に利用できるわけではありません。

クラスタリングにより提供される切り替えおよびフェイルオーバー機能を利用するためには、アプリケーションは回復できなければなりません。アプリケーション回復力により、アプリケーションを使用するクライアントを再構成しなくても、アプリケーションをバックアップ・ノードで再始動できます。そのため、クラスタリングにより提供される機能を十分活用するためには、使用するアプリケーションは特定の要件を満たしていなければなりません。

クラスタリング対応アプリケーションの i5/OS アーキテクチャー

エンド・ユーザーにとって利用価値が高いのは、計画された停止または予期せぬ停止が発生したときにも引き続き使用可能であるアプリケーションを認識する、高可用性を備えたアプリケーションです。

i5/OS には、レベルの異なるさまざまな高可用性アプリケーションをサポートする、アプリケーション回復力に対応したアーキテクチャーが提供されています。この領域におけるハイエンドなアプリケーションは、高い可用性を誇り、高可用性環境を自動的に構築します。これらのアプリケーションの管理は、高可用性管理インターフェースを通じて行います。

これらのアプリケーションには、以下の特性があります。

- プライマリー・ノードが使用できなくなったとき、アプリケーションがバックアップ・クラスター・ノードに切り替えることが可能。
- アプリケーションにおいて回復定義および状況データ域に回復環境の定義がなされており、クラスター管理アプリケーションによりアプリケーションの自動構成と活動化が行える。
- アプリケーションが、クラスター関連イベントをハンドルするアプリケーション CRG 出口プログラムによりアプリケーション回復力を提供し、i5/OS クラスター・リソース・サービスの機能を利用している。
- アプリケーションは、ユーザーにアプリケーション・メニュー画面またはそれより進んだ画面を表示するアプリケーション再始動機能を提供する。

より厳格な可用性および再始動特性を備えたアプリケーションには、以下の特性があります。

- アプリケーションが、アプリケーション CRG 出口プログラムにより、クラスター・イベント (アクション・コード) のさらに強力なハンドリングを通して、拡張アプリケーション回復力を提供する。
- アプリケーションが、さらに高いレベルのアプリケーション再始動サポートを提供する。ホスト中心のアプリケーションの場合、コミットメント制御またはチェックポイント機能によりトランザクション境界の状態へ戻します。クライアント中心のアプリケーションの場合、最小のサービスの中断だけでシームレスにフェイルオーバーします。

高可用性クラスター・アプリケーションの作成

高可用性アプリケーションとは、クラスター環境におけるシステム障害で回復できるものを言います。

いくつかのレベルのアプリケーション可用性が可能です。

1. アプリケーション・エラーが発生した場合、同じノードでアプリケーション自身が再始動しエラーの潜在的な原因 (壊れた制御データなど) を訂正する。まるでアプリケーションが初めて開始されたように見えます。
2. アプリケーションは、ある程度のチェックポイント・リスタート処理を行う。アプリケーションが障害時点に近づいたかのように見えます。
3. システム障害が発生した場合、アプリケーションはバックアップ・サーバーで再始動する。まるでアプリケーションが初めて開始されたように見えます。
4. システム障害が発生した場合、アプリケーションはバックアップ・サーバーで再始動し、ポイント・リスタート処理を行う。アプリケーションが障害時点に近づいたかのように見えます。

5. システム障害が発生した場合、アプリケーションと関連データ双方で、クラスター内の 1 つ以上の別のノードへの整合フェイルオーバーを実行する。まるでアプリケーションが初めて開始されたように見えます。
6. システム障害が発生した場合、アプリケーションと関連データ双方で、クラスター内の 1 つ以上の別のノードへの整合フェイルオーバーを実行する。アプリケーションは、複数のサーバーである程度のチェックポイント・リスタート処理を行う。アプリケーションが障害時点に近づいたかのように見えます。

注: 上記の 1 から 4 の場合、データの回復はユーザーの責任になります。

アプリケーション・プログラムの回復力を高める:

アプリケーション・プログラムの回復力を高める方法について説明します。

回復力の高いアプリケーションには、次のような特性があると考えられます。

- そのアプリケーションは、現在のノードまたは別のノードで再始動できる。
- そのアプリケーションは、IP アドレスを使用することによって、クライアントにアクセスできる。
- そのアプリケーションには状態情報がない (ステートレス)、あるいは状態情報が明らかになっている。
- そのアプリケーションに関連したデータは、切り替え後も使用できる。

クラスター環境内でシステム障害が発生した場合に備えて、アプリケーションを回復力に富むものにしておくのに重要な 3 つの要素は、次のとおりです。

アプリケーションそのもの

そのアプリケーションはエラーもしくはシステム障害に対して、どれほどの耐久性を備えているでしょうか? そのアプリケーションはどれほど容易に再始動できるでしょうか?

このことは、アプリケーションにおいてクラスタリング機能を使用することによって処理できます。

関連データ

障害が発生した場合、関連データの可用性に影響しますか?

- | 重要なデータを切り替えディスクに保管することで、障害時にもデータを使用可能な状態に保つことができます。また、クラスター・ミドルウェアである IBM ビジネス・パートナーの複製プロダクトは、クラスタリング機能を効果的に活用できるため、これによってこの問題を処理することもできます。

制御機能と管理

データやアプリケーションの可用性をサポートする環境を、どれほど容易に定義できますか?

IBM System i High Availability Solutions Manager (iHASM)、ライセンス・プログラム番号 (5761-HAS) では、さまざまなインターフェースを使用して、高可用性ソリューションおよびテクノロジーの構成および管理をすることができます。iHASM ライセンス・プログラムでは、以下のインターフェースが提供されています。

高可用性ソリューション・マネージャーのグラフィカル・インターフェース

このグラフィカル・インターフェースにより、i5/OS 対応の高可用性ソリューションの中から希望するものを選択できます。このインターフェースは、選択したソリューションのテクノロジー要件をすべて検証し、選択したソリューションおよび関連するテクノロジーを構成するため、ソリューションを構成する高可用性テクノロジーすべてを簡単に管理できるようになります。

クラスター・リソース・サービスのグラフィカル・インターフェース

経験のあるユーザーは、このグラフィカル・インターフェースを使用することで、高可用

性ソリューションを柔軟にカスタマイズすることができます。これにより、CRG などのクラスター・テクノロジーの構成および管理が可能になります。また、高可用性ソリューションの一部として使用されている独立ディスク・プールであれば、このインターフェースから構成することもできます。

IBM System i 高可用性ソリューション・マネージャーのコマンド

これらのコマンドでも同様の機能が提供されていますが、コマンド行インターフェースを通じて使用します。

IBM System i 高可用性ソリューション・マネージャー (iHASM) API

これらの API により、独立ディスク・プールの新規機能进行处理することができます。

他にも、サード・パーティー製のクラスター管理インターフェースによって、クラスタリング API を使用し、回復アプリケーションと回復データを組み合わせることで、この問題に対処することもできます。

関連情報

高可用性の管理 (英語)

可用性の高いクラスター・アプリケーションの再始動:

アプリケーションを再始動するには、アプリケーションは、フェイルオーバーまたは切り替えの際のアプリケーションの状態を把握する必要があります。

状態情報はアプリケーションに特有のもので、アプリケーションは必要な情報を判別する必要があります。状態情報が全くなくても、アプリケーションは PC で再始動できます。しかし、その場合、アプリケーション内の位置を再確立しなければなりません。

バックアップ・システム用にアプリケーションの状態情報を保管するのに、使用できる方法が何通りかあります。各アプリケーションは、どの方法が最もよく機能するかを判別する必要があります。

- アプリケーションはすべての状態情報を、要求を出しているクライアント・システムに転送できます。切り替えまたはフェイルオーバーが発生すると、アプリケーションはクライアント上に保管されている状態を使用して、新しいサーバーに状態を再確立します。これは、情報配布 API またはクラスター・ハッシュ・テーブル API を使用すると行えます。
- アプリケーションは、状態情報 (ジョブ情報およびアプリケーションに関連した他の制御構造など) をリアルタイムで複製できます。構造内でのすべての変更に関して、アプリケーションはバックアップ・システムにその変更内容を送ります。
- アプリケーションはアプリケーションに関連付けられた適切な状態情報を、アプリケーション用のクラスター・リソース・グループの出口プログラムのデータ部分に保管できます。この方法は、必要な状態情報の量が少ないことが前提です。これを実行するには、クラスター・リソース・グループ変更 (QcstChangeClusterResourceGroup) API を使用できます。
- アプリケーションは、アプリケーションのデータと共に、バックアップ・システムに複製されたデータ・オブジェクト内の状態情報を保管できます。
- アプリケーションは、切り替え可能な IASP に含まれるデータ・オブジェクト内の情報を保管できます。その IASP にはアプリケーションのデータも含まれています。
- アプリケーションは、クライアントの状態情報を保管できます。
- 状態情報は全く保管されずに、回復を実行する必要があります。

注: アプリケーションがある種のチェックポイント・リスタート処理を使用する場合には、保管が必要な情報が少なくなります。状態情報は、事前に決めたアプリケーション・チェックポイントでのみ保管されます。再始動すると、最後に使用したチェックポイントまで戻ります。これは、データベースのコミットメント制御処理と同様の機能です。

クラスター・リソース・グループ出口プログラムの呼び出し:

クラスター・リソース・グループ出口プログラムは、クラスター環境の様々な局面で呼び出されます。

このプログラムはクラスター内のリソースに対し、環境として必要な回復性を確立します。出口プログラムは、回復装置 CRG に対してはオプションとして設定できますが、他の CRG タイプに対しては必須です。クラスター・リソース・グループ出口プログラムが使用されている場合に、以下のようなクラスター全体に関わるイベントが発生すると、そのプログラムが呼び出されます。

- 予期せぬ形で、ノードがクラスターからはずされた場合
- クラスター・ノード終了 (QcstEndClusterNode) API またはクラスター・ノード項目除去 (QcstRemoveClusterNodeEntry) API を呼び出した結果として、ノードがクラスターからはずされた場合
- クラスター削除 (QcstDeleteCluster) API を呼び出した結果として、クラスターが削除された場合
- クラスター・ノード開始 (QcstStartClusterNode) API を呼び出したことによって、ノードが活動化された場合
- 区画化されたノードとの通信が再確立された場合

出口プログラムは、以下のプロセスを完了します。

- 名前付き活動化グループまたは呼び出し側の活動化グループ (*CALLER) で実行されます。
- 出口プログラムで処理できない例外が生じるか、出口プログラムが取り消される場合には、再始動パラメーターを無視します。
- 取り消しハンドラーを提供します。

クラスター・リソース・グループ API が実行されると、出口プログラムは、クラスター・リソース・グループ作成 (QcstCreateClusterResourceGroup) API で指定されたユーザー・プロファイルと共に、別個のジョブから呼び出されます。その別個のジョブは、出口プログラムが呼び出されると、API により自動的に作成されます。データ CRG の出口プログラムがうまく作動しなかった場合、または異常終了した場合には、取り消しのアクション・コードを使用して、リカバリー・ドメインのすべてのアクティブ・ノードで、クラスター・リソース・グループ出口プログラムが呼び出されます。このアクション・コードによって、終了していないすべての活動がバックアウトされ、クラスター・リソース・グループの元の状態を回復できます。

装置 CRG に対して、異常な切り替えが発生したと仮定します。すべての装置の切り替えを戻した後に、すべての装置が元のプライマリー・ノードで正常にオンに変更された場合、クラスタリングは開始のアクション・コードを使用して、元のプライマリー・ノード上で出口プログラムを呼び出します。

アプリケーション CRG の出口プログラムがうまく作動しなかった場合、または異常終了した場合には、その CRG の状態がアクティブであれば、クラスター・リソース・サービスはアプリケーションを再始動しようとします。クラスター・リソース・グループ出口プログラムは、再始動のアクション・コードを使用して呼び出されます。指定された最大回数の試行でもアプリケーションを再始動できない場合には、フェイルオーバーのアクション・コードを使用して、クラスター・リソース・グループ出口プログラムが呼び出されます。開始のアクション・コードを使用して出口プログラムが呼び出された場合 (CRG の開始、フェイルオーバー、または切り替えの結果として生じることがあります) のみ、再始動カウントがリセットされません。

クラスター・リソース・グループが開始されると、プライマリー・ノードで呼び出されるアプリケーション CRG 出口プログラムは、アプリケーションが終了するか、エラーが生じるまでは、クラスター・リソース・サービスに制御を戻しません。アプリケーション CRG がアクティブになった後、クラスター・リソース・サービスがアプリケーション CRG 出口プログラムにイベントを通知する必要がある場合には、出口プログラムの別のインスタンスが異なるジョブで開始されます。開始または再始動を除くアクション・コードは、戻されることが予想されます。

クラスター・リソース・グループ出口プログラムが呼び出されると、処理中のクラスター・イベント、クラスター・リソースの現在の状態、およびクラスター・リソースの期待される状態を識別するパラメーターの集合が渡されます。

クラスター・リソース・グループ出口プログラムに関する完全な情報は、クラスター API の資料のクラスター・リソース・グループ出口プログラムを参照してください。アクション・コードごとに、クラスター・リソース・グループ出口プログラムにどのような情報が渡されるのかについても説明されています。出口プログラムを作成する際の基礎として使用できるサンプル・ソース・コードが、QUSRTOOL ライブラリーで提供されています。QATTSYSC ファイルの TCSTAPPEXT メンバーを参照してください。

アプリケーション CRG の考慮事項

アプリケーション・クラスター・リソース・グループは、アプリケーション面での回復を管理します。

アプリケーション CRG による IP アドレスのテークオーバーの管理:

クラスター・リソース・サービスを使用して、アプリケーション CRG による IP アドレスのテークオーバーを管理することができます。この管理は手動で行うこともできます。

アプリケーション CRG に関連付けられたアプリケーション・テークオーバー IP アドレスは、2 とおりの方法で管理できます。最も簡単な方法がデフォルトになっていますが、それはクラスター・リソース・サービスに テークオーバー IP アドレスを管理させる方法です。この方法では、リカバリー・ドメイン内の全ノード (リカバリー・ドメインに後で追加されるノードも含む) にテークオーバー IP アドレスを作成するよう、クラスター・リソース・サービスに指示が出されます。この方式が選択されると、その時点で テークオーバー IP アドレスはリカバリー・ドメイン内のすべてのノードで定義できなくなります。

別の方法は、テークオーバー IP アドレスをユーザー自身が管理します。この方法では、クラスター・リソース・サービスは テークオーバー IP アドレスを構成するいかなるステップも実行しないよう指示されます。ユーザーが構成に責任を持ちます。クラスター・リソース・グループを開始する前に、リカバリー・ドメイン内の全ノード (複製ノードは除く) にテークオーバー IP アドレスを追加する必要があります。活動 CRG のリカバリー・ドメインに追加される任意のノードは、追加される前に テークオーバー IP アドレスを構成しておかなければなりません。

関連概念

9 ページの『例: アプリケーション・クラスター・リソース・グループのフェイルオーバー・アクション』

この例は、あるフェイルオーバーのシナリオを示したものです。フェイルオーバーの別のシナリオは、これとは違う動作になる場合があります。

複数サブネット: デフォルトではすべてのリカバリー・ドメイン・ノードは同一のサブネット上にありますが、アプリケーション・テークオーバー IP アドレスを複数のサブネットで機能させることが可能です。リカバリー・ドメイン内のノードを複数のサブネットに広げる際に、アプリケーション・テークオーバー IP アドレスを構成するには、切り替え環境を使用可能にする必要があります。

サブネットをまたいだアプリケーション切り替えを使用可能にする:

一般的に、クラスタリングを行うには、アプリケーション・クラスター・リソース・グループのリカバリー・ドメインのクラスター・ノードが、すべて同じ LAN 上に存在している (同じサブネット・アドレス指定を使用している) 必要があります。クラスター・リソース・サービスは、アプリケーションの CRG の構成時に、ユーザーが構成したテークオーバー IP アドレスをサポートします。

構成されたアプリケーションのテークオーバー IP アドレスを、リカバリー・ドメインのあるノードから別のノードへと切り替える際の基盤として使用されるネットワーク・プロトコルは、アドレス解決プロトコル (ARP) です。しかし、リカバリー・ドメインを拡張して、業務用ルーターで区切られた他の LAN に存在するクラスター・ノードをそこに含めることが可能です。この拡張を行うには、仮想 IP アドレスのサポートを利用して、クラスター・ノードおよびネットワークの業務用ルーターで Routing Information Protocol (RIP) を使用します。

以下の手動構成ステップが、切り替え環境を使用可能にするために必要とされます。この一連の指示は、リカバリー・ドメイン内のすべてのノードにおいて行わなければなりません。また、指定されたアプリケーション CRG のリカバリー・ドメイン内のノードとなる、クラスター内の他のノードに対しても、繰り返し行う必要があります。

1. アプリケーション CRG の使用するテークオーバー IP アドレスを選択します。
 - 混乱を避けるために、このアドレスは、クラスター・ノードまたはルーターによって使用される他の既存アドレスとオーバーラップすべきではありません。たとえば、19.19.19.19 を選択した場合は、19.0.0.0 (19.19.0.0) をシステム・ルーティング・テーブルによって認識されるルートにはしないでください。
 - テークオーバー・インターフェース (例えば、19.19.19.19) を追加します。これを作成する際には、回線記述を *VIRTUALIP、サブネット・マスクを 255.255.255.255 (ホスト経路)、最大伝送単位を 1500 (576 から 16388 の任意の数)、および自動始動を *NO に指定します。このテークオーバー・アドレス (たとえば 19.19.19.19) は、次のステップで関連ローカル・インターフェースとしてそれを識別する前に、*VIRTUALIP アドレスとして存在している必要があります。しかし、アクティブである必要はありません。
2. クラスターを作成する場合、またはクラスターにノードを追加する場合に、目的のテークオーバー IP アドレスを、クラスター通信で使用されるよう指定した IP アドレスのいずれかまたは両方と関連付けます。
 - この例の場合、テークオーバー・アドレス 19.19.19.19 を、その IP アドレス上の関連ローカル・インターフェースにすることで、イーサネット・バスのクラスター・ノードを、ローカルでのクラスタリングに使用できるようにすることを意味します。このことは、各クラスター・ノード上のそれぞれのクラスター・アドレスごとに行う必要があります。

注: この変更内容を CFGTCP 下で有効にするには、クラスター・アドレスを終了する必要があります。

3. クラスターを作成し、CRG を作成します。アプリケーション CRG の場合、「テークオーバー IP アドレスの構成」フィールドに QcstUserCfgsTakeoverIpAddr を指定します。どのアプリケーション CRG も開始しないでください。
4. CFGTCP 下の「TCP/IP アプリケーションの構成 (Configure TCP/IP Applications)」(オプション 20) を使用した後、「RouteD の構成 (Configure RouteD)」(オプション 2)、さらに「RouteD 属性の変更 (Change RouteD attributes)」(オプション 1) を使用して、「供給 (Supply)」が *YES に設定されていることを確認します。設定されていない場合は、*YES に設定し、それぞれのクラスター・ノード上で ROUTED (RIP または RIP-2) を開始または再開します。

- NETSTAT オプション 3 は、現在実行中であれば、ローカル・ポートを使用して ROUTED を表示します。ROUTED は、CRG リカバリー・ドメイン内のすべてのクラスター・ノードで経路を実行し公示している必要があります (Supply = *YES)。
5. リカバリー・ドメイン LAN を相互接続しているネットワーク内のすべての業務用ルーターにおいて、RIP のホスト経路を受け入れて、公示するようにしてください。
 - これは必ずしもルーター用のデフォルト設定ではありません。ルーターの製造元によって言語は異なりますが、RIP インターフェース下では、その言語によってホスト経路を送信し、動的ホストを受信するものと想定されています。
 - このことは、システムを指すルーター・インターフェースと、ルーター間インターフェースの両方にも当てはまります。

注: この構成では、System i マシンをルーターとして使用しないでください。ルーティングの目的で設計された業務用ルーター (IBM またはそれ以外のもの) を使用してください。この機能を処理するように System i ルーティングを構成することはできません。

6. クラスター・ノードのいずれかにおいて、テークオーバー・アドレスを手動でアクティブにします。
 - a. RIP が経路を伝搬するまで待ちます (約 5 分)。
 - b. CRG リカバリー・ドメイン内のすべてのノードから、またこのアドレスを使用する LAN 上で選択したクライアントから、テークオーバー・アドレスを ping します。
 - c. テークオーバー・アドレスが再度終了していることを確認してください。
(CRG の開始時に、指定されたプライマリー・ノード上のアドレスが、クラスタリングによって開始されます。)
7. アプリケーション CRG を開始します。
 - 指定された優先ノード上でのクラスタリングによって、テークオーバー・アドレスが開始され、RIP がリカバリー・ドメイン全体で経路を公示します。RIP がドメイン全体で経路を更新するのに、5 分ほどかかります。RIP 機能は、CRG の開始機能とは独立しています。

重要:

- アプリケーション CRG リカバリー・ドメイン内のすべてのクラスター・ノードで上記の手順が実行されない場合には、切り替えプロセス中にクラスターが停止します。
- レプリカ・ノードへのフェイルオーバーを実行していない場合でも、レプリカ・ノードを後で変更してバックアップを作成する際に、レプリカ・ノードでこの作業を行っておくことをお勧めします。
- 複数の仮想 IP アドレスを使用する場合には、それぞれのアドレスに、別個のアプリケーション CRG と、それらに関連付けられる別個の IP アドレスが必要です。このアドレスは、同じ物理アダプター上の別の論理 IP アドレスであっても、まったく別の物理アダプターであってもかまいません。また、ルーティング・テーブル内であいまいさがないようにするために注意が必要です。このことは、次のように行うことによって、最もよく達成されます。
 - それぞれの仮想 IP アドレスのルーティング・テーブルに *DFTRROUTE を追加します。
 - 複数の IP アドレスを使用するには、CFGTCP (オプション 2) を使用します。
 - すべてのパラメーター (ネクスト・ホップを含む) を同じように設定して、選択したルーターに到達するように設定します。ただし、優先バイインディグ・インターフェースは、この経路によって表される仮想 IP アドレスに関連したローカル・システム IP アドレスに設定する必要があります。

例: アプリケーション・クラスター・リソース・グループのフェイルオーバー・アクション:

この例は、あるフェイルオーバーのシナリオを示したものです。フェイルオーバーの別のシナリオは、これとは違う動作になる場合があります。

再試行限界を超過したり、ジョブが取り消されたりしたために、回復アプリケーションのクラスター・リソース・グループがフェイルオーバーすると、以下のことが発生します。

- CRG のリカバリー・ドメイン内にあるすべてのアクティブ・ノードで、フェイルオーバーのアクション・コードによってクラスター・リソース・グループ出口プログラムが呼び出されます。このことは、クラスター・リソース・サービスが、アプリケーションのアクセス・ポイントを最初のバックアップにフェイルオーバーする準備を進めていることを意味します。
- クラスター・リソース・サービスが、プライマリー・ノード上のテークオーバー・インターネット・プロトコル (IP) 接続を終了します。
- クラスター・リソース・サービスが、最初のバックアップ・ノード (新しいプライマリー・ノード) 上でテークオーバー IP 接続を開始します。
- クラスター・リソース・サービスが、新しいプライマリー・ノード上でのみ開始のアクション・コードによって、クラスター・リソース・グループ出口プログラムを呼び出すジョブを投入します。このアクションによって、アプリケーションが再始動します。

関連概念

7 ページの『アプリケーション CRG による IP アドレスのテークオーバーの管理』
クラスター・リソース・サービスを使用して、アプリケーション CRG による IP アドレスのテークオーバーを管理することができます。この管理は手動で行うこともできます。

例: アプリケーション出口プログラム:

このコード例には、アプリケーション・クラスター・リソース・グループ出口プログラムが含まれています。

このコード例は QUSRTOOL ライブラリーにあります。

注: このコード・サンプルを使用すると、190 ページの『コードに関するライセンス情報および特記事項』の条件に同意したことになります。

```
/* **** */
/*
/* Library:   QUSRTOOL
/* File:     QATTSYSC
/* Member:   TCSTAPPEXT
/* Type:     ILE C
/*
/* Description:
/* This is an example application CRG exit program which gets called for
/* various cluster events or cluster APIs. The bulk of the logic must
/* still be added because that logic is really dependent upon the unique
/* things that need to be done for a particular application.
/*
/* The intent of this example to to provide a shell which contains the
/* basics for building a CRG exit program. Comments throughout the example
/* highlight the kinds of issues that need to be addressed by the real
/* exit program implementation.
/*
/* Every action code that applies to an application CRG is handled in this
/* example.
/*
/* The tcstdtaara.h include is also shipped in the QUSRTOOL library. See
/* the TCSTDTAARA member in the QATTSYSC file.
/*
/* Change log:
/*
```

```

/* Flag Reason Ver Date User Id Description */
/* ----- */
/* ... D98332 v5r1m0 000509 ROCH Initial creation. */
/* $A1 P9950070 v5r2m0 010710 ROCH Dataarea fixes */
/* $A2 D99055 v5r2m0 010913 ROCH Added CancelFailover action code */
/* $A3 D98854 v5r2m0 010913 ROCH Added VerificationPhase action code*/
/* $A4 P9A10488 v5r3m0 020524 ROCH Added example code to wait for data*/
/* CRGs on switchover action code */
/* ----- */
/*****

/*-----*/
/*
/* Header files
/*
/*-----*/
#include /* Useful when debugging */
#include /* offsetof macro */
#include /* system function */
#include /* String functions */
#include /* Exception handling constants/structures */
#include /* Various cluster constants */
#include /* Structure of CRG information */
#include "qusrtool/qatssys/tcstdtaara" /* QCSTHAAPPI/QCSTHAAPPO data areas*/
#include /* API to Retrieve contents of a data area */
#include /* API error code type definition */
#include /* mitime builtin */
#include /* waittime builtin */

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

/*-----*/
/*
/* The following constants are used in the checkDependCrgDataArea()
/* function. The first defines how long to sleep before checking the data
/* area. The second defines that maximum time to wait for the data area
/* to become ready before failing to start the application when the Start
/* CRG function is being run. The third defines the maximum wait time for
/* the Initiate Switchover or failover functions.
/*
/*-----*/
#define WaitSecondsIncrement 30
#define MaxStartCrgWaitSeconds 0
#define MaxWaitSeconds 900

/*-----*/
/*
/* As this exit program is updated to handle new action codes, change the
/* define below to the value of the highest numbered action code that is
/* handled.
/*
/*-----*/
#define MaxAc 21

/*-----*/
/*

```

```

/* If the exit program data in the CRG has a particular structure to it, */
/* include the header file for that structure definition and change the */
/* define below to use that structure name rather than char. */
/* */
/*-----*/
#define EpData char

/*-----*/
/* */
/* Change the following define to the library the application resides in */
/* and thus where the QCSTHAAPPO and QCSTHAAPPI data areas will be found. */
/* */
/*-----*/
#define ApplLib "QGPL"

/*-----*/
/* */
/* Prototypes for internal functions. */
/* */
/*-----*/
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 setAppLCrgDataArea(char);
#pragma argopt(setAppLCrgDataArea)
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)

/*-----*/
/* */
/* Some debug routines */
/* */

```

```

/*
/*-----*/
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);

/*-----*/
/*
/*
/* Type definitions
/*
/*
/*-----*/

/*-----*/
/*
/*
/* This structure defines data that will be passed to the exception and
/* cancel handlers. Extend it with information unique to your application.*/
/*
/*
/*-----*/
typedef struct {
    int *retCode;           /* Pointer to return code
    EpData *epData;        /* Exit program data from the CRG
    Qcst_EXTP0100_t *crgData; /* CRG data
    unsigned int actionCode; /* The action code
    int role;              /* This node's recovery domain role
    int priorRole;         /* This node's prior recovery domainrole
} volatile HandlerDataT;

/*-----*/
/*
/*
/* Function pointer array for handling action codes. When the exit program*/
/* is updated to handle new action codes, add the new function names to
/* this function pointer array.
/*
/*
/*-----*/
static int (*fcn[MaxAc+1]) (int role,
                           int priorRole,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) = {
    newActionCode, /* 0 - currently reserved */
    createCrg,    /* 1 */
    startCrg,    /* 2 */
    restartCrg,  /* 3 */
    endCrg,      /* 4 */
    verifyPhase, /* 5 - currently reserved */
    newActionCode, /* 6 - currently reserved */
    deleteCrg,   /* 7 */
    memberIsJoining, /* 8 */
    memberIsLeaving, /* 9 */
    switchPrimary, /* 10 */
    addNode,     /* 11 */
    rmvNode,     /* 12 */
    chgCrg,      /* 13 */
    deleteCrgWithCmd, /* 14 */
    undoPriorAction, /* 15 */
    endNode,     /* 16 */
    newActionCode, /* 17 - applies only to a device CRG */
    newActionCode, /* 18 - applies only to a device CRG */
    newActionCode, /* 19 - applies only to a device CRG */
    chgNodeStatus, /* 20 */
    cancelFailover /* 21 */
};

```

```

/*-----*/
/*
/* Function pointer array for handling prior action codes when called with */
/* the Undo action code. When the exit program is updated to handle */
/* Undo for new action codes, add the new function names to this function */
/* pointer array. */
/*
/*-----*/
static int (*undoFcn[MaxAc+1]) (int role,
                                int priorRole,
                                Qcst_EXTP0100_t *crgData,
                                EpData *epData) = {
    newActionCode,      /* 0 - currently reserved */
    undoCreateCrg,     /* 1 */
    undoStartCrg,      /* 2 */
    newActionCode,     /* 3 */
    undoEndCrg,        /* 4 */
    newActionCode,     /* 5 - no undo for this action code */
    newActionCode,     /* 6 - currently reserved */
    newActionCode,     /* 7 */
    undoMemberIsJoining, /* 8 */
    undoMemberIsLeaving, /* 9 */
    undoSwitchPrimary, /* 10 */
    undoAddNode,       /* 11 */
    undoRmvNode,       /* 12 */
    undoChgCrg,        /* 13 */
    newActionCode,     /* 14 */
    newActionCode,     /* 15 */
    newActionCode,     /* 16 */
    newActionCode,     /* 17 - applies only to a device CRG */
    newActionCode,     /* 18 - applies only to a device CRG */
    newActionCode,     /* 19 - applies only to a device CRG */
    newActionCode,     /* 20 */
    undoCancelFailover /* 21 */
};

/*****
/*
/* This is the entry point for the exit program.
/*
/*-----*/
void main(int argc, char *argv[]) {

    HandlerDataT hdlData;

/*----- */
/*
/* Take each of the arguments passed in the argv array and castit to
/* the correct data type.
/*
/*----- */
    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];

/*-----*/
/*
/* Ensure the format of the data being passed is correct. */

```

```

/* If not, a change has been made and this exit program needs to be
/* updated to accommodate the change. Add appropriate error logging for
/* your application design.
/*
/*-----*/
if (0 != memcmp(formatName, "EXTP0100", 8))
    abort();

/*-----*/
/*
/* Set up the data that will be passed to the exception and cancel
/* handlers.
/*
/*-----*/
hdlData.retCode    = retCode;
hdlData.epData     = epData;
hdlData.crgData    = crgData;
hdlData.actionCode = *actionCode;
hdlData.role       = UnknownRole;
hdlData.priorRole  = UnknownRole;
_VBDY(); /* force changed variables to home storage location */

/*-----*/
/*
/* Enable an exception handler for any and all exceptions.
/*
/*-----*/
#pragma exception_handler(unexpectedExceptionHandler, hdlData, ¥
                        _C1_ALL, _C2_ALL, _CTLA_INVOKE )

/*-----*/
/*
/* Enable a cancel handler to recover if this job is canceled.
/*
/*-----*/
#pragma cancel_handler(cancelHandler, hdlData)

/*-----*/
/*
/* Extract the role and prior role of the node this exit program is
/* running on. If the cluster API or event changes the recovery domain
/* (node role or membership status), the new recovery domain's offset is
/* passed in Offset_Rcvy_Domain_Array and the offset of the recovery
/* domain as it looked prior to the API or cluster event is passed in
/* Offset_Prior_Rcvy_Domain_Array. If the recovery domain isn't changed,
/* only Offset_Rcvy_Domain_Array can be used to address the recovery
/* domain.
/*
/*-----*/
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(); /* force changed variables to home storage location      */

/*-----*/
/*
/* Enable the following to print out debug information.          */
/*
/*-----*/

/*-----*/
/*
    printParms(*actionCode, hdlData.role, hdlData.priorRole, crgData,
epData);
*/

/*-----*/
/*
/* Do the correct thing based upon the action code. The return code
/* is set to the function result of doAction().
/*
/*-----*/

/*-----*/
/*retCode = doAction(*actionCode,
                    hdlData.role,
                    hdlData.priorRole,
                    crgData,
                    epData);

/*-----*/
/*
/* The exit program job will end when control returns to the operating
/* system at this point.
/*
/*-----*/

return;

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

/*****
/*
/* Get the role of this particular node from one of the views of the
/* recovery domain.
/*
/* APIs and cluster events which pass the updated and prior recovery domain*/
/* to the exit program are:
/*
/* QcstAddNodeToRcvyDomain
/* QcstChangeClusterNodeEntry
/* QcstChangeClusterResourceGroup
/* QcstEndClusterNode (ending node does not get the prior domain)
/* QcstInitiateSwitchOver
/* QcstRemoveClusterNodeEntry (removed node does not get the prior domain)
/* QcstRemoveNodeFromRcvyDomain
/* QcstStartClusterResourceGroup (only if inactive backup nodes are
/* reordered)
/*
/* a failure causing failover
/* a node rejoining the cluster
/* cluster partitions merging
/*
/* All other APIs pass only the updated recovery domain.

```



```

/*                                                                 */
/*****                                                                 */
static int getMyRole(Qcst_EXTP0100_t *crgData, int offset, int
count) {

    Qcst_Rcvy_Domain_Array1_t *nodeData;
    unsigned int iter = 0;

/*-----*/
/*                                                                 */
/* Under some circumstances, the operating system may not be able to */
/* determine the ID of this node and passes *NONE. An example of such a */
/* circumstance is when cluster resource services is not active on a */
/* node and the DLTCRG CL command is used.                               */
/*                                                                 */
/*-----*/
    if (0 == memcmp(crgData->This_Nodes_ID, QcstNone,
sizeof(Qcst_Node_Id_t)))
        return UnknownRole;

/*-----*/
/*                                                                 */
/* Compute a pointer to the first element of the recovery domain array. */
/*                                                                 */
/*-----*/
    nodeData = (Qcst_Rcvy_Domain_Array1_t *)((char *)crgData +
offset);

/*-----*/
/*                                                                 */
/* Find my node in the recovery domain array. I will not be in the */
/* prior recovery domain if I am being added by the Add Node to Recovery */
/* Domain API.                                                         */
/*                                                                 */
/*-----*/
    while ( 0 != memcmp(crgData->This_Nodes_ID,
nodeData->Node_ID,
sizeof(Qcst_Node_Id_t))
        &&
iter < count
    ) {
        nodeData++;
        iter++;
    }

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

/*****                                                                 */
/*                                                                 */
/* Call the correct function based upon the cluster action code. The */
/* doAction() function was split out from main() in order to clarify the */
/* example. See the function prologues for each called function for */
/* information about a particular cluster action.                       */
/*                                                                 */
/* Each action code is split out into a separate function only to help */
/* clarify this example. For a particular exit program, some action codes */

```

```

/* may perform the same function in which case multiple action codes could */
/* be handled by the same function. */
/* */
/*****
static int doAction(int actionCode,
                   int role,
                   int priorRole,
                   Qcst_EXTP0100_t *crgData,
                   EpData *epData) {

/*-----*/
/*
/* For action codes this exit program knows about, call a function to */
/* do the work for that action code. */
/*
/*
/*-----*/

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

/*-----*/
/*
/* IBM has defined a new action code in a new operating system release */
/* and this exit program has not yet been updated to handle it. Take a */
/* default action for now. */
/*
/*
/*-----*/
    return newActionCode(role, priorRole, crgData, epData);
} /* end doAction() */

/*****
/*
/* Action code = QcstCrgAcInitialize */
/*
/* The QcstCreateClusterResourceGroup API was called. A new cluster */
/* resource group object is being created. */
/*
/* Things to consider: */
/* - Check that the application program and all associated objects are on */
/* the primary and backup nodes. If the objects are not there, */
/* consider sending error/warning messages or return a failure return */
/* code. */
/* - Check that required data or device CRGs are on all nodes in the */
/* recovery domain. */
/* - Perform any necessary setup that is required to run the */
/* the application on the primary or backup nodes. */
/* - If this CRG is enabled to use the QcstDistributeInformation API, */
/* the user queue needed by that API could be created at this time. */
/*
/*****
static int createCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

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

/*****
/*
/* Action code = QcstCrgAcStart */

```

```

/* */
/* The QcstStartClusterResourceGroup API was called. A cluster resource */
/* group is being started. */
/* The QcstInitiateSwitchOver API was called and this is the second action */
/* code being passed to the exit program. */
/* The fail over event occurred and this is the second action code being */
/* passed to the exit program. */
/* */
/* A maximum wait time is used when checking to see if all dependent CRGs */
/* are active. This is a short time if the CRG is being started because of */
/* the QcstStartClusterResourceGroup API. It is a longer time if it is */
/* because of a failover or switchover. When failover or switchover are */
/* being done, it make take a while for data or device CRGs to become */
/* ready so the wait time is long. If the Start CRG API is being used, the */
/* dependent CRGs should already be started or some error occurred, the */
/* CRGs were started out of order, etc. and there is no need for a long */
/* wait. */
/* */
/* Things to consider: */
/* - If this node's role is primary, the application should be started. */
/* This exit program should either call the application so that it runs */
/* in this same job or it should monitor any job started by this */
/* exit program so the exit program knows when the application job */
/* ends. By far, the simplest approach is run the application in this */
/* job by calling it. */
/* Cluster Resource Services is not expecting this exit program to */
/* return until the application finishes running. */
/* - If necessary, start any associated subsystems, server jobs, etc. */
/* - Ensure that required data CRGs have a status of active on all nodes */
/* in the recovery domain. */
/* */
/*****/
static int startCrg(int role,
                   int doesNotApply,
                   Qcst_EXTP0100_t *crgData,
                   EpData *epData) {

    unsigned int maxWaitTime;

    /* Start the application if this node is the primary */
    if (role == QcstPrimaryNodeRole) {

/*-----*/
        /*
         * Determine if all CRGs that this application CRG is dependent upon
         * are ready. If the check fails, return from the Start action code.
         * Cluster Resource Services will change the state of the CRG to
         * Inactive.
         */
/*-----*/

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

/*-----*/
        /*
         * Just before starting the application, update the data area to
         * indicate the application is running.
         */
    }
}

```

```

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

/*-----*/
    /*
    /* Add logic to call application here. It is expected that control
    /* will not return until something causes the application to end: a
    /* normal return from the exit program, the job is canceled, or an
    /* unhandled exception occurs. See the cancelHandler() function for
    /* some common ways this job could be canceled.
    /*
    /*
/*-----*/

/*-----*/
    /*
    /* After the application has ended normally, update the data area to
    /* indicate the application is no longer running.
    /*
    /*
/*-----*/
    setApp1CrgDataArea(App1_Ended);
}
else

/*-----*/
    /*
    /* On backup or replicate nodes, mark the status of the application in
    /* the data area as not running.
    /*
    /*
/*-----*/
    setApp1CrgDataArea(App1_Ended);

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

/*****
/*
/* Action code = QcstCrgAcRestart
/*
/* The previous call of the exit program failed and set the return
/* code to QcstFailWithRestart or it failed due to an exception and the
/* exception was allowed to percolate up the call stack. In either
/* case, the maximum number of times for restarting the exit program has
/* not been reached yet.
/*
/*
/* This action code is passed only to application CRG exit programs which
/* had been called with the Start action code.
/*
/*
/*****
static int restartCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

/*-----*/
    /*
    /* Perform any unique logic that may be necessary when restarting the
    /*

```

```

/* application after a failure and then call the startCrg() function to */
/* do the start functions. */
/* */
/*-----*/

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

/*****/
/* */
/* Action code = QcstCrgAcEnd */
/* */
/* The end action code is used for one of the following reasons: */
/* - The QcstEndClusterResourceGroup API was called. */
/* - The cluster has become partitioned and this node is in the secondary*/
/* partition. The End action code is used regardless of whether the */
/* CRG was active or inactive. Action code dependent data of */
/* QcstPartitionFailure will also be passed. */
/* - The application ended. Action code dependent data of */
/* QcstResourceEnd will also be passed. All nodes in the recovery */
/* domain will see the same action code (including the primary). */
/* - The CRG job has been canceled. The exit program on this node will */
/* be called with the End action code. QcstMemberFailure will be */
/* passed as action code dependent data. */
/* */
/* */
/* Things to consider: */
/* - If the CRG is active, the job running the application is canceled */
/* and the IP takeover address is ended AFTER the exit program is */
/* called. */
/* - If subsystems or server jobs were started as a result of the */
/* QcstCrgAcStart action code, end them here or consolidate all logic */
/* to end the application in the cancelHandler() since it will be */
/* invoked for all Cluster Resource Services APIs which must end the */
/* application on the current primary. */
/* */
/*****/
static int endCrg(int role,
                 int priorRole,
                 Qcst_EXTP0100_t *crgData,
                 EpData *epData) {

/*-----*/
/* */
/* End the application if it is running on this node. */
/* */
/*-----*/

endApplication(QcstCrgAcRemoveNode, role, priorRole, crgData,
epData);

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

/*****/
/* */
/* Action code = QcstCrgAcVerificationPhase */
/* */
/* The verification phase action code is used to allow the exit program to */
/* do some verification before proceeding with the requested function */
/* identified by the action code depended data. If the exit program */

```

```

/* determines that the requested function cannot proceed it should return */
/* QcstFailWithOutRestart. */
/* */
/* */
/* NOTE: The exit program will NOT be called with Undo action code. */
/* */
/*****/
static int verifyPhase(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

/*-----*/
/* */
/* Do verification */
/* */

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

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

/*****/
/* */
/* Action code = QcstCrgAcDelete */
/* */
/* The QcstDeleteClusterResourceGroup or QcstDeleteCluster API was called. */
/* A cluster resource group is being deleted while Cluster Resource */
/* Services is active. */
/* If the QcstDeleteCluster API was used, action code dependent data of */
/* QcstDltCluster is passed. */
/* If the QcstDeleteCluster API was used and the CRG is active, the exit */
/* program job which is still active for the Start action code is canceled*/
/* after the Delete action code is processed. */
/* */
/* Things to consider: */
/* - Delete application programs and objects from nodes where they are */
/* no longer needed such as backup nodes. Care needs to be exercised */
/* when deleting application objects just because a CRG is being */
/* deleted since a particular scenario may want to leave the */
/* application objects on all nodes. */
/* */
/*****/
static int deleteCrg(int role,
                    int doesNotApply,
                    Qcst_EXTP0100_t *crgData,
                    EpData *epData) {

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

/*****/
/* */
/* Action code = QcstCrgAcReJoin */
/* */
/* One of three things is occurring- */
/* 1. The problem which caused the cluster to become partitioned has been */
/* corrected and the 2 partitions are merging back together to become */

```

```

/* a single cluster. Action code dependent data of QcstMerge will be */
/* passed. */
/* 2. A node which either previously failed or which was ended has had */
/* cluster resource services started again and the node is joining the */
/* cluster. Action code dependent data of QcstJoin will be passed. */
/* 3. The CRG job on a particular node which may have been canceled or */
/* ended has been restarted. Action code dependent data of QcstJoin */
/* will be passed. */
/* */
/* Things to consider: */
/* - If the application replicates application state information to other */
/* nodes when the application is running, this state information will */
/* need to be resynchronized with the joining nodes if the CRG is */
/* active. */
/* - Check for missing application objects on the joining nodes. */
/* - Ensure the required data CRGs are on the joining nodes. */
/* - If the application CRG is active, ensure the required data CRGs are */
/* active. */
/* */
/*****/
static int memberIsJoining(int role,
                           int priorRole,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) {

/*-----*/
/*
/* Ensure the data area status on this node starts out indicating */
/* the application is not running if this node is not the primary. */
/* */
/*-----*/
    if (role != QcstPrimaryNodeRole) {
        setApp1CrgDataArea(App1_Ended);
    }

/*-----*/
/*
/* If a single node is rejoining the cluster, you may do a certain set of */
/* actions. Whereas if the nodes in a cluster which became partitioned */
/* are merging back together, you may have a different set of actions. */
/* */
/*-----*/
    if (crgData->Action_Code_Dependent_Data == QcstJoin) {
        /* Do actions for a node joining. */
    }
    else {
        /* Do actions for partitions merging. */
    }

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

/*****/
/*
/* Action code = QcstCrgAcFailover */
/* */
/* Cluster resource services on a particular node(s) has failed or ended */
/* for this cluster resource group. The Failover action code is passed */
/* regardless of whether the CRG is active or inactive. Failover can */
/* happen for a number of reasons: */
/* */
/* - an operator canceled the CRG job on a node. Action code dependent */

```

```

/* data of QcstMemberFailure will be passed. */
/* - cluster resource services was ended on the node (for example, the */
/* QSYSWRK subsystem was ended with CRS still active). Action code */
/* dependent data of QcstNodeFailure will be passed. */
/* - the application for an application CRG has failed on the primary */
/* node and could not be restarted there. The CRG is Active. */
/* Action code dependent data of QcstApplFailure will be passed. */
/* - the node failed (such as a power failure). Action code dependent */
/* data of QcstNodeFailure will be passed. */
/* - The cluster has become partitioned due to some communication failure*/
/* such as a communication line or LAN failure. The Failover action */
/* code is passed to recovery domain nodes in the majority partition. */
/* Nodes in the minority partition see the End action code. Action */
/* code dependent data of QcstPartitionFailure will be passed. */
/* - A node in the CRG's recovery domain is being ended with the */
/* QcstEndClusterNode API. The node being ended will see the End Node */
/* action code. All other nodes in the recovery domain will see the */
/* Failover action code. Action code dependent data of QcstEndNode */
/* will be passed for the Failover action code. */
/* - An active recovery domain node for an active CRG is being removed */
/* from the cluster with the QcstRemoveClusterNodeEntry API. Action */
/* code dependent data of QcstRemoveNode will be passed. If an */
/* inactive node is removed for an active CRG, or if the CRG is */
/* inactive, an action code of Remove Node is passed. */
/* */
/* The exit program is called regardless of whether or not the CRG is */
/* active. The exit program may have nothing to do if the CRG is not */
/* active. */
/* */
/* If the CRG is active and the leaving member was the primary node, */
/* perform the functions necessary for failover to a new primary. */
/* */
/* The Action_Code_Dependent_Data field can be used to determine if: */
/* - the failure was due to a problem that caused the cluster to become */
/* partitioned (all CRGs which had the partitioned nodes in the */
/* recovery domain are affected) */
/* - a node failed or had cluster resource services ended on the node (all*/
/* CRGs which had the failed/ended node in the recovery domain are */
/* affected) */
/* - only a single CRG was affected (for example a single CRG job was */
/* canceled on a node or a single application failed) */
/* */
/* */
/* Things to consider: */
/* - Prepare the new primary node so the application can be started. */
/* - The application should NOT be started at this time. The exit */
/* program will be called again with the QcstCrgAcStart action code if */
/* the CRG was active when the failure occurred. */
/* - If the application CRG is active, ensure the required data CRGs are */
/* active. */
/* */
/*****/
static int memberIsLeaving(int role,
                           int priorRole,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) {

/*-----*/
/*
/* If the CRG is active, perform failover. Otherwise, nothing to do.
/*
/*-----*/
if (crgData->Original_Cluster_Res_Grp_Stat == QcstCrgActive) {

```



```

/*-----*/
/*
/* The CRG is active. Determine if my role has changed and I am now
/* the new primary.
/*
/*-----*/

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

/*-----*/
/*
/* I was not the primary but am now. Do failover actions but don't
/* start the application at this time because this exit program will
/* be called again with the Start action code.
/*
/*-----*/

/*-----*/
/*
/* Ensure the data area status on this node starts out indicating
/* the application is not running.
/*
/*-----*/
        setApplCrgDataArea(Appl_Ended);

/*-----*/
/*
/* If the application has no actions to do on the Start action code
/* and will become active as soon as the takeover IP address is
/* activated, then this code should be uncommented. This code will
/* determine if all CRGs that this application CRG is dependent upon
/* are ready. If this check fails, return failure from the action
/* code.
/*
/*-----*/
/*
/* if (QcstSuccessful != checkDependCrgDataArea(MaxWaitSeconds))
/* return QcstFailWithOutRestart;
/*
    }
}

return QcstSuccessful;
} /* end memberIsLeaving()

/*****
/*
/* Action code = QcstCrgAcSwitchover
/*
/* The QcstInitiateSwitchOver API was called. The first backup node in
/* the cluster resource group's recovery domain is taking over as the
/* primary node and the current primary node is being made the last backup.
/*
/* Things to consider:
/* - Prepare the new primary node so the application can be started.
/* - The application should NOT be started at this time. The exit
/* program will be called again with the QcstCrgAcStart action code.
/* - The job running the application is canceled and the IP takeover
/* address is ended prior to the exit program being called on the
/* current primary.

```

```

/* - Ensure required data or device CRGs have switched over and are    */
/*   active.                                                            */
/*                                                                      */
/*****
static int switchPrimary(int role,
                        int priorRole,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

/*-----*/
/*
/* See if I am the old primary.
/*
/*-----*/
if (priorRole == QcstPrimaryNodeRole) {

/*-----*/
/*
/* Do what ever needs to be done to cleanup the old primary before the */
/* switch. Remember that that job which was running the exit program */
/* which started the application was canceled already.
/*
/* One example may be to clean up any processes holding locks on the */
/* database. This may have been done by the application cancel
/* handler if one was invoked.
/*
/*-----*/
}

/*-----*/
/*
/* I'm not the old primary. See if I'm the new primary.
/*
/*-----*/
else if (role == QcstPrimaryNodeRole) {

/*-----*/
/*
/* Do what ever needs to be done on the new primary before the
/* application is started with the QcstCrgAcStart action code.
/*
/*-----*/

/*-----*/
/*
/* Ensure the data area status on this nodes starts out indicating
/* the application is not running.
/*
/*-----*/
setApp1CrgDataArea(App1_Ended);

/*-----*/
/*
/* If the application has no actions to do on the Start action code
/* and will become active as soon as the takeover IP address is
/* activated, then this code should be uncommented. This code will
/* determine if all CRGs that this application CRG is dependent upon
/* are ready. If this check fails, return failure from the action
/* code.
*/

```

```

        /*
    /*-----*/
    /*   if (QcstSuccessful != checkDependCrgDataArea(MaxWaitSeconds))   */
    /*       return QcstFailWithOutRestart;                               */
    }
    else {
    /*-----*/
        /*
        /* This node is one of the other backup nodes or it is a replicate
        /* node.  If there is anything those nodes must do, do it here.  If
        /* not, remove this else block.
        /*
        /*-----*/

    /*-----*/
        /*
        /* Ensure the data area status on this nodes starts out indicating
        /* the application is not running.
        /*
        /*-----*/
        setApp1CrgDataArea(App1_Ended);
    }

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

/*****
/*
/* Action code = QcstCrgAcAddNode
/*
/* The QcstAddNodeToRcvyDomain API was called.  A new node is being added
/* to the recovery domain of a cluster resource group.
/*
/* Things to consider:
/* - A new node is being added to the recovery domain.  See the
/*   considerations in the createCrg() function.
/* - If this CRG is enabled to use the QcstDistributeInformation API,
/*   the user queue needed by that API could be created at this time.
/*
*****/
static int addNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

    /*-----*/

    /*
    /* Determine if I am the node being added.
    /*
    /*-----*/

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

    /*-----*/

```

```

/*
/* Set the status of the data area on this new node.
/*
/*
/*-----*/
    setApp1CrgDataArea(App1_Ended);

/*-----*/

/*
/* Create the queue needed by the Distribute Information API.
/*
/*
/*-----*/

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

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

/*****
/*
/* Action code = QcstCrgAcRemoveNode
/*
/*
/* The QcstRemoveNodeFromRcvyDomain or the QcstRemoveClusterNodeEntry
/* API was called. A node is being removed from the recovery domain of
/* a cluster resource group or it is being removed entirely from the
/* cluster.
/*
/*
/* This action code is seen by:
/* For the QcstRemoveClusterNodeEntry API:
/* - If the removed node is active and the CRG is Inactive, all nodes in
/* the recovery domain including the node being removed see this
/* action code. The nodes NOT being removed see action code dependent
/* data of QcstNodeFailure.
/* - If the removed node is active and the CRG is Active, the node being
/* removed sees the Remove Node action code. All other nodes in the
/* recovery domain see an action code of Failover and action code
/* dependent data of QcstNodeFailure.
/* - If the node being removed is not active in the cluster, all nodes
/* in the recovery domain will see this action code.
/* For the QcstRemoveNodeFromRcvyDomain API:
/* - All nodes see the Remove Node action code regardless of whether or
/* not the CRG is Active. Action code dependent data of
/* QcstRmvRcvyDmnNode will also be passed.
/*
/*
/* Things to consider:
/* - You may want to cleanup the removed node by deleting objects no
/* longer needed there.
/* - The job running the application is canceled and the IP takeover
/* address is ended after the exit program is called if this is the
/* primary node and the CRG is active.
/* - If subsystems or server jobs were started as a result of the
/* QcstCrgAcStart action code, end them here or consolidate all logic
/* to end the application in the cancelHandler() since it will be
/* invoked for all Cluster Resource Services APIs which must end the
/* application on the current primary.
/*
/*
*****/

```

```

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

/*-----*/

/*
/* Determine if I am the node being removed.
/*
/*
/*-----*/

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

/*-----*/
        /*
        /* End the application if it is running on this node.
        /*
        /*
        /*-----*/
        endApplication(QcstCrgAcRemoveNode, role, priorRole, crgData,
epData);

    }
    return QcstSuccessful;
} /* end rmvNode
*/

/*****/
/*
/* Action code = QcstCrgAcChange
/*
/*
/* The QcstChangeClusterResourceGroup API was called. Some attribute
/* or information stored in the cluster resource group object is being
/* changed. Note that not all changes to the CRG object cause the exit
/* program to be called. As of V5R1M0, only these changes will cause the
/* exit program to be called-
/* - the current recovery domain is being changed
/* - the preferred recovery domain is being changed
/*
/*
/* If any of the above changes are being made but additionally the exit
/* program is being changed to *NONE, the exit program is not called.
/*
/*
/* Things to consider:
/* - None unless changing the recovery domain affects information or
/* processes for this cluster resource group. Note that the primary
/* node cannot be changed with the QcstChangeClusterResourceGroup API
/* if the CRG is active.
/*
/*
/*****/
static int chgCrg(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

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

/*****/
/*

```

```

/* Action code = QcstCrgAcDeleteCommand */
/* */
/* The Delete Cluster Resource Group (DLTCRG) CL command has been called */
/* to delete a cluster resource group object, the QcstDeleteCluster API */
/* has been called, or the QcstRemoveClusterNodeEntry API has been called. */
/* In each case, cluster resource services is not active on the cluster */
/* node where the command or API was called. Thus, this function is not */
/* distributed cluster wide but occurs only on the node where the CL */
/* command or API was called. */
/* */
/* If the QcstDeleteCluster API was used, action code dependent data of */
/* QcstDltCluster is passed. */
/* */
/* See the considerations in the deleteCrg() function */
/* */
/*****/
static int deleteCrgWithCmd(int role,
                           int doesNotApply,
                           Qcst_EXTP0100_t *crgData,
                           EpData *epData) {

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

/*****/
/* */
/* Action code = QcstCrgEndNode */
/* */
/* The QcstEndClusterNode API was called or a CRG job was canceled. */
/* */
/* The QcstCrgEndNode action code is passed to the exit program only on the */
/* node being ended or where the CRG job was canceled. On the node where */
/* a Cluster Resource Services job is canceled, action code dependent data */
/* of QcstMemberFailure will be passed. */
/* When Cluster Resource Services ends on this node or the CRG job ends, it */
/* will cause all other nodes in the cluster to go through failover */
/* processing. The action code passed to all other nodes will be */
/* QcstCrgAcFailover. Those nodes will see action code dependent data of */
/* QcstMemberFailure if a CRG job is canceled or QcstNodeFailure if the */
/* node is ended. */
/* */
/* Things to consider: */
/* - The job running the application is canceled and the IP takeover */
/* address is ended after the exit program is called if this is the */
/* primary node and the CRG is active. */
/* - If subsystems or server jobs were started as a result of the */
/* QcstCrgAcStart action code, end them here. */
/* */
/*****/
static int endNode(int role,
                  int priorRole,
                  Qcst_EXTP0100_t *crgData,
                  EpData *epData) {

/*-----*/
/* */
/* End the application if it is running on this node. */
/* */
/*-----*/

    endApplication(QcstCrgEndNode, role, priorRole, crgData, epData);

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

```

```

/*****
/*
/* Action code = QcstCrgAcChgNodeStatus
/*
/* The QcstChangeClusterNodeEntry API was called. The status of a node
/* is being changed to failed. This API is used to inform cluster resource
/* services that the node did not partition but really failed.
/*
/* Things to consider:
/* - The exit program was called previously with an action code of
/* QcstCrgAcEnd if the CRG was active or an action code of
/* QcstCrgAcFailover if the CRG was inactive because cluster resource
/* services thought the cluster had become partitioned. The user is
/* now telling cluster resource services that the node really failed
/* instead of partitioned. The exit program has something to do only
/* if it performed some action previously that needs to be changed now
/* that node failure can be confirmed.
/*
/*****
static int chgNodeStatus(int role,
                        int priorRole,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

    return QcstSuccessful;
} /* end chgNodeStatus()

/*****
/*
/* Action code = QcstCrgAcCancelFailover
/*
/* Cluster resource services on the primary node has failed or ended
/* for this cluster resource group. A message was sent to the failover
/* message queue specified for the CRG, and the result of that message
/* was to cancel the failover. This will change the status of the CRG to
/* inactive and leave the primary node as primary.
/*
/* Things to consider:
/* - The primary node is no longer participating in cluster activities.
/* The problem which caused the primary node to fail should be fixed
/* so that the CRG may be started again.
/*
/*****
static int cancelFailover(int role,
                        int priorRole,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

    return QcstSuccessful;
} /* end cancelFailover()

/*****
/*
/* Action code = exit program does not know it yet
/*
/* A new action code has been passed to this exit program. This can occur
/* after a new i5/OS release has been installed and some new cluster API
/* was called or some new cluster event occurred. The logic in this exit
/* program has not yet been updated to understand the new action code.
/*
/* Two different strategies could be used for the new action code. The
/* correct strategy is dependent upon the kinds of things this particular
/* exit program does for the application.
/*

```

```

/* One strategy is to not do anything and return a successful return code. */
/* This allows the new cluster API or event to run to completion. It */
/* allows the function to be performed even though this exit program */
/* did not understand the new action code. The risk, though, is that the */
/* exit program should have done something and it did not. At a minimum, */
/* you may want to log some kind of error message about what happened so */
/* that programming can investigate and get the exit program updated. */
/* */
/* The opposite strategy is to return an error return code such as */
/* QcstFailWithRestart. Of course doing this means that the new cluster */
/* API or event cannot be used until the exit program is updated for the */
/* new action code. Again, logging some kind of error message for */
/* programming to investigate would be worthwhile. */
/* */
/* Only the designer of the exit program can really decide which is the */
/* better course of action. */
/* */
/*****
static int newActionCode(int role,
                        int doesNotApply,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

/*-----*/
/*
/* Add logic to log an error somewhere - operator message queue, job */
/* log, application specific error log, etc. so that the exit program */
/* gets updated to properly handle the new action code. */
/* */
/* Note that if this is left coded as it is, this is the "don't do */
/* anything" strategy described in the prologue above. */
/* */
/*-----*/

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

/*****
/*
/* Action code = QcstCrgAcUndo */
/* */
/* Note: The exit program is never called with an undo action code for */
/* any of these prior action codes: */
/* QcstCrgAcChgNodeStatus */
/* QcstCrgAcDelete */
/* QcstCrgAcDeleteCommand */
/* QcstCrgAcEndNode */
/* QcstCrgAcRemoveNode (If the node being removed is active in the */
/* cluster and the API is Remove Cluster Node. */
/* The Remove Node From Recovery Domain will call */
/* with Undo and the Remove Cluster Node API will */
/* call with Undo if the node being removed is */
/* inactive. */
/* QcstCrgAcRestart */
/* QcstCrgAcUndo */
/* */
/* APIs that call an exit program do things in 3 steps. */
/* 1. Logic which must be done prior to calling the exit program. */
/* 2. Call the exit program. */
/* 3. Logic which must be done after calling the exit program. */
/* */
/* Any errors that occur during steps 2 or 3 result in the exit program */
/* being called again with the undo action code. This gives the exit */
/* program an opportunity to back out any work performed when it was first */

```



```

/* called by the API. The API will also be backing out any work it */
/* performed trying to return the state of the cluster and cluster objects */
/* to what it was before the API was called. */
/*
/* It is suggested that the following return codes be returned for the */
/* specified action code as that return code will result in the most */
/* appropriate action being taken. */
/*
/* QcstCrgAcInitialize: QcstSuccessful; The CRG is not created. */
/* QcstCrgAcStart: QcstSuccessful; The CRG is not started. */
/* QcstCrgAcEnd: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcReJoin: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcFailover: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcSwitchover: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcAddNode: QcstSuccessful; The node is not added. */
/* QcstCrgAcRemoveNode: QcstFailWithOutRestart; The CRG is set to Indoubt*/
/* The cause of the failure needs to*/
/* investigated. */
/* QcstCrgAcChange: QcstSuccessful; The recovery domain is not */
/* changed. */
/*
/*****/
static int undoPriorAction(int role,
                          int priorRole,
                          Qcst_EXTP0100_t *crgData,
                          EpData *epData) {

/*-----*/
/*
/* The prior action code defines what the exit program was doing when */
/* it failed, was canceled, or returned a non successful return code. */
/*
/*-----*/
    if (crgData->Prior_Action_Code &lt;= MaxAc )
        return (*undoFcn[crgData-&lt;Prior_Action_Code])
                (role, priorRole, crgData,
                epData);
    else

/*-----*/
/*
/* IBM has defined a new action code in a new operating system release */
/* and this exit program has not yet been updated to handle it. Take a*/
/* default action for now. */
/*
/*-----*/
    return newActionCode(role, priorRole, crgData, epData);
} /* end undoPriorAction()

/*****/
/*
/* Action code = QcstCrgAcUndo */
/*
/* Prior action code = QcstCrgAcInitialize */
/*

```

```

/* Things to consider: */
/* The CRG will not be created. Objects that might have been created */
/* on nodes in the recovery domain should be deleted since a subsequent */
/* create could fail if those objects already exist. */
/* */
/*****/
static int undoCreateCrg(int role,
                        int doesNotApply,
                        Qcst_EXTP0100_t *crgData,
                        EpData *epData) {

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

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcStart */
/* */
/* Things to consider: */
/* Cluster Resource Services failed when it was finishing the Start CRG */
/* API after it had already called the exit program with the Start */
/* Action code. */
/* */
/* On the primary node, the exit program job which is running the */
/* application will be canceled. The exit program will then be called */
/* with the Undo action code. */
/* */
/* All other nodes in the recovery domain will be called with the Undo */
/* action code. */
/* */
/*****/
static int undoStartCrg(int role,
                       int doesNotApply,
                       Qcst_EXTP0100_t *crgData,
                       EpData *epData) {

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

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcEnd */
/* */
/* Things to consider: */
/* The CRG will not be ended. If the exit program did anything to bring */
/* down the application it can either restart the application or it can */
/* decide to not restart the application. If the application is not */
/* restarted, the return code should be set to QcstFailWithOutRestart so */
/* the status of the CRG is set to Indoubt. */
/* */
/*****/
static int undoEndCrg(int role,
                     int doesNotApply,
                     Qcst_EXTP0100_t *crgData,
                     EpData *epData) {

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

/*****/

```

```

/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcReJoin */
/* */
/* Things to consider: */
/* An error occurred which won't allow the member to join this CRG */
/* group. Anything done for the Join action code needs to be looked at */
/* to see if something must be undone if this member is not an active */
/* member of the CRG group. */
/* */
/*****/
static int undoMemberIsJoining(int role,
                               int doesNotApply,
                               Qcst_EXTP0100_t *crgData,
                               EpData *epData) {

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

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcFailover */
/* */
/* Things to consider: */
/* This does not mean that the node failure or failing member is being */
/* undone. That failure is irreversible. What it does mean is that the */
/* exit program returned an error from the Failover action code or */
/* Cluster Resource Services ran into a problem after it called the exit */
/* program. If the CRG was active when Failover was attempted, it is */
/* not at this point. End the resilient resource and expect a human to */
/* look into the failure. After the failure is corrected, the CRG will */
/* must be started with the Start CRG API. */
/* */
/* */
/*****/
static int undoMemberIsLeaving(int role,
                               int doesNotApply,
                               Qcst_EXTP0100_t *crgData,
                               EpData *epData) {

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

/*****/
/* */
/* Action code = QcstCrgAcUndo */
/* */
/* Prior action code = QcstCrgAcSwitchover */
/* */
/* Things to consider: */
/* Some error occurred after the point of access was moved from the */
/* original primary and before it could be brought up on the new primary.*/
/* The IP address was ended on the original primary before moving the */
/* point of access but is started on the original primary again. Cluster*/
/* Resource Services will now attempt to move the point of access back */
/* to the original primary. The application exit program and IP takeover*/
/* address will be started on the original primary. */
/* */
/* */
/*****/
static int undoSwitchPrimary(int role,
                             int doesNotApply,

```

```

                Qcst_EXTP0100_t *crgData,
                EpData *epData) {

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

/*****
/*
/* Action code = QcstCrgAcUndo
/*
/* Prior action code = QcstCrgAcAddNode
/*
/* Things to consider:
/* If objects were created on the new node, they should be removed so
/* that a subsequent Add Node to aRecovery Domain does not fail if it
/* attempts to create objects again.
/*
/*
/*****
static int undoAddNode(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

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

/*****
/*
/* Action code = QcstCrgAcUndo
/*
/* Prior action code = QcstCrgAcRemoveNode
/*
/* Things to consider:
/* The node is still in the recovery domain. If objects were removed
/* from the node, they should be added back.
/*
/*
/*****
static int undoRmvNode(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

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

/*****
/*
/* Action code = QcstCrgAcUndo
/*
/* Prior action code = QcstCrgAcChange
/*
/* Things to consider:
/* Changes to the CRG will be backed out so that the CRG and its
/* recovery domain look just like it did prior to the attempted change.
/* Any changes the exit program made should also be backed out.
/*
/*
/*****
static int undoChgCrg(int role,
                      int doesNotApply,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

    return QcstSuccessful;

```

```

} /* end undoChgCrg() */

/*****
/*
/* Action code = QcstCrgAcUndo
/*
/* Prior action code = QcstCrgAcCancelFailover
/*
/* Things to consider:
/* This does not mean that the node failure or failing member is being
/* undone. That failure is irreversible. What it does mean is that
/* Cluster Resource Services ran into a problem after it called the exit
/* program. The CRG will be InDoubt regardless of what is returned from
/* this exit program call. Someone will need to manually look into the
/* the failure. After the failure is corrected, the CRG will must be
/* started with the Start CRG API.
/*
/*
/*
/*****
static int undoCancelFailover(int role,
                             int doesNotApply,
                             Qcst_EXTP0100_t *crgData,
                             EpData *epData) {

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

/*****
/*
/* A simple routine to take a null terminated object name and a null
/* terminated library name and build a 20 character non-null terminated
/* qualified name.
/*
/*
/*****
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 */

/*****
/*
/* The data area is checked to see if all the CRGs that this application
/* is dependent upon are ready. If they are not ready, a wait for a
/* certain amount of time is performed and the data area is checked again.
/* This check, wait loop continues until all dependent CRGs become ready or
/* until the maximum wait time has been reached.
/*
/* The length of the wait can be changed to some other value if a
/* particular situation would be better with shorter or longer wait times.
/*
/*
/*
/*****
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;

```

```

/*-----*/
/*
/* This is an accumulation of the time waited for the dependent CRGs to
/* become ready.
/*
/*
/*-----*/
unsigned int timeWaited = 0;

/*-----*/
/*
/* Build definition of the amount of time to wait.
/*
/*
/*-----*/
_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 );

/*-----*/
/*
/* Build the qualified name of the data area.
/*
/*
/*-----*/
bldDataAreaName(DependCrgDataArea, ApplLib, dataAreaName);

/*-----*/
/*
/* Get the data from the data area that indicates whether or not the
/* CRGs are all ready. This data area is updated by the High
/* Availability Business Partners when it is ok for the application to
/* proceed.
/*
/*
/*-----*/
QWCRDTAA(&data,
        sizeof(data),
        dataAreaName,
        offsetof(Qcst_HAAPPO_t,Data_Status)+1, /* API wants a 1 origin */
        sizeof(data.ready),
        &errCode);

/*-----*/
/*
/* If the dependent CRGs are not ready, wait for a bit and check again.
/*
/*
/*-----*/
while (data.ready != Data_Available) {

/*-----*/
/*
/* If the dependent CRGs are not ready after the maximum wait time,
/* return an error. Consider logging some message to describe why the
/* application did not start so that the problem can be looked into.
/*
/*
/*

```

```

/*-----*/
    if (timeWaited >= maxWaitTime)
        return QcstFailWithOutRestart;

/*-----*/
    /*
    /* Wait to allow the data CRGs to become ready.
    /*
    /*

/*-----*/
    waittime(&timeToWait, options);
    timeWaited += WaitSecondsIncrement;

/*-----*/
    /*
    /* Get information from the data area again to see if the data CRGs are*/
    /* ready.
    /*
    /*

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

    return QcstSuccessful;
} /* end checkDependCrgDataArea */

/*****
/*
/* The application CRG data area is updated to indicate that the
/* application is running or to indicate it is not running. This data area*/
/* information is used by the High Availability Business Partners to
/* coordinate the switchover activities between CRGs that have dependencies*/
/* on each other.
/*
/*
/*****
static void setApp1CrgDataArea(char status) {

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

/*-----*/
    /*
    /* Set up the CL command string with the data area library name, the data*/
    /* area name, and the character to put into the data area. Then run the */
    /* CL command.
    /*
    /*

/*-----*/
    memcpy(cmd, "CHGDTAARA DTAARA(", strlen("CHGDTAARA DTAARA")+1);
    strcat(cmd, ApplLib);
    strcat(cmd, "/");
    strcat(cmd, App1CrgDataArea);
    strcat(cmd, " (425 1) VALUE("); /* @A1C */
    cmdEnd[0] = status;
    strcat(cmd, cmdEnd);

    system(cmd);

```

```

return;
} /* end setApplCrgDataArea */

/*****
/*
/* This function is called any time the exit program receives an exception */
/* not specifically monitored for by some other exception handler. Add */
/* appropriate logic to perform cleanup functions that may be required. */
/* A failure return code is then set and control returns to the operating */
/* system. The job this exit program is running in will then end. */
/*
/* When this function gets called, myData->role may still contain the */
/* UnknownRole value if an exception occurred before this node's role */
/* value was set. To be completely correct, the role should be tested */
/* for UnknownRole before making any decisions based upon the value of */
/* role. */
/*
*****/
static void unexpectedExceptionHandler(_INTRPT_Hndlr_Parms_T
*exData) {

/*----- */
/*
/* Get a pointer to the structure containing data that is passed to the */
/* exception handler. */
/*
/*-----*/
HandlerDataT *myData = (HandlerDataT *)exData->Com_Area;

/*-----*/
/*
/* Perform as much cleanup function as necessary. Some global state */
/* information may must be kept so the exception handler knows what */
/* steps were completed before the failure occurred and thus knows what */
/* cleanup steps must be performed. This state information could be */
/* kept in the HandlerDataT structure or it could be kept in some other */
/* location that this function can address. */
/*
/*-----*/

/*-----*/
/*
/* If this is the primary node and the application was started, end it. */
/* The application is ended because the exit program will be called again*/
/* with the Restart action code and want the restartCrg() function to */
/* always work the same way. In addition, ending the application may */
/* clear up the condition that caused the exception. */
/* If possible, warn users and have them stop using the application so */
/* things are done in an orderly manner. */
/*
/*-----*/
endApplication(myData->actionCode,
myData->role,
myData->priorRole,
myData->crgData,
myData->epData);

/*-----*/

```



```

/*                                                                    */
/* Set the exit program return code.                                  */
/*                                                                    */

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

/*-----*/
/*                                                                    */
/* Let the exception percolate up the call stack.                    */
/*                                                                    */
/*-----*/
return;
} /* end unexpectedExceptionHandler */

/*****
/*                                                                    */
/* This function is called any time the job this exit program is running in*/
/* is canceled. The job could be canceled due to any of the following */
/* (the list is not intended to be all inclusive)- */
/* - an API cancels an active application CRG. The End CRG, Initiate */
/*   Switchover, End Cluster Node, Remove Cluster Node or Delete Cluster */
/*   API cancels the job which was submitted when the exit program was */
/*   called with a Start action code. */
/* - operator cancels the job from some operating system display such as */
/*   Work with Active Jobs */
/* - the subsystem this job is running in is ended */
/* - all subsystems are ended */
/* - the system is powered down */
/* - an operating system machine check occurred */
/*                                                                    */
/* When this function gets called, myData->role may still contain the */
/* UnknownRole value if cancelling occurred before this node's role */
/* value was set. To be completely correct, the role should be tested */
/* for UnknownRole before making any decisions based upon the value of */
/* role. */
/*                                                                    */
/*****
static void cancelHandler(_CNL_Hndlr_Parms_T *cnlData) {

/*-----*/
/*                                                                    */
/* Get a pointer to the structure containing data that was passed to the */
/* cancel handler. */
/*                                                                    */
/*-----*/
HandlerDataT *myData = (HandlerDataT *)cnlData->Com_Area;

/*-----*/
/*                                                                    */
/* Perform as much cleanup function as necessary. Some global state */
/* information may must be kept so the cancel handler knows what */
/* steps were completed before the job was canceled and thus knows if */
/* the function had really completed successfully or was only partially */
/* complete and thus needs some cleanup to be done. This state */
/* information could be kept in the HandlerDataT structure or it could */
/* be kept in some other location that this function can address. */
/*                                                                    */
/*-----*/

```

```

/*-----*/
/*
/* This job is being canceled. If I was running the application as a
/* result of the Start or Restart action codes, end the application now.
/* This job is being canceled because a Switch Over or some other
/* Cluster Resource Services API was used which affects the primary node
/* or someone did a cancel job with a CL command, from a system display,
/* etc.
*/
*/

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

/*-----*/
/*
/* Set the exit program return code.
/*
*/
*/

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

/*-----*/
/*
/* Return to the operating system for final ending of the job.
/*
*/
*/

/*-----*/
return;
} /* end cancelHandler */

/*****
/*
/* A common routine used to end the application by various action code
/* functions, the exception handler, and the cancel handler.
/*
/*
*****/
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)
    {

/*-----*/
/*
/* Add logic to end the application here. You may need to add logic
/* to determine if the application is still running because this
/* function could be called once for an action code and again from
/* the cancel handler (End CRG is an example).
/*
*/
*/

/*-----*/

```

```

/*-----*/
/*
/* After the application has ended, update the data area to indicate
/* the application is no longer running.
/*
/*
/*-----*/
    setApp1CrgDataArea(App1_Ended);
}

return;
} /* end endApplication */

/*****
/*
/* Print out the data passed to this program.
/*
/*
/*****
static void printParms(int actionCode,
                      int role,
                      int priorRole,
                      Qcst_EXTP0100_t *crgData,
                      EpData *epData) {

    unsigned int i;
    char *str;

    /* Print the action code.
    printf("%s", "Action_Code = ");
    printActionCode(actionCode);

    /* Print the action code dependent data.
    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 QcstApp1Failure:    str = "QcstApp1Failure";
                                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);
}

```

```

/* Print the prior action code. */
printf("%s", " Prior_Action_Code = ");
if (crgData->Prior_Action_Code)
    printActionCode(crgData->Prior_Action_Code);
printf("%n");

/* Print the cluster name. */
    printStr(" Cluster_Name = ",
        crgData->Cluster_Name, sizeof(Qcst_Cluster_Name_t));

/* Print the CRG name. */
printStr(" Cluster_Resource_Group_Name = ",
    crgData->Cluster_Resource_Group_Name,
sizeof(Qcst_Crg_Name_t));

/* Print the CRG type. */
printf("%s %n", " Cluster_Resource_Group_Type =
QcstCrgApplResiliency");

/* Print the CRG status. */
printf("%s", " Cluster_Resource_Group_Status = ");
printCrgStatus(crgData->Cluster_Resource_Group_Status);

/* Print the CRG original status. */
printf("%s", " Original_Cluster_Res_Grp_Stat = ");
printCrgStatus(crgData->Original_Cluster_Res_Grp_Stat);

/* Print the Distribute Information queue name. */
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));

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

/* Print the ID of this node. */
printStr(" This_Nodes_ID = ",
    crgData->This_Nodes_ID, sizeof(Qcst_Node_Id_t));

/* Print the role of this node. */
printf("%s %d %n", " this node's role = ", role);

/* Print the prior role of this node. */
printf("%s %d %n", " this node's prior role = ", priorRole);

/* Print which recovery domain this role comes from. */
printf("%s", " Node_Role_Type = ");
if (crgData->Node_Role_Type == QcstCurrentRcvyDmn)
    printf("%s %n", "QcstCurrentRcvyDmn");
else
    printf("%s %n", "QcstPreferredRcvyDmn");

/* Print the ID of the changing node (if any). */
printStr(" Changing_Node_ID = ",
    crgData->Changing_Node_ID, sizeof(Qcst_Node_Id_t));

/* Print the role of the changing node (if any). */
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);

/* Print the takeover IP address. */
printStr(" Takeover_IP_Address = ",
        crgData->Takeover_IP_Address,
sizeof(Qcst_TakeOver_IP_Address_t));

/* Print the job name. */
printStr(" Job_Name = ", crgData->Job_Name, 10);

/* Print the CRG changes. */
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");

/* Print the failover wait time. */
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");

/* Print the failover default action. */
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");

/* Print the failover message queue name. */
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));

/* Print the cluster version. */
printf("%s %d ¥n",
        " Cluster_Version = ", crgData->Cluster_Version);

/* Print the cluster version mod level */
printf("%s %d ¥n",
        " Cluster_Version_Mod_Level = ",
        crgData->Cluster_Version_Mod_Level);

/* Print the requesting user profile. */
printStr(" Req_User_Profile = ",
        crgData->Req_User_Profile,
sizeof(crgData->Req_User_Profile));

/* Print the length of the data in the structure. */
printf("%s %d ¥n",
        " Length_Info_Returned = ",

```

```

crgData->Length_Info_Returned);

/* Print the offset to the recovery domain array. */
printf("%s %d %n",
       " Offset_Rcvy_Domain_Array = ",
crgData->Offset_Rcvy_Domain_Array);

/* Print the number of nodes in the recovery domain array. */
printf("%s %d %n",
       " Number_Nodes_Rcvy_Domain = ",
crgData->Number_Nodes_Rcvy_Domain);

/* Print the current/new recovery domain. */
printRcvyDomain(" The recovery domain:",
               crgData->Number_Nodes_Rcvy_Domain,
               (Qcst_Rcvy_Domain_Array1_t *)
               ((char *)crgData +
crgData->Offset_Rcvy_Domain_Array));

/* Print the offset to the prior recovery domain array. */
printf("%s %d %n",
       " Offset_Prior_Rcvy_Domain_Array = ",
crgData->Offset_Prior_Rcvy_Domain_Array);

/* Print the number of nodes in the prior recovery domain array. */
printf("%s %d %n",
       " Number_Nodes_Prior_Rcvy_Domain = ",
crgData->Number_Nodes_Prior_Rcvy_Domain);

/* Print the prior recovery domain if one was passed. */
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 */

/*****
/*
/* Print a string for the action code.
/*
/*
/*****
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 QcstCrgAcEndNode:   code = "QcstCrgAcEndNode";
                             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 */

```

```

/*****
/*
/* Print the CRG status.
/*
/*
*****/
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 */

```

```

/*****
/*
/* Print the recovery domain.
/*
/*
/*****

```

```

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 */

```

```

/*****
/*
/* Concatenate a null terminated string and a non null terminated string
/* and print it.
/*
/*
/*****
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 */
```

データ回復力の計画

データ回復力は、データがユーザーやアプリケーションに使用可能になる機能です。データ回復力は切り替えディスク、サイト間ミラーリング、または論理複製テクノロジーのいずれかの i5/OS クラスタ・テクノロジーを使用することによって実現できます。

i5/OS 対応のデータ回復力のインプリメントでは、いくつかのテクノロジーを選択できます。これらのテクノロジーを i5/OS クラスタ・リソース・サービスと組み合わせることにより、完璧な高可用性ソリューションを構築できます。これらのテクノロジーは次のように分類できます。

i5/OS 独立ディスク・プール・テクノロジー

これらのテクノロジーはすべて、独立ディスク・プールの i5/OS インプリメントに基づいています。独立ディスク・プール・テクノロジーを使用する高可用性の場合、障害からの回復力を必要とするすべてのデータが独立ディスク・プールに保管されている必要があります。多くの場合、これには独立ディスク・プールへのデータの移行が必要です。この情報は、データの移行が完了していることを想定しています。

次の i5/OS 対応テクノロジーは、独立ディスク・プールに基づいています。

- 切り替えディスク
- リモート・ミラーリング
- メトロ・ミラー
- グローバル・ミラー

論理複製テクノロジー

論理複製は、ジャーナル・ベースのテクノロジーで、データはリアルタイムで別のシステムに複製されます。論理複製テクノロジーでは i5/OS クラスタ・リソース・サービスと、IBM ビジネス・パートナー・アプリケーションによるジャーナリングを使用しています。これらのソリューションには、環境の構成と管理のための高可用性ビジネス・パートナー・アプリケーションが必要です。この情報は、これらの IBM ビジネス・パートナー・ソリューションの特定の要件を提供するものではありません。高可用性のために論理複製ソリューションをインプリメントする場合は、アプリケーション関連の情報を参考にするか、サービス担当員にお問い合わせください。

関連情報



IBM eServer iSeries 独立 ASP: アプリケーションの独立 ASPへの移動ガイド (英語)

どのデータが回復されるべきかを判断する

どのタイプのデータを回復できるように考慮すべきなのかを理解してください。

障害からの高い回復力が必要なのはどのデータであるかを判断することは、担当システムのバックアップ戦略および回復戦略を準備する際に、どのタイプのデータを、バックアップおよび保管する必要があるかを考慮することに似ています。環境内のデータのうち、ビジネスが稼働状態になる上で重要なデータはどのデータであるかを見極める必要があります。

例えば、Web 上で業務を行う場合には、次のようなデータが重要なデータとして挙げられます。

- 本日分の注文
- 在庫
- 顧客レコード

変更の頻度が少ない情報や日常の業務であまり使わないデータは、通常、回復する必要はありません。

切り替えディスクの計画

データの単一コピーが、切り替え可能なハードウェア (拡張装置 (タワー) または論理区画環境の IOP のいずれか) で保守されます。

プライマリー・ノードで障害が発生したときには、切り替え可能ハードウェア上のデータへのアクセスは指定されたバックアップ・ノードへ切り替えられます。さらに、サイト間ミラーリング (XSM) 環境で独立ディスク・プールを使用することもできます。その場合、利用および保護を目的として、起点サイトから地理的に遠く離れたシステムにも、独立ディスク・プールのミラー・コピーを保存することができます。

切り替え可能な独立ディスク・プール上に置かれた切り替え可能リソースまたはサイト間ミラーリングの利点をいかす予定の場合は、慎重な計画が必要です。

また、現行のシステム・ディスク構成を評価して、追加のディスク装置が必要かどうかを判別することも必要です。システム・ディスク構成全般に言えることですが、アプリケーションに対して使用可能なディスク装置の数が、パフォーマンスに大きな影響を及ぼす場合があります。限られた数のディスク装置上でワークロードが増えた場合には、ディスク待機が長くなるため、アプリケーションに対する応答時間も長くなります。独立ディスク・プールを使用して構成されたシステム内の一時ストレージの場合、これは特に重要です。一時ストレージは、すべて SYSBAS ディスク・プールに書き込まれます。アプリケーションで使用される一時ストレージがそれほど多くない場合には、SYSBAS ディスク・プール内のディスク・アームが少なくても対処は可能です。また、オペレーティング・システムおよび基本機能は SYSBAS ディスク・プールに存在することも覚えておいてください。

IBM Systems Director Navigator for i5/OS を使用して独立ディスク・プールの作成などのディスク管理タスクを実行する前に、専用保守ツール (DST) のための適切な権限をセットアップする必要があります。

関連タスク

iSeries ナビゲーターでディスク装置にアクセスする

1 切り替えディスクのハードウェア要件:

- 1 切り替えディスクを使用するには、特定のハードウェアが必要になります。
- 1 切り替えディスクを使用するには、以下のいずれかが必要になります。
 - 1 • 高速リンク (HSL) ループ上にある 1 つ以上の拡張装置 (フレーム/装置)。
 - 1 • 共用バス上の 1 つ以上の IOP、または入出力プールに割り当てられた IOP。LPAR 環境では、拡張装置がなくても、独立した切り替えディスクが含まれる IOP をシステム区画間で切り替えて使用することができます。IOP は、複数の区画によって共用されるバス上に置くか、または入出力プールに割り当てる必要があります。IOP 上にある IOA は、すべて切り替えられます。
- 1 これらのハードウェア要件の他に、切り替えディスクに対しては以下の設備計画が必要です。
 - 1 • 高速リンク (HSL) ケーブルを使用して、拡張装置をクラスター内のシステムに接続する必要があります。この拡張装置は、HSL ループ内において、代替システムまたは代替システムが所有する拡張装置と物理的に隣接している必要があります。それぞれの HSL ループ上には、最大で 2 つのシステム (クラ

スター・ノード)を組み込むことができます。ただし、各システムとも複数の HSL ループに接続できません。各 HSL ループには、最大 4 つの拡張装置を組み込むことができます。ただし、各ループ・セグメントに組み込める拡張装置は、最大で 3 つです。2 つのシステムが組み込まれた HSL ループには、2 つのセグメントが存在し、2 つのシステムによって分離されています。1 つのループ・セグメント上にある拡張装置は、すべて同一のデバイス・クラスター・リソース・グループ (CRG) に組み込まれている必要があります。

- 拡張装置を切り替え可能にするには、ループ・セグメント上の所有システムから物理的に最も離れたところに、拡張装置を置く必要があります。注: ある拡張装置を切り替え可能にしようとしたときに、所有システムからの距離がその装置よりも遠い場所に、切り替え可能になっていない他の拡張装置があると、エラーが発生します。

- 切り替え可能な拡張装置は、デバイス・クラスター・リソース・グループ (装置 CRG) のプライマリー・ノードとして最初に機能するシステム装置に、SPCN ケーブルで接続されている必要があります。システム装置内の 1 次論理区画または 2 次論理区画が、プライマリー・ノードである場合があります。論理区画を使用するには、意図する拡張装置内のシステム・バスを、クラスターに関係する区画によって所有および専用化する必要があります。

切り替えディスクのソフトウェア要件:

i5/OS 高可用性用の切り替えディスクの使用を計画している場合は、最小ソフトウェア要件が満たされていることを確認してください。

- このテクノロジーの新規機能および拡張機能およびフィーチャーを使用するには、このテクノロジーに基づいて高可用性ソリューションに参加している各システムまたは論理区画上のオペレーティング・システムの最新リリースおよび最新バージョンをインストールするよう、お勧めします。

注: 同じ HSL ループにあるシステムについては、高可用性の Web サイトを参照し、i5/OS の互換可能バージョンがあることを確認してください。

- 独立ディスク・プールをインプリメントするために必要な、一部のディスク管理タスクを実行するには、以下のグラフィカル・インターフェースのいずれかが必要です。

- IBM Systems Director Navigator for i5/OS
- System i ナビゲーター

- i5/OS オプション 41 HA 切り替え可能リソースをインストールする必要があります。オプション 41 を選択すると、システム間で独立ディスク・プールを切り替えることができます。システム間で独立ディスク・プールを切り替えるには、システムがクラスターのメンバーであり、独立した切り替えディスクがそのクラスターの装置クラスター・リソース・グループと関連付けられている必要があります。また、オプション 41 は、IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラムの一部として提供される高可用性管理インターフェースの処理にも必要です。

関連情報

高可用性およびクラスター (英語)

切り替えディスクの通信要件:

切り替えディスクを使用するには、少なくとも TCP/IP 通信インターフェースが 1 つ、クラスター内のシステム間に必要となります。

予備として、システム間に 2 つのインターフェースを個別に持つことをお勧めします。

サイト間ミラーリングの計画

サイト間ミラーリングは、リモート・ミラーリング、メトロ・ミラーリング、およびグローバル・ミラーなどの複数の i5/OS 災害時回復および高可用性テクノロジーを提供します。

サイト間ミラーリング・テクノロジーは、通常は互いに一定の距離に離れている別々のサイトを保守することにより、災害時回復をインプリメントします。これらのテクノロジーはそれぞれ特定の通信、ハードウェア、およびソフトウェア要件を持っています。ただし、これらのテクノロジーの 1 つをインプリメントする前に、ご使用のサイトも計画する必要があります。1 つのサイトは、実動またはソース・サイトと見なされます。このサイトにはリモート・サイトにミラーリングされる、またはコピーされる実動データが含まれています。リモート・サイト (バックアップ・サイトまたはターゲット・サイトとも呼ばれます) には、実動データのミラーリングされたコピーが含まれています。実動場所で全サイトの災害が起こった場合は、バックアップ・サイトがミラーリングされたデータを使用して、業務を再開します。サイト間ミラーリング・テクノロジーを構成する前に、サイト計画について以下を検討してください。

どのサイトを実動場所とし、どのサイトをバックアップ・サイトとするかを決定する

それぞれのサイトに配置されている現行のハードウェアおよびソフトウェア・リソースにアクセスして、サイト間ミラーリング・ソリューションで必要となるコンポーネントが欠けていないかを判別します。

実動場所とバックアップ・サイト間の距離を決める

通信速度および他の要因により、選択するミラーリング・テクノロジーにおけるパフォーマンスと待ち時間に対してサイト間の距離が影響を与える場合があります。遠距離にあるサイトに適しているサイト間ミラーリング・テクノロジーもあれば、遠距離サイトではパフォーマンスの低下が発生する可能性のあるテクノロジーもあります。

DST に対する適切な権限を持っていることを確認する

IBM Systems Director Navigator for i5/OS を使用してディスク管理タスクを実行する前に、専用保守ツール (DST) に対する適切な権限をセットアップする必要があります。

関連タスク

iSeries ナビゲーターでディスク装置にアクセスする

リモート・ミラーリングの計画:

リモート・ミラーリングは、サイト間ミラーリングの副次機能です。このテクノロジーは、i5/OS 環境における災害時回復と高可用性を提供します。

1 リモート・ミラーリングのハードウェア要件:

- 1 i5/OS 高可用性にリモート・ミラーリングを使用することを計画している場合には、最小ハードウェア要件が満たされていることを確認します。
- 1 独立ディスク・プールのハードウェア要件がすべて満たされている必要があります。
- 1 少なくとも 2 つの System i モデル (地理的な分離が可能) が必要です。
- 1 各サイトに、少なくとも 2 つのディスク・セット (容量が同じもの) が必要です。
- 1 リモート・ミラーリングされた独立ディスク・プールを使用して、個別の記憶域プールをジョブ用に構成する必要があります。主記憶域プールからリモート・ミラーリングを実行すると、過度のロード状態が発生して、システムが停止する場合があります。
- 1 リモート・ミラーリングは、ディスク・プールが使用可能な場合に実行されます。リモート・ミラーリングの実行時には、時刻 (QTIME) のシステム値を変更しないでください。
- 1 独立ディスク・プールの通信要件は、スループットに影響を及ぼすため、特に重要です。

関連概念

『リモート・ミラーリングの通信要件』

リモート・ミラーリングを使用する i5/OS 高可用性ソリューションをインプリメントする場合、リモート・ミラーリングのトラフィックがシステム・パフォーマンスに悪影響を与えないように、通信回線を計画する必要があります。

リモート・ミラーリングのソフトウェア要件:

i5/OS 高可用性ソリューションの一部としてリモート・ミラーリングを使用することを計画している場合には、以下のソフトウェアが必要になります。

- リモート・ミラーリングの拡張機能を使用するには、IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム番号 (5761-HAS) がインストールされている必要があります。
- このテクノロジーの新規機能および拡張機能およびフィーチャーを使用するには、このテクノロジーに基づいて高可用性ソリューションに参加している各システムまたは論理区画上のオペレーティング・システムの最新リリースおよび最新バージョンをインストールするよう、お勧めします。

注: 同じ HSL ループにあるシステムについては、高可用性の Web サイトを参照し、i5/OS の互換可能バージョンがあることを確認してください。

- 独立ディスク・プールをインプリメントするために必要な、一部のディスク管理タスクを実行するには、以下のグラフィカル・インターフェースのいずれかが必要です。
 - IBM Systems Director Navigator for i5/OS
 - System i ナビゲーター
- i5/OS オプション 41 HA 切り替え可能リソースをインストールする必要があります。オプション 41 を選択すると、システム間で独立ディスク・プールを切り替えることができます。システム間で独立ディスク・プールを切り替えるには、システムがクラスターのメンバーであり、独立した切り替えディスクがそのクラスターの装置クラスター・リソース・グループと関連付けられている必要があります。また、オプション 41 は、IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラムの一部として提供される高可用性管理インターフェースの処理にも必要です。

関連情報

高可用性およびクラスター (英語)

リモート・ミラーリングの通信要件:

リモート・ミラーリングを使用する i5/OS 高可用性ソリューションをインプリメントする場合、リモート・ミラーリングのトラフィックがシステム・パフォーマンスに悪影響を与えないように、通信回線を計画する必要があります。

以下は、推奨事項になります。

- リモート・ミラーリングを行うと、通信トラフィックが増大する場合があります。リモート・ミラーリングが同一の IP 接続を他のアプリケーションと共有すると (クラスタリングなど)、リモート・ミラーリングは中断状態になり、同期が行われず。同様に、応答のクラスタリングが受け入れられなかった場合には、ノードの区画化が行われます。リモート・ミラーリングでは、独自の専用通信回線を使用することをお勧めします。独自の通信回線がないと、リモート・ミラーリングは同じ通信回線を使用する他のアプリケーションと争う形になるため、ユーザー・ネットワークのパフォーマンスおよびスループットに影響が生じます。また、これによって、クラスター・ハートビート・モニターにも悪影響を与えることになるため、クラスター区画状態を引き起こす原因になります。以上の点から、リモート・ミラーリングとクラスターの両方に専用の通信回線を持つことをお勧めします。リモート・ミラーリングでは、最大 4 つの通信回線をサポートします。

リモート・ミラーリングは、最良のパフォーマンスを得るために、変更内容を複数の回線に分散します。構成済みの各通信回線で、データが順番に (1 から 4 まで) 繰り返し送信されます。4 つの通信回線を使えば最高のパフォーマンスが得られますが、2 つの通信回線でも十分なパフォーマンスが得られます。

ノード間の通信回線を複数使用してリモート・ミラーリングを行う場合は、これらの回線を異なるサブネットに分割して、両方のシステムで回線の使用のバランスを取るようにするとよいでしょう。

- 複数のアプリケーションまたはサービスで同じ通信回線を使用する必要がある構成の場合、i5/OS の TCP/IP 機能を介して Quality of Service (QoS) をインプリメントすることによって、こうした問題はある程度軽減されます。i5/OS の Quality of Service (QoS) ソリューションを使うと、ネットワーク全体における TCP/IP アプリケーションのネットワーク優先順位と帯域幅を、ポリシーによって要求することができます。

ノード間の通信回線を複数使用してリモート・ミラーリングを行う場合は、これらの回線を異なるサブネットに分割して、両方のシステムで回線の使用のバランスを取るようにするとよいでしょう。

- 各データ・ポート接続のスループットが一致していることを確認します。つまり、システム・ペア間のすべての接続に対して、スピードおよび接続タイプを同じにすることです。スループットが異なると、最も低速の接続に合わせる形で、パフォーマンスが制限されます。

- TCP/IP 接続の仮想私設網を構成することを検討してみてください。これには、以下のようなメリットがあります。

- データが暗号化されるため、データ伝送のセキュリティが向上する
- 送信内容の冗長度が増すため、データ伝送の信頼性が向上する

関連概念

52 ページの『リモート・ミラーリングのハードウェア要件』

i5/OS 高可用性にリモート・ミラーリングを使用することを計画している場合には、最小ハードウェア要件が満たされていることを確認します。

関連資料

サービス品質 (英語)

リモート・ミラーリングのジャーナル計画:

i5/OS リモート・ミラーリングに基づいて高可用性をインプリメントする場合には、ジャーナル管理を計画してください。

ジャーナル管理を行うことにより、システムが異常終了した場合にトランザクションが失われるのを防ぎます。オブジェクトをジャーナル処理すると、そのオブジェクトに加えられた変更をシステムが記録します。ジャーナル処理は、インプリメントする高可用性ソリューションにかかわらず、異常なシステム停止が起きたときにデータ損失を防ぐベスト・プラクティスと見なすことができます。

関連情報

ジャーナル管理

リモート・ミラーリングのバックアップ計画:

リモート・ミラーリングに基づく高可用性をインプリメントする前に、この環境内でのバックアップ戦略を理解した上で、計画する必要があります。

高可用性ソリューションを構成する前に、現行のバックアップ戦略を評価して、必要に応じて適切な変更を行います。リモート・ミラーリングでは、独立ディスク・プールのミラー・コピーへの同時アクセスは許可

| されていないため、リモート・バックアップの実行に対する影響が生じます。リモートでミラーリングされ
| たコピーからテープにバックアップを行う場合、実動システムでのミラーリングを静止して、ミラーリング
| されたコピーを切り離す (トラッキングは有効にする) 必要があります。トラッキングを行うことで、実動
| での変更を追跡できます。これによって、ミラーリングされたコピーがオンラインに戻ったときに、同期化
| させることが可能です。次に、切り離された独立ディスク・プールのコピーをオンに変更して、バックアッ
| プ手順を実行し、独立ディスク・プールをオフに変更してから、元の実動ホストに再接続する必要があります
| ます。このプロセスでは、実動コピーとミラーリングされたコピーの間で、データの再同期が必要になります
| が、部分的なもので十分です。

| バックアップの実行中に同期化が実行されると、システムは無防備な状態で稼働することになります。トラ
| ッキングを有効にして、ミラーリングを中断することをお勧めします。これにより、同期処理が高速化され
| ます。ソース・システムとターゲット・システム間のすべての通信パスが一定期間失われるなどといった、
| 送信中断が持続的に発生した場合にも、同期が必要となります。冗長通信パスを使用することでも、通信障
| 害に関連するリスクをある程度減らすことができます。

| リモート・ミラーリングを行う際には、リモート・ミラーリングの保守が可能な同一サイトの他のシステム
| に、独立ディスク・プールの実動コピーを切り替えられるようなシステムまたは論理区画を、少なくとも 3
| つ使用することをお勧めします。

| 関連概念

| 123 ページの『シナリオ: リモート・ミラーリング環境でのバックアップの実行』
| このシナリオでは、リモート・ミラーリングを使用する i5/OS 高可用性ソリューションで、リモート・
| バックアップを実行する際に必要なタスクの概要を説明します。

| 84 ページの『シナリオ: リモート・ミラーリングによる切り替えディスク』
| このシナリオでは、3 つのノード・クラスターにおいてリモート・ミラーリングによる切り替えディス
| クを使用する、i5/OS 高可用性ソリューションについて説明します。このソリューションは、災害時回
| 復と高可用性の両方を提供します。

| リモート・ミラーリングのパフォーマンス計画:

| リモート・ミラーリング・ソリューションをインプリメントする場合、パフォーマンスに対する潜在的な影
| 響を最小限にするために、ご使用の環境を理解したうえで計画する必要があります。

| リモート・ミラーリングのパフォーマンスに影響を及ぼす要因は、さまざまです。リモート・ミラーリング
| 環境で最高のパフォーマンスを得るために、総合的に計画しなければならない考慮事項として、以下のよう
| な要因が挙げられます。

| CPU の考慮事項

| リモート・ミラーリングは CPU の負荷を増加させるため、CPU 容量を十分に確保しなければなりません。
| CPU 容量を増やすためには、追加のプロセッサが必要な場合があります。通常、リモート・ミラー
| リングを実行するために使用する区画では、部分プロセッサ以外も必要になります。最小の CPU 構成で
| は、リモート・ミラーリングの実行中に、CPU のオーバーヘッドが 5% から 20% 程度になる場合があり
| ます。ミラーリングされたシステムの持つプロセッサが実動システムよりも少ない場合に、多数の書き込
| み操作が発生すると、CPU のオーバーヘッドが激増し、パフォーマンスに影響を及ぼす可能性があります
| ます。

| マシン・プール・サイズの考慮事項

| リモート・ミラーリングのパフォーマンスを最良にするには、特に同期時において、少なくとも以下の公式
| によって指定された量だけマシン・プール・サイズを増やしてください。

- | • 追加するマシン・プール・ストレージの量は、次のとおりです。 300 MB + .3MB x 独立ディスク・プ
- | ール内のディスク ARM の数。独立ディスク・プールのディスク ARM が 90 の場合および 180 の場
- | 合に、それぞれで必要となる追加のマシン・プール・ストレージは、次の例にあるとおりです。
- | - 300 + (.3 x 90 ARM) = 327 MB の追加マシン・プール・ストレージ
- | - 300 + (.3 x 180 ARM) = 354 MB の追加マシン・プール・ストレージ

| 切り替えまたはフェイルオーバーの際にターゲット・ノードが十分なストレージを持てるように、クラスタ
| ー・リソース・グループ (CRG) 内のすべてのノードにおいて、マシン・プール・ストレージに余裕を持た
| せることが必要になります独立ディスク・プール内のディスク装置が増えれば、より多くの作業を平行して
| 行えるようになるため、パフォーマンスが向上するというのは当然のことです。

| パフォーマンス調節機能がマシン・プール・サイズを削減しないようにするには、以下のいずれかを実行す
| る必要があります。

- | 1. 共用記憶域プールの処理 (WRKSHRPOOL) コマンドまたは共用記憶域プールの変更 (CHGSHRPOOL)
| コマンドを使用して、マシン・プールの最小サイズを、計算した量 (公式から算出したりリモート・ミラ
| ーリング用の追加サイズを、現行のサイズに加えたもの) に設定します。

| 注: このオプションは、共用記憶域プールの処理 (WRKSHRPOOL) オプションとあわせて使用すること
| をお勧めします。

- | 2. システム値「記憶域プールと活動レベルの自動調整」(QPFRADJ) をゼロに設定して、パフォーマンス・
| アジャスターによるマシン・プールのサイズの変更を禁止します。

| ディスク装置の考慮事項

| ディスク装置および IOA のパフォーマンスは、リモート・ミラーリングのパフォーマンス全体に影響を及
| ぼす可能性があります。これは特に、ミラーリングされたシステム上でディスク・サブシステムが遅くなる
| 場合に当てはまります。リモート・ミラーリングが同期モードである場合、ミラーリングされたコピーのデ
| ィスクへの書き込みに合わせる形で、実動コピーのすべての書き込み操作が制限を受けます。そのため、タ
| ーゲット側のディスク・サブシステムが遅いと、ソース側のパフォーマンスにも影響を及ぼすことになりま
| す。リモート・ミラーリングを非同期モードで実行することにより、このようなパフォーマンスへの影響を
| 最低限に抑えることができます。非同期モードで実行すると、ターゲット側のディスク・サブシステムの待
| 機が軽減され、変更されたメモリー・ページがターゲット側のメモリーにある場合には、ソース側に確認が
| 送り返されます。

| システム・ディスク・プールの考慮事項

| システム・ディスク構成全般に言えることですが、アプリケーションに対して使用可能なディスク装置の数
| が、パフォーマンスに大きな影響を及ぼす場合があります。限られた数のディスク装置上でワークロードが
| 増えた場合には、ディスク待機が長くなるため、アプリケーションに対する応答時間も長くなります。独立
| ディスク・プールを使用して構成されたシステム内の一時ストレージの場合、これは特に重要です。一時ス
| トレージは、すべて SYSBAS ディスク・プールに書き込まれます。アプリケーションで使用される一時ス
| トレージがそれほど多くない場合には、SYSBAS ディスク・プール内のディスク・アームが少なくても対
| 処は可能です。また、オペレーティング・システムおよび基本機能は SYSBAS ディスク・プールに存在す
| ることも覚えておいてください。

| ネットワーク構成の考慮事項


| ネットワークの配線および構成が、リモート・ミラーリングのパフォーマンスに影響を及ぼすことがありま
| す。ネットワーク・アドレッシングが、データ・ポート IP アドレスの各セットとは異なるサブネットでセ

1 ットアップされるようにするとともに、ネットワーク配線および構成も、同様の方法でセットアップするよ
1 うにしてください。

メトロ・ミラーの計画:

i5/OS の高可用性は、高可用性と災害時回復を提供するメトロ・ミラーをサポートしています。このテクノ
ロジーを使用する高可用性ソリューションを効果的に構成し、管理するには、適切な計画が必要です。

関連情報

 [DS6000 でコピー・サービス機能を使用するためのガイドラインと推奨事項](#)

 [DS8000 でコピー・サービス機能を使用するためのガイドラインと推奨事項](#)


メトロ・ミラーのハードウェア要件:

メトロ・ミラー・テクノロジーを使用する i5/OS 高可用性ソリューションの構成および管理を行うには、
最小ハードウェア要件が満たされていることを確認してください。

推奨される最小ハードウェア要件は、以下のとおりです。

- IBM System Storage™ の外部ストレージ装置 DS6000™ または DS8000™ の少なくとも 1 つ (各システムに接続されているもの) によって地理的に分離された、少なくとも 2 つの System i モデル。外部ストレージ・ユニットの DS6000 および DS8000 が、外部ストレージ用のファイバー・チャンネル接続をサポートする、すべての System i モデルでサポートされていること。
- サポートされるファイバー・チャンネルのアダプターのうち、以下に挙げたもののいずれかが必要です。
 - 2766 2 ギガビット・ファイバー・チャンネル・ディスク・コントローラー PCI
 - 2787 2 ギガビット・ファイバー・チャンネル・ディスク・コントローラー PCI-X
 - 5760 4 ギガビット・ファイバー・ディスク・コントローラー PCI-X
- DS6000 または DS8000 で外部ロード・ソースをサポートするために、新規 IOP が必要になります。
 - SAN ロード・ソースの 2847 PCI-X IOP 機能
- システム・ストレージの適切なディスク・サイジングを、構成前に行っておく必要があります。ソース用のディスク・セット、同様にターゲット用のディスク装置セット、さらに各整合性コピー用に別のセットが必要になります。

関連情報

 [iSeries™ および IBM TotalStorage: i5 の外付けディスク実装ガイド \(英語\)](#)

 [IBM System Storage DS6000 Information Center へようこそ](#)

 [IBM System Storage DS8000 Information Center へようこそ](#)

1 メトロ・ミラーのソフトウェア要件:

1 メトロ・ミラーを使用する i5/OS 高可用性ソリューションを構成する前に、最小ソフトウェア要件が満た
1 されていることを確認してください。

1 メトロ・ミラーの最小ソフトウェア要件は、以下のとおりです。

- 1 • 高可用性ソリューション内の各 System i モデルで i5/OS V6R1 を稼働させて、IBM System i High
1 Availability Solutions Manager (iHASM) ライセンス・プログラム (5761-HAS) で使用できるようにして
1 おく必要があります。

注: 以前のリリース用に、Lab Services のオフラインである IBM Copy Services for System i を使用して、IBM System Storage ソリューションを処理することもできます。複数のプラットフォーム上でグローバル・ミラーを使用している場合、または複数の System i 区画上にグローバル・ミラーをインプリメントする場合は、IBM Copy Services for System i も使用できます。

- IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム (5761-HAS) を、メトロ・ミラーを使用する高可用性ソリューションに属する各システムにインストールしておきます。
- i5/OS オプション 41 HA 切り替え可能リソースをインストールする必要があります。オプション 41 を選択すると、システム間で独立ディスク・プールを切り替えることができます。システム間で独立ディスク・プールを切り替えるには、システムがクラスターのメンバーであり、独立した切り替えディスクがそのクラスターの装置クラスター・リソース・グループと関連付けられている必要があります。また、オプション 41 は、IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラムの一部として提供される高可用性管理インターフェースの処理にも必要です。
- iHASM ライセンス・プログラムでストレージを制御するには、ストレージ・コマンド行インターフェース (DSCLI) も必要になります。DSCLI は、すべての IBM System Storage ソリューションの必須ソフトウェアです。Flashcopy、メトロ・ミラー、グローバル・ミラーなどの IBM System Storage ソリューションを管理するために、これらのストレージ・ソリューションを使用する高可用性ソリューションに属しているシステムまたは区画のそれぞれに DSCLI をインストールする際に、必要となる要件があります。DSCLI の追加のソフトウェア要件は、以下のとおりです。
 - Java™ バージョン 1.4
 - オプション 35 (暗号サービス・プロバイダー) を各システムまたは区画にインストールすること
- 最新の PTF がインストールされていることを確認してください。

関連情報



iSeries™ および IBM TotalStorage: i5 の外付けディスク実装ガイド (英語)



IBM System Storage DS6000 Information Center へようこそ



IBM System Storage DS8000 Information Center へようこそ

メトロ・ミラーの通信要件:

メトロ・ミラーを使用する i5/OS 高可用性ソリューションを構成する前に、最小の通信要件が満たされていることを確認してください。

メトロ・ミラー・テクノロジーを使用するには、ストレージ域ネットワーク (SAN) を既に使用しているか、または使用を計画する必要があります。

SAN は、中央管理された、情報保護型の専用インフラストラクチャーで、システムとストレージ・システム間の any-to-any 型相互接続を可能にします。SAN 接続には、DS8000 外部ストレージ・ユニット、DS6000 外部ストレージ・ユニットなどの、IBM System Storage を使用することが必要です。

メトロ・ミラーを使用する i5/OS 高可用性ソリューションの最小通信要件は、以下のとおりです。

- サポートされるファイバー・チャネルのアダプターのうち、以下に挙げたもののいずれかが必要です。
 - 2766 2 ギガビット・ファイバー・チャネル・ディスク・コントローラー PCI
 - 2787 2 ギガビット・ファイバー・チャネル・ディスク・コントローラー PCI-X
 - 5760 4 ギガビット・ファイバー・ディスク・コントローラー PCI-X

- System i 製品は、さまざまな SAN 切り替えおよびディレクターをサポートしています。サポートされる切り替えおよびディレクターの完全なリストについては、ストレージ・エリア・ネットワーク (SAN) に関する Web サイトを参照してください。
- 回復性およびパフォーマンスの全般を改善するために、マルチパス入出力の利用を強くお勧めします。マルチパス入出力を使用すると、ストレージ内の同一の論理ディスク装置に複数のファイバー・チャネル・デバイスを構成することが可能になります。正しく構成されていれば、デバイス、入出力格納装置、あるいは HSL ループが失敗した場合でも、ディスク装置への接続は失われずに済みます。マルチパスでは、使用可能な接続 (パス) 全体にワークロードを伝搬させることができるので、パフォーマンス上のメリットもあります。複数のディスク装置の各接続は、個別に機能します。複数の接続を使用することで、いずれかのパスで障害が発生してもディスク・ストレージを使用できるため、回復力が向上します。

関連資料



ストレージ・エリア・ネットワーク (SAN) Web サイト (英語)

メトロ・ミラーのジャーナル計画:

すべての高可用性ソリューションのリカバリー時間を増やすために、ジャーナル処理は重要です。IBM システム・ストレージ・ベースのテクノロジー (メトロ・ミラーなど) の場合、データ・ミラーリングが System i ストレージの外側で発生するため、ジャーナリングを使用して外部ストレージ・ユニットに強制的に書き込み操作を行うことが不可欠になります。

ジャーナル管理を行うことにより、システムが異常終了した場合にトランザクションが失われるのを防ぎます。オブジェクトをジャーナル処理すると、そのオブジェクトに加えられた変更をシステムが記録します。ジャーナル処理は、インプリメントする高可用性ソリューションにかかわらず、異常なシステム停止が起きたときにデータ損失を防ぐベスト・プラクティスと見なすことができます。

関連情報

ジャーナル管理

メトロ・ミラーのバックアップ計画:

メトロ・ミラーでは、FlashCopy® を使用することによって、IBM System Storage の外部ストレージ・ユニットに保管されたデータのコピーを作成することができます。

FlashCopy を操作すると、ポイント・イン・タイムのコピーが作成できます。FlashCopy 操作が処理されるとすぐに、ソース・ボリュームおよびターゲット・ボリュームの両方がアプリケーションで使用できるようになります。メトロ・ミラー、グローバル・ミラーなどといった IBM System Storage テクノロジーと一緒に FlashCopy を使用することで、リモート・サイトにあるデータのポイント・イン・タイムのコピーを、整合性が保たれた形で作成することができます。このコピーは、標準のバックアップ手順でバックアップすることができます。FlashCopy をインプリメントする前に、以下を実行してください。

- FlashCopy 関係のソース・ボリュームとターゲット・ボリュームを判別します。パフォーマンスを向上させるには、異なるランクにある FlashCopy のターゲット・ボリュームを選択します。
- FlashCopy でのデータ整合性に関する考慮事項を理解します。この環境では、データがシステム・メモリー・キャッシュに保管されて、しばらくしてからディスクに書き込まれます。こうしたタイプの再始動アクションを防ぐには、FlashCopy ソース・ボリュームに関連したすべてのデータを、FlashCopy 操作を実行する前にディスクに書き込むようにしてください。

- 既存のメトロ・ミラーのソース・ボリュームを、FlashCopy のターゲット・ボリュームとして使用することができます。これにより、FlashCopy ペアのターゲット・ボリュームを使用してポイント・イン・タイムのコピーを作成し、リモート・ロケーションにあるメトロ・ミラーのソース・ボリュームに、そのデータをミラーリングすることができます。

メトロ・ミラーのパフォーマンス計画:

メトロ・ミラーを構成する前に、パフォーマンスに関する考慮事項を理解しておいてください。

メトロ・ミラーを使用する前に、以下の要件およびガイドラインを検討してください。

- メトロ・ミラー関係のソース・ボリュームおよびターゲット・ボリュームは、同じストレージ・タイプでなければなりません。
- ソース論理ボリュームとターゲット論理ボリュームが同じサイズであるか、またはターゲット側のサイズがソース側より大きくなければなりません。
- メトロ・ミラー環境では、共通ボリュームの小さなセットに対するすべての更新内容を、単一のターゲットのストレージ・ユニットに送信しないことにより、ワークロードを分散します。ターゲット・サイトのストレージ・ユニットにおけるパフォーマンスに影響が生じると、ソース・サイトのパフォーマンスにも悪影響が生じます。
- システム・ディスク構成全般に言えることですが、アプリケーションに対して使用可能なディスク装置の数が、パフォーマンスに大きな影響を及ぼす場合があります。限られた数のディスク装置上でワークロードが増えた場合には、ディスク待機が長くなるため、アプリケーションに対する応答時間も長くなります。独立ディスク・プールを使用して構成されたシステム内の一時ストレージの場合、これは特に重要です。一時ストレージは、すべて SYSBAS ディスク・プールに書き込まれます。アプリケーションで使用される一時ストレージがそれほど多くない場合には、SYSBAS ディスク・プール内のディスク・アームが少なくても対処は可能です。また、オペレーティング・システムおよび基本機能は SYSBAS ディスク・プールに存在することも覚えておいてください。

関連情報



DS6000 でコピー・サービス機能を使用するためのガイドラインと推奨事項



DS8000 でコピー・サービス機能を使用するためのガイドラインと推奨事項

グローバル・ミラーの計画:

i5/OS 高可用性はグローバル・ミラーに対応しています。これは、外部ストレージ・ソリューションを使用する環境での高可用性と災害時回復を提供します。このテクノロジーを使用する高可用性を効果的に構成し、管理するには、適切な計画が必要です。

IBM System Storage グローバル・ミラー・テクノロジーでは、すべてのユーザーが 1 つのグローバル・ミラー接続を共有しなければなりません。i5/OS の高可用性グローバル・ミラーでは、所定の System Storage サーバー上のグローバル・ミラー・セッションでアクティブにすることができる System i[™] 区画は 1 つだけです。他の System i 区画や、他のプラットフォームのサーバーは、同時にグローバル・ミラーを使用することはできません。グローバル・ミラー・セッションに複数のユーザーを追加すると、予想不能の結果が生じます。

複数のプラットフォーム上でグローバル・ミラーを使用している場合、または複数の System i 区画上にグローバル・ミラーをインプリメントする場合は、IBM Copy Services for System i を使用できます。このオフリングは Lab Services から入手できます。

関連情報

DS6000 でコピー・サービス機能を使用するためのガイドラインと推奨事項

DS8000 でコピー・サービス機能を使用するためのガイドラインと推奨事項

グローバル・ミラーのハードウェア要件:

グローバル・ミラー・テクノロジーを使用する i5/OS 高可用性ソリューションの構成および管理を行うには、最小ハードウェア要件が満たされていることを確認してください。

グローバル・ミラーを使用するための最小ハードウェア要件は、以下のとおりです。

- IBM System Storage の外部ストレージ装置 DS6000 または DS8000 の少なくとも 1 つ (各システムに接続されているもの) によって地理的に分離された、少なくとも 2 つの System i モデル。外部ストレージ・ユニットの DS6000 および DS8000 が、外部ストレージ用のファイバー・チャネル接続をサポートする、すべての System i モデルでサポートされていること。
- サポートされるファイバー・チャネルのアダプターのうち、以下に挙げたもののいずれかが必要です。
 - 2766 2 ギガビット・ファイバー・チャネル・ディスク・コントローラー PCI
 - 2787 2 ギガビット・ファイバー・チャネル・ディスク・コントローラー PCI-X
 - 5760 4 ギガビット・ファイバー・ディスク・コントローラー PCI-X
- DS6000 または DS8000 で外部ロード・ソースをサポートするために、新規 IOP が必要になります。
 - SAN ロード・ソースの 2847 PCI-X IOP 機能
- システム・ストレージの適切なディスク・サイジングを、構成前に行っておく必要があります。ソース用のディスク・セット、同様にターゲット用のディスク装置セット、さらに各整合性コピー用に別のセットが必要になります。

関連情報

iSeries™ および IBM TotalStorage: i5 の外付けディスク実装ガイド (英語)

IBM System Storage DS6000 Information Center へようこそ

IBM System Storage DS8000 Information Center へようこそ

グローバル・ミラーのソフトウェア要件:

グローバル・ミラーを使用する i5/OS 高可用性ソリューションを構成する前に、最小ソフトウェア要件が満たされていることを確認してください。

グローバル・ミラーの最小ソフトウェア要件は、以下のとおりです。

- 高可用性ソリューション内の各 System i モデルで i5/OS V6R1 を稼働させて、IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム (5761-HAS) で使用できるようにしておく必要があります。

注: 以前のリリース用に、Lab Services のオファリングである IBM Copy Services for System i を使用して、IBM System Storage ソリューションを処理することもできます。複数のプラットフォーム上でグローバル・ミラーを使用している場合、または複数の System i 区画上にグローバル・ミラーをインプリメントする場合は、IBM Copy Services for System i も使用できます。

- IBM System i High Availability Solutions Manager(iHASM)-5761-HAS ライセンス製品が、グローバル・ミラーを使用する高可用性ソリューションに属する各システム上にインストールされていること。

・ iHASM ライセンス・プログラムでストレージを制御するには、ストレージ・コマンド行インターフェース (DSCLI) も必要になります。DSCLI は、すべての IBM System Storage ソリューションの必須ソフトウェアです。Flashcopy、メトロ・ミラー、グローバル・ミラーなどの IBM System Storage ソリューションを管理するために、これらのストレージ・ソリューションを使用する高可用性ソリューションに属しているシステムまたは区画のそれぞれに DSCLI をインストールする際に、必要となる要件があります。DSCLI の追加のソフトウェア要件は、以下のとおりです。

– Java バージョン 1.4

– オプション 35 (暗号サービス・プロバイダー) を各システムまたは区画にインストールすること

・ 最新の PTF がインストールされていることを確認してください。

関連情報



iSeries™ および IBM TotalStorage: i5 の外付けディスク実装ガイド (英語)



IBM System Storage DS6000 Information Center へようこそ



IBM System Storage DS8000 Information Center へようこそ

グローバル・ミラーの通信要件:

グローバル・ミラーを使用する i5/OS 高可用性ソリューションを構成する前に、最小の通信要件が満たされていることを確認してください。

グローバル・ミラー・テクノロジーを使用するには、ストレージ域ネットワーク (SAN) を既に使用しているか、または使用を計画する必要があります。

SAN は、中央管理された、情報保護型の専用インフラストラクチャーで、システムとストレージ・システム間の any-to-any 型相互接続を可能にします。SAN 接続には、DS8000 外部ストレージ・ユニット、DS6000 外部ストレージ・ユニットなどの、IBM System Storage を使用することが必要です。

グローバル・ミラーを使用する i5/OS 高可用性ソリューションの最小通信要件は、次のとおりです。

・ サポートされるファイバー・チャネルのアダプターのうち、以下に挙げたもののいずれかが必要です。

– 2766 2 ギガビット・ファイバー・チャネル・ディスク・コントローラー PCI

– 2787 2 ギガビット・ファイバー・チャネル・ディスク・コントローラー PCI-X

– 5760 4 ギガビット・ファイバー・ディスク・コントローラー PCI-X

・ System i 製品は、さまざまな SAN 切り替えおよびディレクターをサポートしています。サポートされる切り替えおよびディレクターの完全なリストについては、ストレージ・エリア・ネットワーク (SAN) に関する Web サイトを参照してください。

・ 回復性およびパフォーマンスの全般を改善するために、マルチパス入出力の利用を強くお勧めします。マルチパス入出力を使用すると、ストレージ内の同一の論理ディスク装置に複数のファイバー・チャネル・デバイスを構成することが可能になります。正しく構成されていれば、デバイス、入出力格納装置、あるいは HSL ループが失敗した場合でも、ディスク装置への接続は失われずに済み、マルチパスでは、使用可能な接続 (パス) 全体にワークロードを伝搬させることができるので、パフォーマンス上のメリットもあります。複数のディスク装置の各接続は、個別に機能します。複数の接続を使用することで、いずれかのパスで障害が発生してもディスク・ストレージを使用できるため、回復力が向上します。

関連資料



ストレージ・エリア・ネットワーク (SAN) Web サイト (英語)

グローバル・ミラーのジャーナル計画:

すべての高可用性ソリューションのリカバリー時間を増やすために、ジャーナル処理は重要です。IBM System Storage ベースのテクノロジー (グローバル・ミラーなど) の場合、データ・ミラーリングを System i ストレージの外側で発生させるために必要となる外部ストレージ・ユニットに対して、ジャーナル処理によって強制的に書き込み操作が行われます。

ジャーナル管理を行うことにより、システムが異常終了した場合にトランザクションが失われるのを防ぎます。オブジェクトをジャーナル処理すると、そのオブジェクトに加えられた変更をシステムが記録します。ジャーナル処理は、インプリメントする高可用性ソリューションにかかわらず、異常なシステム停止が起きたときにデータ損失を防ぐベスト・プラクティスと見なすことができます。

関連情報

ジャーナル管理

グローバル・ミラーのバックアップ計画:

高可用性ソリューション内でグローバル・ミラー・テクノロジーを使用する場合、FlashCopy を使用して、データのポイント・イン・タイム・コピーを作成することができます。

FlashCopy を操作すると、ポイント・イン・タイムのコピーが作成できます。FlashCopy 操作が処理されるとすぐに、ソース・ボリュームおよびターゲット・ボリュームの両方がアプリケーションで使用できるようになります。メトロ・ミラー、グローバル・ミラーなどといった IBM System Storage テクノロジーと一緒に FlashCopy を使用することで、リモート・サイトにあるデータのポイント・イン・タイムのコピーを、整合性が保たれた形で作成することができます。このコピーは、標準のバックアップ手順でバックアップすることができます。FlashCopy をインプリメントする前に、以下を実行してください。

- FlashCopy 関係のソース・ボリュームとターゲット・ボリュームを判別します。パフォーマンスを向上させるには、異なるランクにある FlashCopy のターゲット・ボリュームを選択します。
- FlashCopy でのデータ整合性に関する考慮事項を理解します。この環境では、データがシステム・メモリー・キャッシュに保管されて、しばらくしてからディスクに書き込まれます。こうしたタイプの再始動アクションを防ぐには、FlashCopy ソース・ボリュームに関連したすべてのデータを、FlashCopy 操作を実行する前にディスクに書き込むようにしてください。

グローバル・ミラーのパフォーマンス計画:

グローバル・ミラーを構成する前に、パフォーマンスに関する考慮事項を理解しておいてください。

メトロ・ミラーを使用する前に、以下のパフォーマンスのガイドラインを検討してください。

- メトロ・ミラー関係のソース・ボリュームおよびターゲット・ボリュームは、同じストレージ・タイプでなければなりません。
- メトロ・ミラー関係のソース・ボリュームおよびターゲット・ボリュームは、同じストレージ・タイプでなければなりません。
- システム・ディスク構成全般に言えることですが、アプリケーションに対して使用可能なディスク装置の数が、パフォーマンスに大きな影響を及ぼす場合があります。限られた数のディスク装置上でワークロードが増えた場合には、ディスク待機が長くなるため、アプリケーションに対する応答時間も長くなります。独立ディスク・プールを使用して構成されたシステム内の一時ストレージの場合、これは特に重要です。一時ストレージは、すべて SYSBAS ディスク・プールに書き込まれます。アプリケーションで使用される一時ストレージがそれほど多くない場合には、SYSBAS ディスク・プール内のディスク・

1 アームが少なくても対処は可能です。また、オペレーティング・システムおよび基本機能は SYSBAS デ
1 イスク・プールに存在することも覚えておいてください。

1 関連情報

1  DS6000 でコピー・サービス機能を使用するためのガイドラインと推奨事項

1  DS8000 でコピー・サービス機能を使用するためのガイドラインと推奨事項

論理複製の計画

データの複数のコピーは、論理複製を使って保守されます。データは、クラスターのプライマリー・ノードからリカバリー・ドメインにある指定されたバックアップ・ノードに複製つまりコピーされます。プライマリー・ノードで障害が発生したときには、指定されたバックアップ・ノードが 1 次アクセス・ポイントを引き継ぐので、データは使用可能なままです。

論理複製とは、何かのコピーをリアルタイムで作成することです。つまり、クラスター内のあるノードに含まれているオブジェクトを、同じクラスター内にある他の 1 つ以上のノードにコピーする処理のことです。論理複製は、システム上にまったく同じオブジェクトを作成し、保管します。クラスター内の特定のノードに含まれているオブジェクトに変更を加えると、その変更は同じクラスター内にある他のノードにも複製されます。

論理複製に使用するソフトウェア・テクノロジーを決めなければなりません。クラスターで論理複製を行うときは、次のソリューションを使用することができます。

• IBM ビジネス・パートナーのプロダクト

承認を受けたクラスター IBM ビジネス・パートナー製のデータ複製ソフトウェアを使って、複数のノードに対してオブジェクトを複製することができます。

• ユーザー作成複製アプリケーション

IBM ジャーナル管理は、システムのオブジェクトの活動を記録する手段となります。ジャーナル管理の利点をいかして、論理複製を実現するアプリケーションを作成することができます。

関連情報

ジャーナル管理

論理複製に使用するシステムの決定:

論理複製にどのシステムを使用するかは決定では、いくつかの考慮事項があります。

そのような考慮事項には、次のものがあります。

- パフォーマンス容量
- ディスク容量
- 重要なデータ
- 災害時対策

システムがフェイルオーバーした場合には、プライマリー・システムおよびバックアップ・システムで、実行していたデータおよびアプリケーションは何であったかを、把握しなければなりません。フェイルオーバーが実行される事態を想定すると、最も処理能力の高いシステムに重要なデータを置きたいと思いかもありません。とはいえ、ディスク・スペースがいっぱいになっても困ります。プライマリー・システムのスペースに余裕がなくなってフェイルオーバーが実行されると、ディスク・スペースの不足のため、バックアッ

プ・システムもフェイルオーバーが実行される可能性が非常に高くなります。洪水、台風、竜巻などの自然災害によってデータ・センターが完全に破壊されてしまうことがないように、複製の対象となるシステムは別の遠隔地に設置するのが最善です。

クラスター・ミドルウェアの IBM ビジネス・パートナーおよび使用可能なクラスタリング・プロダクト:

IBM 管理ソリューションに加え、論理複製テクノロジーを使用する高可用性ビジネス・パートナーのクラスター・ミドルウェア製品を購入することもできます。

- | IBM クラスター・ミドルウェアのビジネス・パートナーは、複製専用のおよびクラスター管理機能用のソフトウェア・ソリューションを提供します。ほとんどのビジネス・パートナー・ソリューションは論理複製に基づいています。論理複製は、オブジェクトをコピーし、リアルタイムでレコード・レベルの変更を行います。つまり、クラスター内のあるノードに含まれているオブジェクトを、同じクラスター内にある他の1つ以上のノードにコピーする処理のことです。複製を行えば、システム上にまったく同じオブジェクトを作成したり保管したりできます。クラスター内の特定のノードに含まれているオブジェクトに変更を加えると、その変更は同じクラスター内にある他のノードにも複製されます。

論理複製のジャーナル処理計画:

論理複製を使用している場合は、ジャーナル処理を使用して、データの実動コピーからデータのバックアップ・コピーに、強制的に書き込みを行ってください。

ジャーナル管理を行うことにより、システムが異常終了した場合にトランザクションが失われるのを防ぎます。オブジェクトをジャーナル処理すると、そのオブジェクトに加えられた変更をシステムが記録します。ジャーナル処理は、インプリメントする高可用性ソリューションにかかわらず、異常なシステム停止が起きたときにデータ損失を防ぐベスト・プラクティスと見なすことができます。

論理複製環境では、ジャーナル処理がソリューションの基本となるため、このテクノロジーを基礎とするソリューションをインプリメントするための要件になります。論理複製を使用すると、複製するオブジェクトのサイズにより、バックアップ・システムへのリアルタイム・コピーが制限されることがあります。例えば、ジャーナルしたファイル内には、レコードを更新するプログラムが常駐しています。このプログラムは、同じ操作の一環として、ジャーナル処理されていないオブジェクト（ユーザー・スペースなど）も更新します。ユーザー・スペースがバックアップ・システムに完全に複製されるときに、バックアップ・コピーの整合性が完全に確認されます。実際のところ、1次システムに障害が発生した際に、ユーザー・スペースのオブジェクトが完全に複製されていない場合、完全に複製されたデータのある最新の有効な操作と一致するようユーザー・スペースの状態を調整するには、手動のリカバリー・プロセスが必要になります。

関連情報

ジャーナル管理

論理複製のバックアップ計画:

サード・パーティー製アプリケーションが提供する論理複製テクノロジーを使用している場合、この環境内での操作のバックアップを計画してください。

論理複製は、実動コピーのファイルまたはプログラムなどのオブジェクトへの変更内容を、バックアップ・コピーに複製します。複製は、ほぼリアルタイム（同時）で行われます。通常、ファイルなどのオブジェクトがジャーナル処理される場合、複製は記録レベルで処理されます。このテクノロジーの主な利点は、バックアップ・コピーにリアルタイムでアクセスして、バックアップ操作を行えることです。データの実動バージョンを妨げることなしに、データのバックアップ・コピーのリモート・バックアップを実行することができます。

論理複製のパフォーマンス計画:

高可用性ソリューションの一環として論理複製テクノロジーを使用するには、このソリューションでのパフォーマンスに対する潜在的な影響について理解しておいてください。

論理複製を使用すると、複製プロセスの待ち時間によって、パフォーマンスに影響が生じる場合があります。これには、起動システムに変更が加えられた時間と、これらの変更がバックアップ・システムで有効になる時間との時間差が関係しています。同期リモート・ジャーナル処理を使用することで、この時間差を大幅に縮めることができます。使用された送信メカニズムにかかわらず、ご使用の環境がピーク時の場合でも複製ボリュームを管理できるように、送信ボリュームの予想を適切に行って、通信回線および通信速度を計画する必要があります。大容量の環境では、送信機能が適切に計画されていても、リプレイのバックログおよび待ち時間がターゲット側で問題になることがあります。

環境回復力の計画

環境の回復力は、高可用性環境において定義されたリソース間で、オブジェクトと属性の整合が保たれることを保証します。適切に整合のとれた環境機能がどのリソースで必要とされているかを識別し、これらのリソース属性が高可用性ソリューションにおいて整合性を保持することを保証するクラスター管理可能ドメインを作成する必要があります。

クラスター管理可能ドメインの計画

クラスター管理可能ドメインには、クラスター管理可能ドメイン内のノードで同期されるリソースを管理するための計画が必要です。アプリケーションがいかなるシステムにおいて高可用性環境で一貫して実行されるようにするために、アプリケーションの振る舞いに影響を与えるすべてのリソースを判別する必要があります。同様に、アプリケーションが実行される、またはアプリケーション・データが存在するクラスター・ノードの識別も必要です。

クラスター管理者は、クラスター管理可能ドメインを作成してから、各ノードによって同期されるモニター対象リソースを追加することができます。i5/OS クラスターは、クラスター管理可能ドメインにより同期することができるシステム・リソースのリストを提供しています。これは、モニター対象リソース項目 (MRE) で表されます。

クラスター管理可能ドメインを設計するときは、次のような設問に対する答を出す必要があります。

クラスター管理可能ドメインにどのノードを組み込むか。

クラスター内のどのノードがクラスター管理可能ドメインによって管理されるかを定める必要があります。これらは、アプリケーションが実行できるシステム、またはアプリケーション・データが保管されているシステムを表すクラスター・ノードであり、整合した操作環境が必要です。複数のクラスター管理可能ドメインにノードを収容することはできません。例えば、クラスターに 4 つのノードがある場合 (ノード A、ノード B、ノード C およびノード D)、ノード A と B を特定のクラスター管理可能ドメインに配置し、ノード C と D を別のドメインに収容することが可能です。ただし、ノード B と C が 3 つ目のクラスター管理可能ドメインにあるのに、元のクラスター管理可能ドメインにも収容することはできません。

クラスター管理可能ドメインの命名規則をどうするか。

クラスター環境の複雑さとサイズによっては、対等 CRG とクラスター管理可能ドメインに対して標準命名規則を設定したほうがよい場合があります。クラスター管理可能ドメインを作成するときに対等 CRG を作成しているため、ユーザーはクラスター管理可能ドメインを表す CRG から他の対等 CRG を差別化したいと考えます。例えば、クラスター管理可能ドメインを表す対等 CRG を *ADMDMNI* や *ADMDMN2* などと命名し、他の対等 CRG を *PEERI* と命名することができます。また、クラスター・リソース・グループ情報のリスト (*QcstListClusterResourceGroupIn*) API を使っ

て、対等 CRG がクラスター管理可能ドメインとして使用されているかどうかを判別することができます。クラスター管理可能ドメインを表す対等 CRG は、そのアプリケーション識別子である QIBM.AdminDomain によって識別することができます。

モニター対象リソース項目 (MRE) の計画

モニター対象リソースは、クラスター管理可能ドメイン内で定義できる i5/OS オブジェクトです。これらのリソースは、高可用性環境のシステム全体で整合されている必要があります。整合が保たれていない場合、障害時にアプリケーションが期待通りの性能を発揮しないことがあります。ご使用の環境内で、サポートされているどのリソースをモニターすべきかを計画する必要があります。

どのシステム・リソースを同期する必要があるかを判別する必要があります。これらのリソースのそれぞれに属性を選択し、同期するものをカスタマイズすることができます。複数ノードで実行されるアプリケーションは、正しく稼働するために特定の環境変数が必要な場合があります。また、いくつかのノード上に散在するデータの場合も、特定のユーザー・プロファイルへのアクセスが必要になることがあります。クラスター管理可能ドメインにより、どのリソースを管理する必要があるかを決めるときは、事前にアプリケーションとデータの操作上の要件に注意してください。

クラスターの計画

高可用性ソリューションをインプリメントする前に、クラスターのすべての前提条件を満たしていることを確認する必要があります。

クラスターのハードウェア要件

高可用性ソリューションをインプリメントするには、クラスターを計画し、構成する必要があります。クラスターは、高可用性環境でシステムとリソースをグループ化します。

以下は、クラスターのハードウェアの必要最小要件です。

- 少なくとも 2 つの System i モデルまたは論理区画が必要になります。クラスターは、1 クラスター内に最高 128 のシステムをサポートします。i5/OS V4R4M0 以降を実行できるすべての System i モデルは、クラスターリング使用と互換性があります。
- クラスター区画を引き起こす可能性のある、突然の停電から保護するために、外部無停電電源装置またはその相当品が推奨されます。
- クラスターリングを実施する際には、インターネット・プロトコル (IP) マルチキャスト機能を使用します。物理メディアによっては、マルチキャストが正常にマップを行わない場合があります。
- 独立ディスク・プールを必要とするデータ回復テクノロジーの使用を計画している場合は、選択したデータ回復テクノロジー固有のハードウェアについて計画する必要もあります。保護ディスクに障害が発生した場合にフェイルオーバーが発生するのを防ぐため、ディスク保護に別の方法を使用することもできます。

関連概念

49 ページの『データ回復力の計画』

データ回復力は、データがユーザーやアプリケーションに使用可能になる機能です。データ回復力は切り替えディスク、サイト間ミラーリング、または論理複製テクノロジーのいずれかの i5/OS クラスター・テクノロジーを使用することによって実現できます。

関連資料

74 ページの『クラスターの計画チェックリスト』

クラスター構成チェックリストを完了して、環境が正しく準備されていることを確認してから、クラスターの構成を開始してください。

関連情報

無停電電源装置

IP マルチキャストリング

ディスク保護

クラスタのソフトウェア要件

クラスタリングを使用するためには、正しいソフトウェアとライセンスが必要です。

1. i5/OS V6R1 がインストール済み。
2. TCP/IP 接続ユーティリティー機能がインストール済み。
3. 切り替えディスクやサイト間ミラーリングなどのデータ回復テクノロジーの使用を計画している場合は、追加の要件があります。
4. 次のインターフェースの使用を計画している場合には、オプション 41 (高可用性切り替え可能リソース) が必要です。
 - System i Navigator Cluster Management インターフェース

注: System i Navigator Cluster Management インターフェースの処理については、i5/OS V5R4 Information Center の『クラスタ』を参照してください。

- IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム。このライセンス・プログラムは、オプション 41 を必要とする次のインターフェースを提供します。
 - 高可用性ソリューション・マネージャーのグラフィカル・インターフェース
 - クラスタ・リソース・サービスのグラフィカル・インターフェース
 - IBM System i High Availability Solutions Manager (iHASM) コマンド
 - IBM System i High Availability Solutions Manager (iHASM) API

注:

5. また、IBM ビジネス・パートナー・プロダクトを使用したり、クラスタ API を使用して独自の高可用性管理アプリケーションを作成することもできます。

関連概念

50 ページの『切り替えディスクの計画』

データの単一コピーが、切り替え可能なハードウェア (拡張装置 (タワー) または論理区画環境の IOP のいずれか) で保守されます。

52 ページの『サイト間ミラーリングの計画』

サイト間ミラーリングは、リモート・ミラーリング、メトロ・ミラーリング、およびグローバル・ミラーなどの複数の i5/OS 災害時回復および高可用性テクノロジーを提供します。

49 ページの『データ回復力の計画』

データ回復力は、データがユーザーやアプリケーションに使用可能になる機能です。データ回復力は切り替えディスク、サイト間ミラーリング、または論理複製テクノロジーのいずれかの i5/OS クラスタ・テクノロジーを使用することによって実現できます。

関連資料

74 ページの『クラスタの計画チェックリスト』

クラスタ構成チェックリストを完了して、環境が正しく準備されていることを確認してから、クラスタの構成を開始してください。

関連情報

クラスタ API

クラスタの通信要件

任意のタイプの通信メディアをクラスタリング環境で使用します。ただし、そのタイプのメディアがインターネット・プロトコル (IP) をサポートしていることを前提とします。

クラスタ・リソース・サービスは TCP/IP および UDP/IP プロトコルを使用して、ノード間の通信を行います。ローカル・エリア・ネットワーク (LAN)、広域ネットワーク (WAN)、OptiConnect システム・エリア・ネットワーク (SAN)、またはこれらの接続装置の組み合わせがサポートされています。以下の要因に基づいて選択を行ってください。

- トランザクションの量
- 応答時間の要件
- ノード間の距離
- コストについての考慮事項

これらの考慮事項は、リソースのプライマリー・ロケーションと、バックアップ・ロケーションとを接続するために使用される接続メディアを決定する際に当てはまります。クラスタの計画段階では、サイトが失われるような災害を切り抜けるために、リモート・ロケーションにある 1 つ以上のバックアップ・ノードを指定するようお勧めします。

容量の不足が原因で生じ得るパフォーマンスの問題を回避するためには、ノードからノードへ送られる多量の情報を処理するのに使用されている通信メディアを評価する必要があります。トークンリング、イーサネット、非同期転送モード (ATM)、SPD OptiConnect、高速リンク (HSL) OptiConnect、仮想 OptiConnect (論理区画間的高速内部接続) など、使用したい物理メディアを選択することができます。

HSL OptiConnect は、i5/OS ソフトウェア (i5/OS オプション 23 - i5/OS OptiConnect) に備わったテクノロジーです。これを使用して、高可用性ソリューションを構成できます。HSL OptiConnect は、高速リンク (HSL) ループ・テクノロジーを使用してクラスタ・ノード間的高速、2 地点間接続を提供するシステム・エリア・ネットワークです。HSL OptiConnect は、標準 HSL ケーブルを必要としますが、他のハードウェアは必要としません。

切り替え可能ハードウェア (回復力の高い装置 CRG と呼ばれる) の場合は、環境の中に切り替えディスクが必要になります。論理区画環境において、これは、論理区画で共用されるバス上にあるディスク装置のコレクション、または 1 つの入出力プールに割り当てられた入出力プロセッサに接続されたディスク装置のコレクションです。複数システム環境では、リカバリー・ドメインにあるシステムも含む HSL ループ上に適切に構成された 1 つ以上の切り替え可能な拡張装置 (タワー) になります。切り替え可能な拡張装置は、LPAR 環境でも使用できます。

注: TCP/IP のみを使用し、Systems Network Architecture (SNA) や IPX を使用しない 2810 LAN アダプターを使っている場合は、回線記述の処理 (WRKLIND) コマンドを使用して、特定の回線記述に「TCP/IP の場合に使用可能 (*YES)」を指定することにより、OS/400® V4R5M0 システムのアダプター性能を向上することができます。「TCP の場合に使用可能 (*YES)」は OS/400 V5R1M0 以降のリリースでは自動的に設定されます。

関連概念

50 ページの『切り替えディスクの計画』

データの単一コピーが、切り替え可能なハードウェア (拡張装置 (タワー) または論理区画環境の IOP のいずれか) で保守されます。

関連資料

74 ページの『クラスタの計画チェックリスト』

クラスタ構成チェックリストを完了して、環境が正しく準備されていることを確認してから、クラスタの構成を開始してください。

クラスタに使用するネットワークの専用化:

通常の操作時は、基本的なクラスタリング通信トラフィックは最小限に抑えられています。とはいえ、クラスタ内のノードごとに冗長通信パスを構成するよう強くお勧めします。

冗長通信パスを構成することは、クラスタ内の 2 つのノードの間に 2 本の回線を構成することを意味します。最初の通信パスに障害が発生しても、ノード間で実行されている通信を 2 番目の通信パスが引き継ぐため、クラスタ内の 1 つ以上のノードがクラスタ区画に入る可能性を最小限に抑えることができます。これらのパスを構成する際に考慮すべき事柄は、たとえ 2 本の通信回線を用意したとしても、それらの回線をどちらもシステム上の同じアダプターに接続してしまうのであれば、障害がそのアダプターで発生した場合に、それらの回線が 2 本とも使用不可になる危険性があるということです。ただし、クラスタ区画を常に阻止できるとは限らないことに注意しなければなりません。システムで停電が発生した場合や、ハードウェア障害が発生した場合、クラスタが分割される可能性があります。2 つの回線を構成すると、一方の回線をクラスタリング・トラフィック専用、もう一方の回線は通常のトラフィック用と、クラスタリング用の専用回線がダウンした場合のバックアップ回線にすることもできます。典型的なネットワーク関連クラスタ区画が発生しないようにする最も有効な手段としては、クラスタ内のすべてのノードを結び付ける冗長通信パスを構成します。

ヒント: クラスタ通信:

通信パスをセットアップする際には、以下のヒントを参考にしてください。

- クラスタリング・ハートビート機能に関係した非クラスタ活動を処理し、増加する活動のモニターを継続して行えるように、通信回線には十分な帯域幅を持たせてください。
- 最善の信頼性を得るために、1 つ以上のノードにリンクしている通信パスを、1 つだけ構成することは避けてください。
- ノードとの通信を維持するために使用する回線に、過度の負担をかけないでください。
- 単一機器の故障がシステム全体の故障となるような構成 (単一のアダプター、同一の入出力プロセッサ (IOP)、または同一の拡張装置に 2 本の通信回線が接続するような構成など) は、できるだけ避けてください。
- 通信回線を通して極端に大量のデータを送信する場合は、データ複製とハートビート・モニターを別々のネットワークで行ってください。
- 望ましいプロトコルはユーザー・データグラム・プロトコル (UDP) マルチキャストで、クラスタ通信インフラストラクチャーがクラスタ内のノード間でクラスタ管理情報を送信するために使用します。物理メディアがマルチキャスト機能をサポートする場合、クラスタ通信は UDP マルチキャストを使用して、管理メッセージを特定のノードから、同じサブネット・アドレスをサポートするローカル・クラスタ・ノードすべてに送信します。リモート・ネットワーク上のノードに送信されるメッセージは、必ず UDP 2 地点間機能を使って送信されます。マルチキャスト・メッセージの場合、クラスタ通信はルーティング機能に依存しません。
- クラスタ管理メッセージをサポートするマルチキャスト・トラフィックは、変動しやすい性質を持っています。特定の LAN (共通サブネット・アドレスをサポートするもの) 上のノード数、およびクラスタ管理者が選択したクラスタ管理構造の複雑さによっては、クラスタに関連したマルチキャスト・パケットは毎秒 40 パケットを超えることも珍しくありません。このような変動は、旧式のネットワーク装置に悪影響を与えることがあります。一例として、すべての UDP マルチキャスト・パケットを評価しなければならない Simple Network Management Protocol (SNMP) エージェントの役目を果たす

LAN における装置上の問題があります。旧式のネットワーク装置の中には、この種のトラフィックに見合う適切な帯域幅を持っていないものがあります。ユーザーまたはネットワーク管理者は、UDP マルチキャスト・トラフィックを処理するためのネットワーク容量を検討し、クラスタリングによってネットワークのパフォーマンスが低下しないようにする必要があります。

クラスタのパフォーマンス計画:

それぞれの通信環境には大きな違いがあり得るので、クラスタ通信に影響を与える変数を各環境に合わせて最適な値に調整するための機能が用意されています。

デフォルト値は、一般的な環境のほとんどの適用されますが、デフォルト値が合わない環境の場合は、それぞれの環境に合わせてクラスタ通信を調整できます。初級レベルの調整と上級レベルの調整を使用することができます。

初級レベルの調整

初級レベルの調整。タイムアウトとメッセージ送信間隔については、高、中、低のレベルの値がそれぞれ事前に定義されているので、その事前定義の値に調整パラメーターを設定できます。中レベルを選択すると、クラスタ通信のパフォーマンス・パラメーターと構成パラメーターには、デフォルト値が使用されます。低レベルを選択した場合は、クラスタリング機能のハートビート間隔値と各種メッセージ・タイムアウト値が増えます。ハートビートの頻度が低くなり、タイムアウト時間が長くなると、クラスタによる通信障害の検出能力は低下します。高レベルを選択した場合は、クラスタリング機能のハートビート間隔値と各種メッセージ・タイムアウト値が減ります。ハートビートの頻度が高くなり、タイムアウト時間が短くなると、クラスタによる通信障害の検出能力は向上します。

上級レベルの調整

上級レベルの調整では、事前定義済みの値の範囲を使用して、個別のパラメーターを調整することができます。したがって、通信環境内のいろいろな特殊事情に合わせて、きめの細かい調整ができます。上級レベルの調整を望む場合は、IBM サポート担当者などの援助を受けることをお勧めします。個々のパラメーターの設定が不適切であれば、パフォーマンスの低下につながります。

調整可能なクラスタ通信パラメーター:

クラスタ・リソース・サービス変更 (QcstChgClusterResourceServices) API を使用すれば、クラスタ・トポロジー・サービスおよびクラスタ通信のパフォーマンス・パラメーターと、構成パラメーターの一部を、クラスタリングが行われる多数の固有なアプリケーション環境およびネットワーク環境により良く適合するように調整することができます。

クラスタの変更 (CHGCLU) コマンドは基本レベルのチューニングを行いますが、QcstChgClusterResourceServices API は基本レベルと拡張レベルのどちらのチューニングも行うことができます。

QcstChgClusterResourceServices API と クラスタ構成変更 (CHGCLUCFG) コマンドは、クラスタのパフォーマンスと構成を調整するために使用できます。これらの API とコマンドは、基本レベルの調整をサポートしています。このレベルの調整では、タイムアウトとメッセージ送信間隔について、高、中、低のレベルの値がそれぞれ事前に定義されているので、クラスタをその事前定義の値に調整できます。拡張レベルの調整が必要な場合は、API を使用して個々のパラメーターを事前定義値の範囲で調整できます (通常は、IBM サポート担当員の指示を仰いでください)。個々のパラメーターに不適切な変更を行うと、クラスタのパフォーマンスが低下することがあります。

クラスター・パラメーターを調整する時期と方法

CHGCLU コマンドと QcstChgClusterResourceServices API には、詳細を理解していなくてもクラスターのパフォーマンスと構成パラメーターを設定できるよう、高速バスが用意されています。この基本レベルの調整は主に、ハートビート感度およびクラスター・メッセージ・タイムアウト値に影響します。基本レベルの調整サポートの有効値は以下のとおりです。

1 (高タイムアウト値/より低頻度のハートビート)

クラスター通信に調整を行って、ハートビート頻度を低くし、各種のメッセージ・タイムアウト値を大きくします。ハートビートの頻度が低くなり、タイムアウト時間が長くなると、通信障害に対するクラスターの対応が遅くなる (クラスターの検出能力が落ちる) ことになります。

2 (デフォルト値)

クラスター通信のパフォーマンス・パフォーマンスおよび構成パラメーターに、通常のデフォルト値を使用します。この設定値を使用すると、すべてのパラメーターをオリジナルのデフォルト値に戻すことができます。

3 (低タイムアウト値/より高頻度のハートビート)

ハートビート間隔を小さくし、各種のメッセージ・タイムアウト値を小さくするように、クラスター通信に調整を行います。ハートビートをより頻繁にし、タイムアウト値を短くすると、クラスターは、通信障害に対する対応が早く (感度が高く) なります。

ノードの区画化が発生するようなハートビート障害の場合について、応答時間の例を次の表に示します。

注: 時刻は、分:秒のフォーマットで指定します。

	1 (より低感度)			2 (デフォルト)			3 (より高感度)		
	ハートビート問題の検出	分析	合計	ハートビート問題の検出	分析	合計	ハートビート問題の検出	分析	合計
単一サブネット	00:24	01:02	01:26	00:12	00:30	00:42	00:04	00:14	00:18
複数サブネット	00:24	08:30	08:54	00:12	04:14	04:26	00:04	02:02	02:06

通常のネットワーク負荷および使用する特定の物理メディアに応じて、クラスター管理者は、ハートビート感度およびメッセージ・タイムアウトのレベルの調整を行う場合があります。高速で信頼性の高いトランスポート、たとえば、共通 OptiConnect バス上にクラスターのすべてのシステムをもつ OptiConnect などでは、迅速な検出によってより早くフェイルオーバーを実行させるため、さらに高感度の環境を確立したいことがあります。この場合、オプション 3 を選択します。負荷が大きい 10Mbps のイーサネット・バスで実行しているときに、ネットワーク・ピーク・ロード (デフォルト設定によって発生) が原因で区画化が時折起きてしまう場合は、オプション 1 を選択してください。これによって、ピーク・ロードに対するクラスタリングの感度を軽減することができます。

クラスター・リソース・サービス変更 API を使用すれば、個々のネットワーク環境が持つ固有の状態に合わせて、特定のパラメーターを個別に調整することも可能です。たとえば、OptiConnect バスに共通なすべてのノードをもつクラスターをもう一度考えてみましょう。メッセージ・フラグメント・サイズのパラメーターを最大の 32,500 バイトに設定して、OptiConnect の最大伝送単位 (MTU) サイズにできるだけ一致させることで、デフォルトの 1,464 バイトの場合よりもクラスター・メッセージのパフォーマンスを大幅に向上させることができます。これによって、大きなメッセージのフラグメント化および再組み立てのオーバーヘッドが削減されます。当然、この利点は、クラスター・アプリケーションと、それらのアプリケーション

ンの結果行われるクラスター・メッセージングの使用量に応じて変わってきます。その他のパラメーターは API 資料に定義されています。これらのパラメーターを使用すると、クラスター・メッセージングのパフォーマンスを調整したり、区画化に対するクラスターの感度を変更したりすることができます。

関連資料

QcstChgClusterResourceServices API (英語)

関連情報

クラスターの変更 (CHGCLU) コマンド (英語)

クラスター・リソース・サービスの設定の変更:

メッセージ・タイムアウトと再試行に影響するデフォルト値は、最も典型的なインストール・システムに合わせて設定されています。しかし、自分の通信環境によりしっかりと適合するように、これらの値を変更することが可能です。

値は次のどちらかの方法で調整することができます。

- ご使用の環境に合った、一般的なパフォーマンス・レベルに設定する
- 特定のメッセージ調整パラメーターの値を設定して、具体的な調整を行う

最初の方式では、メッセージ・トラフィックが 3 つの通信レベルのうち 1 つに調整されます。通常レベルはデフォルトであり、これについてはハートビート・モニターで詳細に説明されています。

2 番目の方式は、通常、専門家の助言が受けられる場合に限り行うべきです。

クラスター・リソース・サービス変更 (QcstChgClusterResourceServices) API では、両方の方式を詳細に説明しています。

関連資料

QcstChgClusterResourceServices API (英語)

関連情報

ハートビート・モニター

複数リリース・クラスターの計画:

複数のクラスター・バージョンのノードを含むクラスターを作成する場合は、クラスター作成時に特定のステップが必要です。

デフォルトでは、現行クラスター・バージョンにはクラスターに最初に追加されたノードの潜在クラスター・バージョンが設定されます。このノードがクラスターに存在する最も低いバージョン・レベルである場合、この方法は適切であると言えます。しかし、このノードがそれよりも新しいバージョン・レベルであった場合、その後、それよりも低いバージョン・レベルのノードは追加できません。代わりに、クラスターの作成時のターゲット・クラスター・バージョン値を使用して、現行クラスター・バージョンを、クラスターに最初に追加されたノードの潜在クラスター・バージョンよりも 1 低い値に設定します。

- 1 注: IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラムを使用している場合は、クラスター内のすべてのシステムに V6R1 が必要です。

たとえば、2 つのノードからなるクラスターを作成する場合は考慮してみましょう。このクラスターのノードは、以下に従います。

ノード ID	リリース	潜在クラスター・バージョン
ノード A	V5R4	5
ノード B	V6R1	6

クラスターがノード B から作成された場合は、これが混合リリース・クラスターになることを必ず示すようにしてください。ターゲット・クラスター・バージョンは、通信を要求するノードの潜在ノード・バージョンよりも 1 小さいバージョンでクラスターのノードが通信することを示すように設定されなければなりません。

クラスターのパフォーマンス計画

クラスターに変更を加えると、クラスターを管理するのに必要なオーバーヘッドに影響を与える可能性があります。

クラスタリングが必要とするリソースは、ハートビート・モニターを実行すること、クラスター・リソース・グループとクラスター・ノードを管理すること、およびクラスター・リソース・グループとクラスター・ノードとの間で行われるメッセージ交換を処理することだけです。クラスタリング環境が操作可能になった後にオーバーヘッドが増大するのは、クラスターに変更を加えたときだけです。

通常の稼働環境においては、クラスタリング・アクティビティーがクラスター・システムに与える影響は、最低限で済むはずですが。

クラスターの計画チェックリスト

クラスター構成チェックリストを完了して、環境が正しく準備されていることを確認してから、クラスターの構成を開始してください。

表 1. クラスターの TCP/IP 構成チェックリスト

TCP/IP 要件	
—	クラスターに組み込むすべてのノードで、TCP/IP の開始 (STRTCP) コマンドを使用して TCP/IP を開始します。
—	TCP ループバック・アドレス (127.0.0.1) を構成し、状況がアクティブであることを確認します。クラスター内のすべてのノードで TCP/IP ネットワーク状況の処理 (WRKTCPSTS) コマンドを使用して、TCP/IP ループバック・アドレスを検査してください。
—	対象ノードで TCP/IP ネットワーク状況の処理 (WRKTCPSTS) コマンドを使用して、所定のノードのクラスタリングに使用した IP アドレスの状況がアクティブになっていることを確認してください。

表1. クラスターの TCP/IP 構成チェックリスト (続き)

TCP/IP 要件	
—	<p>クラスターのすべてのノードで INETD がアクティブになっていることを確認します。STRTCPSVR *INETD コマンドを使用するか、または IBM Systems Director Navigator for i5/OS で以下のステップを実行してください。</p> <ol style="list-style-type: none"> 1. Web ブラウザーで、http://mysystem:2001 と入力します。mysystem はシステムのホスト名です。 2. ご使用のユーザー・プロファイルとパスワードで、システムにログオンします。 3. IBM Systems Director Navigator for i5/OS で「ネットワーク」を選択します。 4. 「ようこそ」ページで、「TCP/IP サーバー」を選択します。 5. 「TCP/IP サーバー (TCP/IP Server)」ページで、「TCP/IP サーバー」を選択します。使用可能な TCP/IP サーバーのリストが表示されます。 6. リストから「INETD」を選択します。 7. 「アクションの選択」メニューから、「開始」を選択します。サーバーの状況が「開始済み」に変更されず。 <p>INETD サーバーを開始するには、TCP/IP サーバーの開始 (STRTCPSVR) コマンドを使用して、*INETD パラメーターを指定してもかまいません。これを検証するには、対象ノードのアクティブなジョブ・リストに、ユーザー QTCP (QTOGINTD) ジョブが存在することを確認します。</p>
—	<p>/QIBM/ProdData/OS400/INETD/inetd.config に指定されている INETD のユーザー・プロファイルが、最低限のものより大きい権限をもっていないことを確認します。このユーザー・プロファイルの権限が最低限の権限を超えている場合、クラスター・ノードの開始は失敗します。デフォルトでは、INETD のユーザー・プロファイルとして QUSER が指定されます。</p>
—	<p>クラスター内の各 IP アドレスが、UDP データグラムへの経路指定が可能であり、クラスター内の他の各 IP アドレスにこのデータグラムを送信できることを確認します。ローカル IP アドレスを指定するには PING コマンドを使用し、UDP メッセージを指定するには TRACEROUTE コマンドを使用してください。</p>
—	<p>ポート 5550 と 5551 が他のアプリケーションによって使用されていないことを確認します。これらのポートは IBM クラスタリング用に予約済みです。ポートの使用状況を表示するには、『TCP/IP ネットワーク状況の処理 (WRKTCPSTS)コマンド』を使用してください。ポート 5550 が開かれ、INETD が開始した後に、クラスタリングによって listen 状態になります。</p>

表2. クラスターの管理可能ドメイン・チェックリスト

クラスター・リソース・サービスのクラスター・インターフェースに関する考慮事項	
	<p>IBM System i High Availability Solutions Manager (iHASM ライセンス・プログラム (5761-HAS) をインストールします。高可用性ソリューションにあるすべてのクラスター・ノード上に、有効なライセンス・キーが存在しなければなりません。</p>
—	<p>オプション 41 (i5/OS - HA 切り替え可能リソース) をインストールします。デバイス・ドメインにあるすべてのクラスター・ノード上に、有効なライセンス・キーが存在しなければなりません。</p>
—	<p>すべてのホスト・サーバーが、ホスト・サーバーの開始 (STRHOSTSVR) コマンド: STRHOSTSVR SERVER(*ALL) によって開始されたことを確認します。</p>

クラスターに切り替え可能装置を使用する場合、以下の要件が満たされていなければなりません。

表3. クラスター用の回復装置構成チェックリスト

回復装置要件	
—	<p>IBM System i High Availability Solutions Manager (iHASM ライセンス・プログラム (5761-HAS) をインストールします。高可用性ソリューションにあるすべてのクラスター・ノード上に、有効なライセンス・キーが存在しなければなりません。</p>

表 3. クラスター用の回復装置構成チェックリスト (続き)

回復装置要件	
—	デバイス・ドメインにあるすべてのクラスター・ノードに、オプション 41 (HA 切り替え可能リソース) がインストールされていて、有効なライセンス・キーが存在することを確認します。
—	ディスク管理機能にアクセスするには、DST アクセスおよびユーザー・プロファイルを備えた保守ツール・サーバー (STS) を構成します。詳細については、『ディスク装置の使用可能化およびアクセス』を参照してください。
—	<p>1 つのシステム上の複数の論理区画の間で回復装置を切り替えている場合、論理区画の管理のために HMC 以外のものを使用しているのであれば、区画の仮想 OptiConnect を有効にしてください。これを行うには、専用保守ツール (DST) にサインオンします。詳細については『仮想 OptiConnect』を参照してください。</p> <p>区画の管理にハードウェア管理コンソールを使用している場合には、「OptiConnect」タブで区画プロファイル・プロパティを変更することによって、切り替え可能構成に属する区画ごとに仮想 OptiConnect を有効にしてください。変更内容を実際に反映させるには、区画プロファイルを活動化する必要があります。</p>
—	<p>HSL OptiConnect ループの拡張装置が 2 つのシステム間で切り替えられていて、一方のシステムに論理区画がある場合、その区画に対して HSL OptiConnect を使用可能にします。論理区画の管理に HMC 以外のものを使用している場合には、専用保守ツール (DST) にサインオンして作業を行います。</p> <p>区画の管理にハードウェア管理コンソールを使用している場合には、「OptiConnect」タブで区画プロファイル・プロパティを変更することによって、切り替え可能構成に含まれる区画ごとに HSL OptiConnect を有効にしてください。変更内容を実際に反映させるには、区画プロファイルを活動化する必要があります。</p>
—	<p>複数の論理区画の間で回復装置を切り替えており、さらに HMC 以外のものを使用して論理区画を管理しているのであれば、それらの区画の間でバスを共有できるように構成するか、または入出力プールを構成しなければなりません。ある区画で「共有バスを所有 (own bus shared)」としてバスを構成し、装置切り替えに参加する他のすべての区画では「共有バスを使用 (use bus shared)」として構成する必要があります。</p> <p>ハードウェア管理コンソールを使用して区画の管理を行っている場合には、入出力プロセッサ、入出力アダプター、および接続されているすべてのリソースから成る入出力プールを構成することによって、単一の独立ディスク・プールを複数の区画の間で切り替えられるようにしてください。区画ごとに、その入出力プールにアクセス可能でなければなりません。詳細は、『ハードウェアを切り替え可能にする (Make your hardware switchable)』を参照してください。切り替え可能なデバイスのハードウェア計画の要件に関する詳細は、『切り替えディスクのハードウェア要件』を参照してください。</p>
—	HSL ループにある拡張装置を異なる 2 つのシステム間で切り替えて使用するときには、拡張装置を切り替え可能として構成します。詳細については、『ハードウェアを切り替え可能にする (Make your hardware switchable)』を参照してください。
—	既存の HSL ループに拡張装置を追加するときには、その同じループにあるすべてのサーバーを再始動します。
—	通信バスの最大伝送単位 (MTU) は、調整可能なクラスター通信パラメーター、メッセージ・フラグメント・サイズより大きくなければなりません。クラスター IP アドレスの MTU を確認するには、対象ノードで TCP/IP ネットワーク状況の処理 (WRKTCPS) コマンドを使用してください。MTU は、通信バス全体にわたって、すべてのステップで検査されます。通信バスの MTU の値を増やすよりも、クラスターの作成後にメッセージ・フラグメント・サイズの値を減らすほうが簡単です。メッセージ・フラグメント・サイズの詳細については、『調整可能なクラスター通信パラメーター』を参照してください。クラスター・リソース・サービス情報検索 (QcstRetrieveCRSInfo) API を使用して調整パラメーターの現行設定値を表示し、クラスター・リソース・サービス変更 (QcstChgClusterResourceServices) API を使用して設定値を変更することができます。
—	リモート・ミラーリングの場合は、両方のノードが異なるサイト名に割り当てられていることを確認します。

表4. クラスターのセキュリティ構成チェックリスト

セキュリティ要件	
—	リモート・ノードの開始を試行するには、ターゲット・ノードで「クラスターへの追加可能 (ALWADDCLU)」ネットワーク属性を適切に設定してください。これは、環境に応じて *ANY または *RQSAUT に設定されていなければなりません。この属性が *RQSAUT に設定されている場合、i5/OS オプション 34 (デジタル証明書マネージャー) および CCA 暗号サービス・プロバイダー (オプション 35) をインストールする必要があります。ALWADDCLU ネットワーク属性の設定値に関する詳細については、『ノードをクラスターに追加できるようにする』を参照してください。
—	/QIBM/ProdData/OS400/INETD/inetd.config に指定されている INETD のユーザー・プロファイルの状況を使用可能にします。このユーザー・プロファイルには、*SECADM または *ALLOBJ 特殊権限を付与することはできません。デフォルトでは、INETD のユーザー・プロファイルとして QUSER が指定されます。
—	クラスター・リソース・サービス API を呼び出すユーザー・プロファイルが、すべてのクラスター・ノードに存在し、かつ *IOSYSCFG 権限を持っていることを確認します。
—	クラスター・リソース・グループ (CRG) の出口プログラムを実行させるユーザー・プロファイルが、すべてのリカバリー・ドメイン・ノードに存在していることを確認します。

表5. クラスターのジョブ構成チェックリスト

ジョブ考慮事項	
—	クラスター・リソース・サービス API により、要求を処理するようにジョブを投入できます。このジョブは、クラスター・リソース・グループの作成時に指定された出口プログラムを実行するユーザー・プロファイル、または API を要求したユーザー・プロファイル (回復装置 CRG 内の装置のみをオンに構成変更する場合) のいずれかで実行されます。ユーザー・プロファイルに関連するジョブ待ち行列を保守するサブシステムにおいて、ジョブ待ち行列から実行できるジョブの数が *NOMAX として構成されていることを確認してください。
—	CRG 用に定義されたユーザー・プロファイルから取得されたジョブ記述によって指定されたジョブ待ち行列に、ジョブが投入されます。デフォルトのジョブ記述では、QBATCH ジョブ待ち行列にジョブが送られます。このジョブ待ち行列は複数のユーザーに使用されるため、出口プログラム・ジョブはタイミングよく実行されないかもしれません。固有ユーザー待ち行列を持つ固有ジョブ記述の使用を検討してください。
—	出口プログラム・ジョブが実行されると、ジョブ記述のルーティング・データを使用して、使用する主記憶域プールと実行時属性を選択します。デフォルト値を使うと、実行優先度 50 のバッチ・ジョブを持つジョブがプールで実行されます。これらは、出口プログラム・ジョブの望ましいパフォーマンスをもたらしません。出口プログラム・ジョブを開始するサブシステム (固有ジョブ待ち行列を使用する同じサブシステム) は、同じサブシステムまたは別のサブシステムで開始された他のジョブで使用されることのないプールを出口プログラム・ジョブに割り当てるべきです。さらに、出口プログラム・ジョブに優先度 15 を割り当て、ほぼすべてのユーザー・ジョブよりも先に実行されるようにしてください。
—	QMLTTHDACN システム値を 1 または 2 に設定します。

クラスターの構成と管理に使用できるソフトウェア・インターフェースが、いくつかあります。これらのインターフェースの 1 つが、クラスター・リソース・サービスのインターフェースです。クラスター・リソース・サービスを使用するには、以下の要件を満たす必要があります。

表6. クラスターのクラスター・リソース・サービス構成チェックリスト

クラスター・リソース・サービスのグラフィカル・インターフェースに関する考慮事項	
—	IBM System i High Availability Solutions Manager (iHASM ライセンス・プログラム (5761-HAS) をインストールします。高可用性ソリューションにあるすべてのクラスター・ノード上に、有効なライセンス・キーが存在しなければなりません。

表 6. クラスターのクラスター・リソース・サービス構成チェックリスト (続き)

クラスター・リソース・サービスのグラフィカル・インターフェースに関する考慮事項	
—	オプション 41 (i5/OS - HA 切り替え可能リソース) をインストールします。デバイス・ドメイン nad iHASM LP.. にあるすべてのクラスター・ノード上に、有効なライセンス・キーが存在しなければなりません。
—	すべてのホスト・サーバーが、ホスト・サーバーの開始 (STRHOSTSVR) コマンド: STRHOSTSVR SERVER(*ALL) によって開始されたことを確認します。

FlashCopy の計画

IBM Systems Storage 外部記憶装置を使用する i5/OS 高可用性環境で、バックアップ・ウィンドウを縮小する手段として FlashCopy を使用することができます。FlashCopy を使用する前に、最小必要要件が満たされていることを確認する必要があります。

FlashCopy のハードウェア要件

i5/OS 高可用性ソリューションの FlashCopy を使用するには、最小ハードウェア要件が満たされていることを確認してください。

FlashCopy を使用するための最小ハードウェア要件は、以下のとおりです。

- 各システムに接続された外部ストレージ・ユニットである IBM System Storage DS6000 または DS8000 の少なくとも 1 つに地理的に分割された、少なくとも 2 つの System i モデルまたは論理区画。外部ストレージ・ユニットの DS6000 および DS8000 が、外部ストレージ用のファイバー・チャンネル接続をサポートする、すべての System i モデルでサポートされていること。
- サポートされるファイバー・チャンネルのアダプターのうち、以下に挙げたもののいずれかが必要です。
 - 2766 2 ギガビット・ファイバー・チャンネル・ディスク・コントローラー PCI
 - 2787 2 ギガビット・ファイバー・チャンネル・ディスク・コントローラー PCI-X
 - 5760 4 ギガビット・ファイバー・ディスク・コントローラー PCI-X
- システム・ストレージの適切なディスク・サイジングを、構成前に行っておく必要があります。ソース用のディスク・セット、同様にターゲット用のディスク装置セット、さらに各整合性コピー用に別のセットが必要になります。

FlashCopy のソフトウェア要件

i5/OS 高可用性ソリューションで FlashCopy を使用するには、最小ソフトウェア要件が満たされている必要があります。

FlashCopy の最小ソフトウェア要件は、以下のとおりです。

- 高可用性ソリューション内の各 System i モデルで i5/OS V6R1 を稼働させて、IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム (5761-HAS) で使用できるようにしておく必要があります。

注: 以前のリリース用に、Lab Services のオフアリングである IBM Copy Services for System i を使用して、IBM System Storage ソリューションを処理することもできます。複数のプラットフォーム上でグローバル・ミラーを使用している場合、または複数の System i 区画上にグローバル・ミラーをインプリメントする場合は、IBM Copy Services for System i も使用できます。

- IBM System i High Availability Solutions Manager (iHASM) が各システムにインストールされていること。
- 最新の PTF がインストールされていることを確認してください。

FlashCopy の通信要件

最初の段落と要約で使用する簡略説明を、ここに記載します。

FlashCopy を使用するための最小通信要件は、以下のとおりです。

- IBM System Storage の外部ストレージ装置 DS6000 または DS8000 の少なくとも 1 つ (各システムに接続されているもの) によって地理的に分離された、少なくとも 2 つの System i モデル。外部ストレージ・ユニットの DS6000 および DS8000 が、外部ストレージ用のファイバー・チャンネル接続をサポートする、すべての System i モデルでサポートされていること。
- サポートされるファイバー・チャンネルのアダプターのうち、以下に挙げたもののいずれかが必要です。
 - 2766 2 ギガビット・ファイバー・チャンネル・ディスク・コントローラー PCI
 - 2787 2 ギガビット・ファイバー・チャンネル・ディスク・コントローラー PCI-X
 - 5760 4 ギガビット・ファイバー・ディスク・コントローラー PCI-X
- DS6000 または DS8000 で外部ロード・ソースをサポートするために、新規 IOP が必要になります。
 - SAN ロード・ソースの 2847 PCI-X IOP 機能
- システム・ストレージの適切なディスク・サイジングを、構成前に行っておく必要があります。ソース用のディスク・セット、同様にターゲット用のディスク装置セット、さらに各整合性コピー用に別のセットが必要になります。

高可用性のためのセキュリティー計画

高可用性ソリューションを構成する前に、ご使用の環境で現在のセキュリティー戦略を再評価し、高可用性を容易にするための適切な変更を行う必要があります。

クラスター全体の情報の配布

クラスター全体の情報の使用と管理に関するセキュリティー上の意義の考察

情報配布 (QcstDistributeInformation) API を使用すれば、クラスター・リソース・グループのリカバリー・ドメイン内の 1 つのノードから、そのリカバリー・ドメイン内の他のノードにメッセージを送信することができます。これは、出口プログラムの処理で役に立つことがあります。ただし、その情報は暗号化されていないので、注意が必要です。セキュリティーが確保されているネットワークを使用している場合を除き、このメカニズムでセキュアな情報を送信することは避けてください。

クラスター・ハッシュ・テーブル API を使用して、非永続的データをクラスター・ノード間で共有したり複製したりすることができます。このデータは、非永続記憶装置に保管されます。これは、そのクラスター・ノードがクラスタリングされたハッシュ・テーブルの一部になっている間しかデータを取り出せないことを意味します。これらの API は、クラスタリングされたハッシュ・テーブル・ドメインに定義されているクラスター・ノードからでなければ使用できません。クラスター・ノードは、クラスター内でアクティブになっていなければなりません。

クラスター・メッセージ機能を使用して配布される情報も、同様にセキュリティー保護が十分ではありません。これには、低レベルのクラスター・メッセージ機能も含まれます。出口プログラムのデータに変更が加えられた場合、そのデータを含むメッセージの暗号化は行われません。

ファイアウォールを備えたクラスターの使用に関する考慮事項

ファイアウォールを使用しているネットワーク内でクラスタリングを使用する場合、いくつかの制限事項と要件に配慮する必要があります。

ファイアウォールを備えたクラスタリングを使用する場合、他のクラスター・ノードとのやりとりのために、アウトバウンド・メッセージの送信とインバウンド・メッセージの受信を行う機能を、各ノードに提供する必要があります。各ノード上の各クラスター・アドレスが、他のどのノードのどのクラスター・アドレスとも通信できるように、ファイアウォールに開口部を設けなければなりません。ネットワーク上を行き来する IP パケットは、多様なトラフィック・タイプをとることがあります。タイプが ICMP のクラスタリングは、ping を使用しますが、UDP と TCP も使用します。ファイアウォールの構成時に、このタイプをベースにトラフィックをフィルタリングすることができます。クラスタリングの稼働のためには、ICMP、UDP、および TCP のトラフィックをファイアウォールで許可する必要があります。アウトバウンド・トラフィックは任意のポートを通して送信できるのに対して、インバウンド・トラフィックはポート 5550 および 5551 で受信されます。

すべてのノード上のユーザー・プロファイルの保守

クラスター内のすべてのノードにあるユーザー・プロファイルを保守するために、2 種類のメカニズムを使用することができます。

- 1 高可用性環境では、プロファイル名が同一のユーザー・プロファイルは、システム全体で同一と見なされます。この名前は、クラスターにおける固有 ID です。ただし、ユーザー・プロファイルには、ユーザー識別番号 (UID) およびグループ識別番号 (GID) も付けられています。独立ディスク・プールがあるシステム上で使用不可になり、別のシステム上で使用可能になる際の切り替え時に発生する、内部処理の量を削減するには、UID 値および GID 値をデバイス CRG のリカバリー・ドメイン全体で同期化する必要があります。高可用性環境全体でユーザー・プロファイルを同期化する方式は、2 種類あります。

一方のメカニズムは、クラスター内のすべてのノードを通して共用リソースをモニターするためのクラスター管理可能ドメインを作成するためのメカニズムです。クラスター管理可能ドメインは、ユーザー・プロファイルに加えていくつかのタイプのリソースをモニターできるので、すべてのノードを通して共用されているリソースの管理を簡単に行うことができます。クラスター管理可能ドメインがアクティブになっているときに、ユーザー・プロファイルを更新すると、変更内容は自動的に他のノードに伝搬されます。クラスター管理可能ドメインがアクティブになっていない場合は、クラスター管理可能ドメインがアクティブになってから、変更内容が伝搬されます。この方式は、高可用性環境でユーザー・プロファイルを自動的に保守しますので、使用することをお勧めします。

もう一方のメカニズムでは、管理者は System i ナビゲーターの「マネージメント・セントラル」も使用できるため、複数のシステムおよびシステム・グループを対象に各種機能を実行することができます。このサポートには、オペレーターがクラスター内の複数のシステムにまたがって実行しなければならない、いくつかの一般的なユーザー管理タスクも含まれています。マネージメント・セントラルを使用すると、システムのグループに対してユーザー・プロファイル機能を実行することができます。管理者であれば、ユーザー・プロファイルの作成時に、ターゲット・システムで伝搬後のコマンドが実行されるように設定することもできます。

重要:

- クラスター内でパスワード同期を活用するユーザー・プロファイルを共有する予定の場合、サーバー・セキュリティ保持 (QRETSVRSEC) システム値を 1 に設定する必要があります。
- ユーザー・プロファイルのモニター対象リソース項目 (MRE) を追加した後に QRETSVRSEC を 0 に変更して、その後にパスワードを変更すると (パスワードがモニターされている場合)、MRE のグローバル状況が「不整合」に設定されます。MRE は使用不可としてマークされます。この変更後にユーザー・プロファイルに加えられた変更内容は、同期化されません。この問題を解決するには、QRETSVRSEC を 1 に変更して、MRE を除去してから、MRE を再度追加してください。

関連タスク

110 ページの『クラスター管理可能ドメインの作成』

- 高可用性ソリューションにおけるクラスター管理可能ドメインは、クラスター内のシステムおよび区画間における、リソースの同期化を保持するメカニズムを提供します。

高可用性の構成

ご使用の i5/OS 環境で高可用性ソリューションを構成する前に、適切な計画を作成済みであること、および高可用性と災害時回復時に必要とされるリソースと目的について理解していることを確認してください。高可用性および高可用性テクノロジーに関連したタスクの構成シナリオを使用することにより、独自の高可用性ソリューションを作成します。

シナリオ: 高可用性の構成

- ユーザーのニーズと回復要件に基づく高可用性ソリューションをインプリメントするのに役立つ、さまざまな i5/OS 高可用性環境と段階的な構成タスクの例を、構成シナリオによって提供します。

- これらのシナリオには、高可用性のビジネス目標に関する説明が含まれており、高可用性ソリューション内のリソースを示す図が提示されています。ソリューションのそれぞれの例には、高可用性のセットアップおよびテストを行うための、段階的な指示が記載されています。ただし、この情報はすべての構成例を説明するものではなく、高可用性を検査するために、追加のテストが必要となる場合があります。

シナリオ: 論理区画間の切り替えディスク

- このシナリオでは、単一システムに常駐する 2 つの論理区画間で切り替えられるディスク・プールを使用する i5/OS 高可用性ソリューションについて説明しています。

概説

- 論理区画化を使用すると、1 つの i5/OS システム機能を複数の独立したシステムのように稼働させることができます。このソリューションは、すでに論理区画が業務環境内で構成済みの場合は、良い選択と言えます。

- このシナリオでは、論理区画の構成については説明しません。

目標

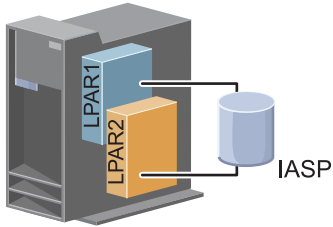
- このソリューションには以下の利点があります。
 - 使用可能なシステム・リソースを使用するため低コストであること。
 - 計画停止中の業務リソースに対する可用性の提供。
 - いくつかの計画外の停止 (単一の論理区画の障害など) 時の業務リソースに対する可用性の提供。
 - データの単一コピーの使用による、必要なディスク装置数の最小化。
 - 同期が不要なカレント・データがソリューションに含まれること。

- このソリューションには以下の制限事項があります。

- サイト全体が停止した場合の災害時回復機能はありません。
- 論理区画に構成には、要件があります。
- 区画間の冗長ハードウェアには、要件が必要とされる場合があります。
- 独立ディスク・プールにあるデータの論理コピーは 1 つのみです。つまり、データは RAID で保護可能ですが、SPOF (Single Point of Failure) になる可能性があります。
- 両方の論理区画からディスク・プールへの同時アクセス機能はありません。

詳細

次の図は、このシナリオの環境を表しています。



構成ステップ

このシナリオに関連した高可用性テクノロジーを構成するには、以下のタスクを実行します。

1. クラスターのチェックリストの完了
2. クラスターの作成
3. ノードの追加
4. ノードの開始
5. デバイス・ドメインへのノードの追加
6. クラスター管理可能ドメインの作成
7. クラスター管理可能ドメインの開始
8. 独立ディスク・プールの作成
9. モニター対象リソース項目の追加
10. ハードウェアを切り替え可能にする
11. 装置 CRG の作成
12. 装置 CRG の開始
13. ディスク・プールを使用可能にする
14. 切り替えを実行して高可用性ソリューションをテスト

シナリオ: システム間の切り替えディスク

このシナリオで紹介する i5/OS 高可用性ソリューションは、2 つのシステム間で切り替えディスクを使用して、データ、アプリケーション、または装置に対して、計画停止および計画外の停止時に高可用性を提供するものです。

概説

切り替えディスク・テクノロジーを使用することにより、このソリューションはシンプルな高可用性ソリューションを提供します。このソリューションでは、切り替えディスクに保管されているデータのコピーは常にカレントに保たれるため、システム間でのデータの同期は必要なく、伝送時におけるデータ紛失のリスクを除去できます。

目標

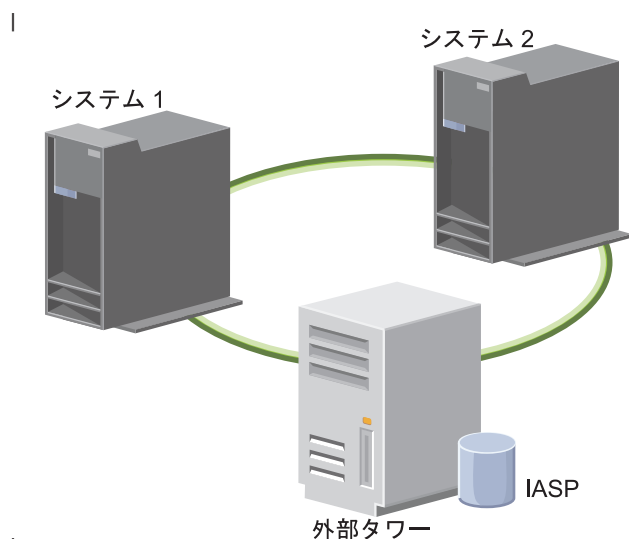
このソリューションには以下の利点があります。

- 計画停止中の業務リソースに対する可用性を提供します。
- 一部の計画外停止中の業務リソースに対する可用性を提供します。

- | • データの単一コピーによる必要なディスク装置数の最小化。
- | • パフォーマンス・オーバーヘッドの最小化。
- | • データが現行のままで保持され、同期化の必要がないようにします。
- | このソリューションには以下の制限事項があります。
- | • サイト全体が停止した場合の災害時回復機能はありません。
- | • 独立ディスク・プール上のデータの論理コピーは 1 つのみです。つまり、データは RAID で保護可能ですが、SPOF (Single Point of Failure) になる可能性があります。
- | • 両方のシステムからのディスク・プールへの同時アクセス機能はありません。

| 詳細

| 次の図は、このシナリオの環境を表しています。



| 構成ステップ

- | 1. 計画チェックリストの完了
- | 2. クラスターの作成
- | 3. ノードの追加
- | 4. ノードの開始
- | 5. デバイス・ドメインへのノードの追加
- | 6. クラスター管理可能ドメインの作成
- | 7. クラスター管理可能ドメインの開始
- | 8. 独立ディスク・プールの作成
- | 9. モニター対象リソース項目の追加
- | 10. ハードウェアを切り替え可能にする
- | 11. 装置 CRG の作成
- | 12. 装置 CRG の開始
- | 13. ディスク・プールを使用可能にする
- | 14. 切り替えを実行して高可用性ソリューションをテスト

シナリオ: リモート・ミラーリングによる切り替えディスク

このシナリオでは、3つのノード・クラスターにおいてリモート・ミラーリングによる切り替えディスクを使用する、i5/OS 高可用性ソリューションについて説明します。このソリューションは、災害時回復と高可用性の両方を提供します。

概説

実動場所 (Uptown) で、切り替えディスクは2つのノード間で独立ディスク・プールを移動するために使われます。また、このソリューションはリモート・ミラーリングも使用して、2番目の場所 (Downtown) で独立ディスクのコピーを生成します。こうすることにより、このソリューションは災害時回復と高可用性の両方を提供します。このソリューションの利点は、根本的には基本的な切り替えディスク・ソリューションと同じです。ただし、別のロケーションでアプリケーション・データを複製することにより、アプリケーション・データに災害時回復を提供できる付加メリットがあります。実動場所 (Uptown) には、論理区画間で切り替えることができる独立ディスク・プールがあり、計画停止 (フィックス適用時など) において、高速切り替え時間による高可用性を提供します。このソリューションはまた、サイト間ミラーリングおよびリモート・ミラーリングによる災害時回復も提供します。

リモート・ミラーリングは、サイト間ミラーリングの副次機能で、データはリモート・ロケーションにある独立ディスク・プールのコピーにミラーリングされます。実動場所 (Uptown) にある独立ディスク・プールのデータは、バックアップ・サイト (Downtown) の独立ディスク・プールにミラーリングされます。このソリューションは、外部ストレージを基にした他のソリューション (IBM System Storage のグローバル・ミラーリングやメトロ・ミラーリングなど) よりもシンプルで費用のかからない代替ソリューションを提供します。ただし、リモート・ミラーリングは、外部ストレージ・ソリューションが提供するすべてのパフォーマンス・オプションを提供するものではありません。

目標

このソリューションには以下の利点があります。

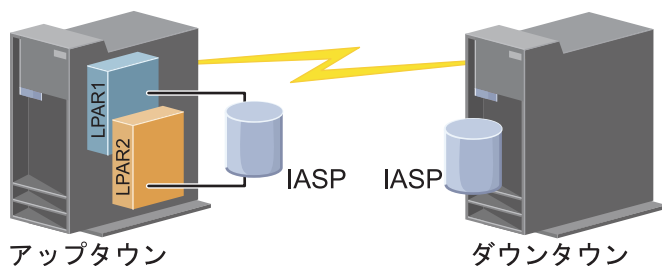
- 計画停止中の業務リソースに対する可用性を提供します。
- 一部の計画外停止中の業務リソースに対する可用性を提供します。
- 全サイトにわたる災害時に、業務リソースに対する可用性を提供します。
- 各サイトが、必要なディスク装置数を最小に抑える、データの単一コピーを持てるようにします。
- データが現行のままで保持され、同期化の必要がないようにします。

このソリューションには以下の制限事項があります。

- ディスク・プールに同時アクセスできません。ただし、データの2次コピーをオフライン処理するために、ミラー・コピーを切り離すことができます。
- リモート・ミラーリングの対応に必要な中央演算処理装置 (CPU) の増加による、パフォーマンスへの影響の可能性があります。
- 冗長通信パスと十分な帯域幅の使用を検討してください。

詳細

次の図は、このソリューションを表しています。



構成ステップ

1. クラスターの計画チェックリストの完了
2. クラスターの作成
3. ノードの追加
4. ノードの開始
5. デバイス・ドメインへのノードの追加
6. 装置 CRG の作成
7. サイト名の定義
8. クラスター管理可能ドメインの作成
9. クラスター管理可能ドメインの開始
10. 独立ディスク・プールの作成
11. モニター対象リソース項目の追加
12. ハードウェアを切り替え可能にする
13. リモート・ミラーリングの構成
14. ディスク・プールを使用可能にする
15. 切り替えを実行して構成をテスト

関連タスク

- 120 ページの『リモート・ミラーリングの構成』
リモート・ミラーリングは、サイト間ミラーリングの副次機能です。リモート・ミラーリングを使用して高可用性ソリューションを構成するには、実動システムとバックアップ・システムの間で、ミラーリング・セッションを構成する必要があります。

シナリオ: リモート・ミラーリングを使用するサイト間ミラーリング

このシナリオでは、2 つのノード・クラスターでリモート・ミラーリングを使用する i5/OS 高可用性ソリューションについて説明します。このソリューションは、災害時回復と高可用性の両方を提供します。

概説

リモート・ミラーリングは、サイト間ミラーリングの副次機能で、データはリモート・ロケーションにある独立ディスク・プールのコピーにミラーリングされます。このソリューションは、実動システム (システム 1) 上でサイト全体の障害が発生した場合に、災害時回復を行います。このソリューションでは、バックアップ・サイト (システム 2) へのフェイルオーバーが発生し、ミラーリングされたデータの複製操作が継続されます。このソリューションは、外部ストレージを基にした他のソリューション (IBM System Storage

のグローバル・ミラーリングやメトロ・ミラーリングなど) よりもシンプルで費用のかからない代替ソリューションを提供します。ただし、リモート・ミラーリングは、外部ストレージ・ソリューションが提供するすべてのパフォーマンス・オプションを提供するものではありません。

目標

このソリューションには以下の利点があります。

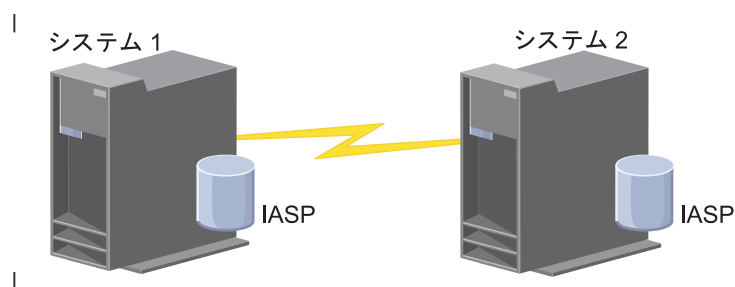
- 計画停止中の業務リソースに対する可用性を提供します。
- 一部の計画外停止中の業務リソースに対する可用性を提供します。
- 災害時に、業務リソースに対する可用性を提供します。
- データが現行のまま保持され、同期化の必要がないようにします。

このソリューションには以下の制限事項があります。

- ディスク・プールに同時アクセスできません。ただし、データの 2 次コピーをオフライン処理するために、ミラー・コピーを切り離すことができます。
- リモート・ミラーリングのサポートにより多くの中央演算処理装置 (CPU) が必要なため、パフォーマンスに影響を与えます。
- 冗長通信パスと十分な帯域幅の使用を検討してください。

詳細

次の図は、このソリューションを示しています。



構成ステップ

1. クラスターの計画チェックリストの完了
2. クラスターの作成
3. ノードの追加
4. ノードの開始
5. デバイス・ドメインへのノードの追加
6. クラスター管理可能ドメインの作成
7. クラスター管理可能ドメインの開始
8. 独立ディスク・プールの作成
9. モニター対象リソース項目の追加
10. ハードウェアを切り替え可能にする
11. 装置 CRG の作成
12. 装置 CRG の開始
13. ディスク・プールを使用可能にする

- | 14. リモート・ミラーリングの構成
- | 15. 切り替えを実行して構成をテスト

| シナリオ: メトロ・ミラーリングを使用するサイト間ミラーリング

| このシナリオでは、外部ストレージを基礎として、比較的近い場所に分置されているストレージ・システム
| で災害時回復と高可用性を可能にする、i5/OS 高可用性ソリューションについて説明します。メトロ・ミラ
| ーとは、IBM System Storage ソリューションの一種で、実動場所にあるストレージ・ユニットからバック
| アップ場所にあるストレージ・ユニットに、同期的にデータをコピーするものです。この方法では、データ
| はバックアップ場所において、整合された状態を保ちます。

| 概説

| メトロ・ミラー・ソリューションを使用したサイト間ミラーリングは、大都市圏内にある外部ストレージ・
| ユニットを使用することによって、高可用性と災害時回復を可能にします。独立ディスク・プールは、外部
| ストレージ装置間で複製され、計画停止と計画外の停止の、両方の場合で使用可能となります。実動ボリュ
| ムに対するホスト更新をメトロ・ミラーが受け取った場合、バックアップ・ボリュームに対して対応する
| 更新を行います。メトロ・ミラーでサポートされる距離は、最大 300 キロメートル (186 マイル) です。
| メトロ・ミラーの応答時間の遅延は、ボリューム間の距離に比例します。

| このシナリオでは、ネイティブの i5/OS 高可用性テクノロジーの構成について説明します。IBM System
| Storage DS6000 または DS8000 シリーズに関するインストール手順や構成手順は扱っていません。ここで
| は、IBM System Storage ソリューションが i5/OS 高可用性構成に先立って実施されていることが前提とな
| ります。DS6000 のインストールおよび構成について詳しくは、IBM System Storage DS6000 Information
| Center を参照してください。DS8000 のインストールおよび構成について詳しくは、IBM System Storage
| DS8000 Information Center を参照してください。

| 目標

| このソリューションには以下の利点があります。

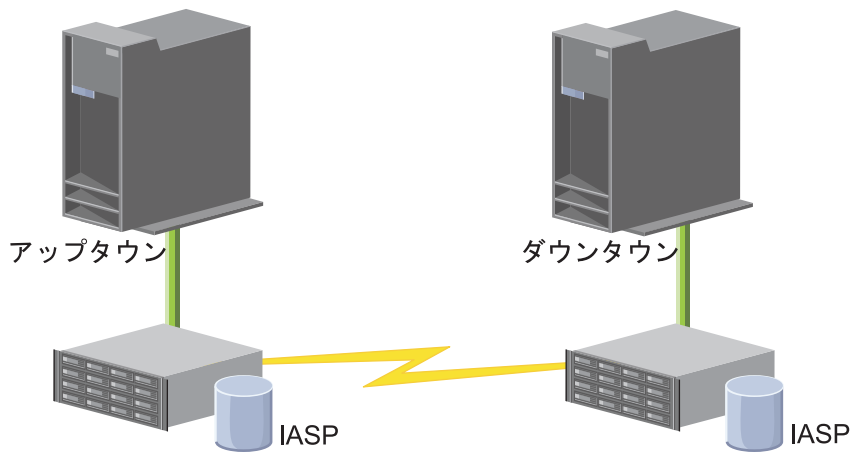
- | • 外部ストレージ・ユニットによって複製が完全に管理されるため、System i の CPU は使用されませ
| ん。システム・レベルの障害がシステムで発生した場合でも、複製はストレージ・ユニットで続行され
| ます。
- | • 計画停止または計画外の停止時 (保守上の停止、ソフトウェア/PTF 関連の停止など) および災害時回復
| 時でも、業務リソースを使用できます。
- | • 入出力は整合性は保持されているため、同期化する必要はありません。
- | • ジャーナル処理を使用すると、リカバリー時間が速くなります。計画外の停止またはフェイルオーバー
| の際に、ジャーナル処理によってデータ回復を素早く行うことができます。ジャーナル処理では、ミラ
| ーリングが発生しているディスクに対して、強制的にデータ変更を行います。ジャーナル処理を使用し
| ない場合、メモリー内のデータが失われる場合があります。ジャーナル処理によって、これらのデー
| タ・レベル・トランザクションのリカバリーが可能になり、リカバリー時間にも貢献します。
- | • メトロ・ミラーのソース側でもターゲット側でも、FlashCopy 機能を使用することができます。

| このソリューションでは、以下の制限事項があります。

- | • 外部ストレージ・ハードウェアが必要
- | • 冗長通信パスと十分な帯域幅の使用を検討する必要がある
- | • ディスク・プールに同時アクセスできない

詳細

次の図は、このソリューションを表しています。



構成ステップ

1. クラスターの計画チェックリストの完了
2. クラスターの作成
3. ノードの追加
4. ノードの開始
5. デバイス・ドメインへのノードの追加
6. クラスター管理可能ドメインの作成
7. クラスター管理可能ドメインの開始
8. 独立ディスク・プールの作成
9. モニター対象リソース項目の追加
10. ハードウェアを切り替え可能にする
11. 装置 CRG の作成
12. 装置 CRG の開始
13. ディスク・プールを使用可能にする
14. メトロ・ミラー・セッションの構成
15. 切り替えを実行して構成をテスト

シナリオ: グローバル・ミラーを使用するサイト間ミラーリング

このシナリオでは、外部ストレージを基礎として、非常に離れた場所に分置されているストレージ・システムで災害時回復と高可用性を可能にする、i5/OS 高可用性ソリューションについて説明します。グローバル・ミラーとは、IBM Systems Storage ソリューションの一種で、実動場所に設置されたストレージ・ユニットからバックアップ場所に設置されたストレージ・ユニットに、非同期的にデータをコピーするものです。この方法では、データはバックアップ場所において、整合された状態を保ちます。

概説

グローバル・ミラー・ソリューションを使用したサイト間ミラーリングでは、離れた場所に設置された複数の外部ストレージ・ユニットを使用することによって、災害時回復ソリューションを可能にします。独立ディスク・プールは、外部ストレージ装置間で複製され、計画停止と計画外の停止の、両方の場合で使用可能となります。

このシナリオでは、ネイティブの i5/OS 高可用性テクノロジーの構成について説明します。IBM System Storage DS6000 または DS8000 シリーズに関するインストール手順や構成手順は扱っていません。ここでは、IBM System Storage ソリューションが i5/OS 高可用性構成に先立って実施されていることが前提となります。DS6000 のインストールおよび構成について詳しくは、IBM System Storage DS6000 Information Center を参照してください。DS8000 のインストールおよび構成について詳しくは、IBM System Storage DS8000 Information Center を参照してください。

目標

このソリューションには以下の利点があります。

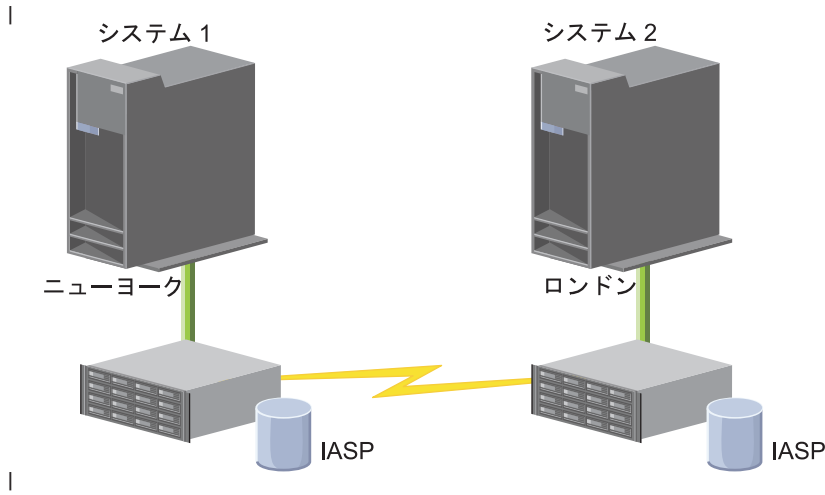
- 外部ストレージ・ユニットによって複製が完全に管理されるため、System i の CPU は使用されません。システム・レベルの障害がシステムで発生した場合でも、複製はストレージ・ユニットで続行されます。
- 計画停止または計画外の停止時（保守上の停止、ソフトウェア/PTF 関連の停止など）および災害時回復時でも、業務リソースを使用できます。
- ジャーナル処理を使用すると、リカバリー時間が速くなります。計画外の停止またはフェイルオーバーの際に、ジャーナル処理によってデータ回復を素早く行うことができます。ジャーナル処理では、ミラーリングが発生しているディスクに対して、強制的にデータ変更を行います。ジャーナル処理を使用しない場合、メモリー内のデータが失われる場合があります。ジャーナル処理によって、これらのデータ・レベル・トランザクションのリカバリーが可能になり、リカバリー時間にも貢献します。
- グローバル・ミラーのソース側でもターゲット側でも、FlashCopy 機能を使用することができます。

このソリューションでは、以下の制限事項があります。

- 外部ストレージ・ハードウェアが必要
- 冗長通信パスと十分な帯域幅の使用を検討する必要がある
- ディスク・プールに同時アクセスできない
- 所定の System Storage サーバーでグローバル・ミラーを構成することができる System i 区画は 1 つだけです。他の System i 区画や、他のプラットフォームのサーバーは、同時にグローバル・ミラーを使用することはできません。グローバル・ミラー・セッションに複数のユーザーを追加すると、予想不能の結果が生じます。

詳細

次の図は、このソリューションを表しています。



構成ステップ

1. クラスターの計画チェックリストの完了
2. クラスターの作成
3. ノードの追加
4. ノードの開始
5. デバイス・ドメインへのノードの追加
6. クラスター管理可能ドメインの作成
7. クラスター管理可能ドメインの開始
8. 独立ディスク・プールの作成
9. モニター対象リソース項目の追加
10. ハードウェアを切り替え可能にする
11. 装置 CRG の作成
12. 装置 CRG の開始
13. ディスク・プールを使用可能にする
14. グローバル・ミラー・セッションの構成
15. 切り替えを実行して構成をテスト

高可用性のための TCP/IP のセットアップ

クラスター・リソース・サービスは、高可用性環境内のシステムまたは論理区画である他のクラスター・ノードとの通信を確立するのに IP だけを使用するので、すべてのクラスター・ノードが IP に到達できなければなりません。すなわち、クラスターでノードを接続するためには IP インターフェースが構成されている必要があります。

IP アドレスは、ネットワーク管理者が各クラスター・ノード上の TCP/IP ルーティング・テーブルにおいて手動で設定するか、ネットワーク内のルーター上で実行されているルーティング・プロトコルによって生成します。この TCP/IP ルーティング・テーブルは、各ノードを検出するためにクラスタリングが使用するマップです。したがって、各ノードには固有な IP アドレスが必要です。

1 つのノードには最多で 2 つの IP アドレスを割り当てることができます。これらのアドレスは、いかなる手段によっても、他のネットワーク通信アプリケーションが変更してはなりません。各アドレスを割り当てる際には、どのアドレスがどのタイプの通信回線を使用するかに注意してください。特定のタイプの通信メディアを優先して使用したい場合には、優先して使用したいそのメディアによって、1 番目の IP アドレスを構成してください。1 番目の IP アドレスは、信頼メッセージ機能およびハートビート・モニターによって、優先的に扱われる IP アドレスです。ノード上のすべての IP アドレスは、クラスター内の他のすべての IP アドレスに到達可能でなければなりません。ping をして、UDP メッセージの経路追跡を両方向で使用できれば、アドレスからアドレスへの到達が可能であるということです。

注: クラスタリングには、ループバック・アドレス (127.0.0.1) が確実にアクティブであることが必要です。デフォルトでは、このアドレス (メッセージをローカル・ノードに戻すために使用される) はアクティブであるのが普通です。しかし、何らかの間違いによって終了した場合には、このアドレスを再びアクティブにしない限り、クラスターのメッセージ機能は実行できません。

TCP/IP 構成属性の設定

クラスター・リソース・サービスを使用可能にするには、ネットワークの TCP/IP 構成でいくつかの属性を設定しなければなりません。

これらの属性を設定しないと、クラスターにノードを追加することはできません。

- 他のネットワークと通信するためのルーターとして System i 製品を使用する予定の場合に、そのサーバー上で他のルーティング・プログラムが実行されていなければ、CHGTCPA (TCP/IP 属性の変更) コマンドを使用して IP データグラムの転送を *YES に設定します。
- INETD サーバーを START に設定します。INETD サーバーの開始方法については、『INETD サーバーの開始』を参照してください。
- CHGTCPA (TCP/IP 属性の変更) コマンドを使用して UDP チェックサムを *YES に設定します。
- ブリッジを使用してトークンリング・ネットワークに接続する場合は、MCAST の転送を *YES に設定します。
- i5/OS に対して OptiConnect を使用してクラスター・ノード間の通信を行う場合は、STRSBS(QSOC/QSOC) を指定して QSOC サブシステムを開始します。

INETD サーバーの開始

ノードを追加または開始したり、区画のマージ処理を行うためには、インターネット・デーモン (INETD) サーバーを開始しなければなりません。

INETD サーバーをクラスター内で常時実行することをお勧めします。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS で「ネットワーク」を選択します。
4. 「ようこそ」ページで、「TCP/IP サーバー」を選択します。
5. 「TCP/IP サーバー (TCP/IP Server)」ページで、「TCP/IP サーバー」を選択します。使用可能な TCP/IP サーバーのリストが表示されます。
6. リストから「INETD」を選択します。
7. 「アクションの選択」メニューから、「開始」を選択します。サーバーの状況が「開始済み」に変更されます。

INETD サーバーを開始するには、TCP/IP サーバーの開始 (STRTCPSVR) コマンドを使用して、*INETD パラメーターを指定してもかまいません。INETD サーバーが開始されると、対象ノードのアクティブなジョブのリストに、ユーザー QTCP (QTOGINTD) ジョブが表示されます。

関連資料

TCP/IPサーバーの開始 (STRTCPSVR)

クラスタの構成

i5/OS で高可用性をインプリメントするには、回復力のあるリソースの制御および管理を行うように構成されたクラスタが必要です。切り替えディスク、サイト間ミラーリング、論理複製などのデータ回復テクノロジーとあわせてクラスタ・テクノロジーを使用すると、高可用性ソリューションには欠かせない重要なインフラストラクチャーが提供されます。

クラスタ・リソース・サービスにより、クラスタ・トポロジーの保守、ハートビート・モニタリングの実行、およびクラスタ構成とクラスタ・リソース・グループの作成や管理を可能にする、統合サービスが提供されます。またクラスタ・リソース・サービスは、クラスタ内の各ノードのトラックを保持する信頼メッセージ機能を提供し、全ノードがクラスタ・リソースに関する整合性の取れた情報を有するようにします。IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム番号 (5761-HAS) の一部である、クラスタ・リソース・サービスのグラフィカル・ユーザー・インターフェースを使用することにより、高可用性ソリューション内でクラスタの構成および管理を行うことができます。さらに、クラスタ構成の処理を可能にする制御言語 (CL) コマンドのセットを、ライセンス・プログラムによって提供することもできます。

また、アプリケーション・プロバイダーまたはお客様が、アプリケーションの可用性を拡張するために使用できる、アプリケーション・プログラム・インターフェース (API) および諸機能も備えています。

これらの IBM テクノロジーに加えて、論理複製テクノロジーを備えたクラスタを使用するアプリケーションが、高可用性ビジネス・パートナー様から提供されます。

クラスタの作成

クラスタを作成するには、クラスタに少なくとも 1 つのノードを組み込むとともに、そのクラスタ内に配置されることになるノードのうち、少なくとも 1 つのノードに対するアクセス権を有していなければなりません。

指定されるノードが 1 つだけの場合、そのノードは現在アクセスしているシステムである必要があります。クラスタの作成要件の完全なリストについては、74 ページの『クラスタの計画チェックリスト』を参照してください。

- | クラスタ内で切り替え可能な装置を使用する場合、またはサイト間ミラーリング・テクノロジーを使用し
- | て高可用性ソリューションを構成する場合には、追加の要件が発生します。これらのテクノロジーを使用す
- | る高可用性ソリューションの構成例については、『シナリオ: 高可用性ソリューションの構成』を参照して
- | ください。段階的な構成タスク、およびこのソリューションが提供する障害補償範囲の概要を、各シナリオ
- | で説明します。これらの例を使用して、高可用性ソリューションの構成や、ユーザーのニーズに合わせたカ
- | スタマイズを行うことができます。

クラスタを作成するには、以下の手順を実行します。

- | 1. Web ブラウザーで、http://mysystem:2001 と入力します。mysystem はシステムのホスト名です。
- | 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。

- 1 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
- 1 4. 「ようこそ」ページで、「**新規クラスター**」を選択します。
- 1 5. 「新規クラスター」ウィザードの指示に従い、クラスターを作成します。

新規クラスターの作成後、「ようこそ」ページが変更されて、ページ上部にクラスターの名前が表示されません。クラスター処理用のタスクが、「ようこそ」ページにリストされます。

新しいノードを追加するために必要なクラスターを作成してから、CRG を作成します。

関連情報

クラスターの作成 (CRTCLU) コマンド

クラスターの作成 (QcstCreateCluster) API (英語)

ノードをクラスターに追加できるようにする:

ノードをクラスターに追加するためには、クラスターへの追加可能 (ALWADDCLU) ネットワーク属性の値を設定する必要があります。

クラスター・ノードとして設定するサーバーに対して、ネットワーク属性変更 (CHGNETA) コマンドを使用します。CHGNETA コマンドは、システムのネットワーク属性を変更します。ALWADDCLU ネットワーク属性によって、クラスターのノードとしてノードを追加することが、他のシステムで可能かどうかを指定します。

注: ネットワーク属性 ALWADDCLU を変更するには、*IOSYSCFG 権限がなければなりません。

使用可能な値は、次のとおりです。

***SAME**

値は変更されません。システム出荷時の設定値は *NONE です。

***NONE**

他のシステムは、このシステムをクラスターのノードとして追加できません。

***ANY** 他のどんなシステムも、このシステムをクラスターのノードとして追加できます。

***RQSAUT**

クラスター追加要求が認証された後ならば、他のどんなシステムもこのシステムをクラスターのノードとして追加できます。

ALWADDCLU ネットワーク属性を検査すれば、追加しようとしているノードをクラスターの一部とすることが許可されているかどうか、および X.509 デジタル証明書を使用することによって、クラスター要求の妥当性を検査する必要があるかどうか分かります。デジタル証明書は、電子的に検証できる身分証明書のような形式になっています。妥当性検査が必要とされた場合、要求を出しているノードと追加しようとしているノードには、以下のものがシステムにインストールされていなければなりません。

- i5/OS オプション 34 (デジタル証明書マネージャー)
- i5/OS オプション 35 (CCA 暗号サービス・プロバイダー)

ALWADDCLU に対して *RQSAUT を選択した場合は、i5/OS クラスター・セキュリティー・サーバー・アプリケーションの認証局信頼リストを正しく設定する必要があります。このサーバー・アプリケーションの ID は、QIBM_QCST_CLUSTER_SECURITY です。少なくとも、クラスターへの加入を許可するノードについては、認証局を追加してください。

ノードの追加:

クラスター・リソース・サービスのグラフィカル・インターフェースを使用すると、最初にクラスターを作成するときに、単純な 2 つのノードのクラスターを作成することができます。i5/OS 高可用性ソリューション内で、クラスターにノードを追加することができます。

高可用性ソリューションの一部として新規クラスターを作成した場合、クラスター内のアクティブ・ノードを介してノードを追加する必要があります。

既存クラスターへノードを追加するには、以下の手順で行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・ノードの処理**」タスクを選択して、クラスター内のノードのリストを表示します。
5. 「**ノード**」タブで、「**アクションの選択**」メニューをクリックし、「**ノードの追加**」アクションを選択します。「**ノードの追加**」ページが表示されます。
6. 「**ノードの追加**」ページで、新規ノードの情報を指定します。「**OK**」をクリックして、ノードを追加します。新規ノードが、ノードのリストに表示されます。1 つのクラスターには、最大 128 ノードを組み込むことができます。

ノードの開始:

クラスター・ノードを開始すると、i5/OS 高可用性環境のノードでクラスタリングおよびクラスター・リソース・サービスが開始されます。

ノードは、自分自身を開始することができ、クラスターにアクティブ・ノードを検出した場合には、現在アクティブなクラスターと再結合することができます。

ノードでクラスタリングを開始するには、以下の手順で行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**ノード**」タブで、開始するノードを選択します。
5. 「**アクションの選択**」メニューをクリックして、「**開始**」を選択します。指定されたノードでクラスター・リソース・サービスが正常に開始されると、ノードの状況が「**開始済み**」に設定されます。

デバイス・ドメインへのノードの追加:

デバイス・ドメインとは、デバイス・リソースを共有するクラスター内のノードのサブセットのことです。

- 1 切り替えディスク、サイト間ミラーリングなどといった、独立ディスク・プール・ベースのテクノロジーを
- 1 含む高可用性ソリューションをインプリメントするには、ノードをデバイス・ドメインのメンバーとして定
- 1 義する必要があります。ノードをデバイス・ドメインに追加すると、クラスターのリカバリー・ドメインを
- 1 定義するデバイス・クラスター・リソース・グループ (CRG) を作成することができます。装置 CRG のリ
- 1 カバリー・ドメインに入るすべてのノードは、同じデバイス・ドメインに所属している必要があります。ク
- 1 ラスター・ノードは、単一のデバイス・ドメインにしか所属することができません。

デバイス・ドメインの作成および管理を行うには、i5/OS オプション 41 (HA 切り替え可能リソース) をインストールする必要があります。デバイス・ドメインのすべてのクラスター・ノード上に、有効なライセンス・キーが存在しなければなりません。

ノードをデバイス・ドメインに追加するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで「クラスター・ノードの処理」タスクを選択して、クラスターにあるノードのリストを表示します。
5. 「ノード」タブで、デバイス・ドメインに追加するノードを選択します。
6. 「アクションの選択」メニューで、「プロパティ」を選択します。
7. 「クラスター化」タブで、「装置ドメイン」フィールドのノードの追加先となるデバイス・ドメインの名前を指定します。

クラスター・リソース・グループ (CRG) の作成:

クラスター・リソース・グループ (CRG) は、アプリケーション、データ、装置などといった、高可用性リソースを管理します。高可用性環境における特定のリソース・タイプを、それぞれの CRG タイプで管理します。

クラスター・リソース・サービスのグラフィカル・インターフェースを使うと、高可用性リソースの管理用にさまざまな CRG を作成することができます。CRG タイプは、それぞれ個別に使用することもでき、他の CRG と一緒に使用することもできます。例えば、ご使用のスタンドアロン・ビジネス・アプリケーションで、高可用性が必要になったとします。アプリケーションで高可用性を使用可能にした後に CRG を作成することで、そのアプリケーションの可用性の管理に役立てることができます。

データではなくアプリケーションのみを、停止時に使用できるようにしたい場合、アプリケーション CRG を作成することができます。ただし、データとアプリケーションの両方を使用可能にしたい場合には、独立ディスク・プールに両方とも保管することができます (これは装置 CRG で定義できます)。停止が発生した場合、独立ディスク・プール全体がバックアップ・ノードに切り替わり、アプリケーションとそのデータの両方が使用可能になります。

アプリケーション CRG の作成:

高可用性ソリューションのアプリケーションに高可用性を持たせたい場合、アプリケーション・クラスター・リソース・グループ (CRG) を作成して、そのアプリケーションのフェイルオーバーを管理することができます。

アプリケーション CRG の作成時に、アクティブなテークオーバー IP アドレスを許可するよう指定することができます。アクティブなテークオーバー IP アドレスが許可されたアプリケーション CRG を開始すると、その CRG の開始が許可されます。

アプリケーション CRG を作成するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。

4. 「クラスター・リソース・サービス」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、「**アクションの選択**」メニューをクリックします。
6. 「**新規アプリケーション CRG**」を選択し、「**実行**」をクリックします。「新規アプリケーション CRG」ページが表示されます。
7. 「**一般**」ページで、アプリケーション CRG に関する以下の情報を指定します。
 - 「**名前**」フィールドで、CRG の名前を指定します。10 文字を超える名前は付けられません。
 - 「**テークオーバー IP アドレス**」フィールドで、アプリケーション CRG に関連付ける IP アドレスを指定します。この値は、小数点付き 10 進数の形式で、最大 15 文字までにします。テークオーバー IP アドレスを使うと、アプリケーションが現在実行されているシステムに関係なく、アプリケーションにアクセスすることができます。「**テークオーバー IP アドレスの構成**」フィールドは、ユーザーとクラスター・リソース・サービスのどちらが、IP アドレスの作成を担当するのかを判別します。
 - 「**記述**」フィールドに、CRG の説明を入力します。50 文字を超える説明は入力できません。
 - 「**再始動を許可**」を選択して、アプリケーション CRG に対する再始動の試行回数を指示します。この値によって、バックアップ・ノードへのフェイルオーバーが発生する前に、同一のノードでアプリケーションを再始動する試行回数が決まります。
 - 「**テークオーバー IP アドレスの構成**」フィールドで、アプリケーション CRG のテークオーバー IP アドレスの構成および管理を、クラスター・リソース・サービスとユーザーのどちらに担当させるかを選択します。使用可能な値は、次のとおりです。

クラスター・リソース・サービス

この値を指定する場合、CRG を作成する前に、リカバリー・ドメイン内のノード上にテークオーバー IP アドレスが存在してはなりません。すべてのリカバリー・ドメイン・ノード上に、これが作成されています。IP アドレスがすでに存在している場合、アプリケーション CRG の作成は失敗します。

ユーザー

この値を指定する場合、CRG を開始する前に、リカバリー・ドメインで定義されているすべてのプライマリー・ノードとバックアップ・ノード上に、テークオーバー IP アドレスを追加する必要があります。

- 「**アクティブ・テークオーバー IP アドレスの許可**」を選択して、テークオーバー IP アドレスがアプリケーション CRG に割り当てられるときにアクティブになるようにします。「テークオーバー IP アドレスの構成」フィールドが「クラスター・リソース・サービス」に設定されている場合にのみ、このフィールドは有効になります。
- 「**配布情報ユーザー待ち行列**」フィールドで、配布情報を受け取るユーザー待ち行列の名前を指示します。10 文字を超える名前は付けられません。「**ライブラリー**」フィールドで、配布情報を受け取るユーザー待ち行列が含まれるライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBL などをライブラリー名にすることはできません。10 文字を超える名前は付けられません。

注: 「配布情報ユーザー待ち行列」の設定をブランクにした場合、「ライブラリー」名をブランクに、「フェイルオーバー待ち時間」を 0 に、「フェイルオーバー・デフォルト・アクション」を 0 に、それぞれ設定する必要があります。

- 「**フェイルオーバー・メッセージ待ち行列**」フィールドで、このクラスター・リソース・グループでフェイルオーバーが発生したときにメッセージを受け取る、メッセージ・キューの名前を指定します。このフィールドが設定されている場合、出口プログラムが完了した後に、指定されたメッセージ・キューがリカバリー・ドメイン内のすべてのノード上に存在していなければなりません。フェイ

ルオーバー・メッセージ・キューを独立ディスク・プール内に置くことはできません。「ライブラリー」フィールドで、フェイルオーバー・メッセージを受け取るメッセージ・キューが含まれる、ライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBLなどをライブラリー名にすることはできません。

- 「フェイルオーバー待ち時間」フィールドで、クラスター・メッセージ・キューでフェイルオーバー・メッセージに対する応答を待つ分数を指定する。使用可能な値は、次のとおりです。

待機しない

ユーザー介入なしでフェイルオーバーが継続します。

無期限に待機

フェイルオーバー照会メッセージに対する応答が受け取られるまで、フェイルオーバーは無期限に待機します。

数値 フェイルオーバー照会メッセージに対する応答を待機する分数を指定します。指定した分数内に応答が受け取られない場合、「フェイルオーバー・デフォルト・アクション」フィールド内の値によって、継続方法が指定されます。

- 「フェイルオーバー・デフォルト・アクション」フィールドで、クラスター・メッセージ・キュー上のフェイルオーバー・メッセージに対する応答が、フェイルオーバー待ち時間の制限内に受け取られなかった場合に、どのようなクラスタリングを行うかを指定します。このフィールドは、「フェイルオーバーの続行」または「フェイルオーバーの取り消し」のいずれかに設定することができます。

8. 「出口プログラム」ページで、CRG 出口プログラムの情報を指定することができます。装置 CRG の例外を持つすべての CRG タイプで、出口プログラムが必要になります。CRG のクラスター関連イベントが発生し、そのイベントに対する応答が行われた後に、出口プログラムが呼び出されます。

9. 「リカバリー・ドメイン」ページで、リカバリー・ドメインにノードを追加し、クラスター内での役割を指定します。

データ CRG の作成:

データ・クラスター・リソース・グループ (CRG) は、主に論理複製アプリケーション (複数の高可用性ビジネス・パートナー各社によって提供されています) で使用されます。論理複製に基づいて高可用性ソリューションをインプリメントする場合、データ CRG を作成して、プライマリー・ノードとバックアップ・ノード間のデータの複製を支援することができます。

データ CRG を作成するには、以下の手順を実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで、「クラスター・リソース・グループの処理」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、「アクションの選択」メニューをクリックします。
6. 「新規データ CRG」を選択して、「実行」をクリックします。「新規データ CRG」ページが表示されます。
7. 「一般」ページで、データ CRG に関する以下の情報を指定します。
 - 「名前」フィールドで、CRG の名前を指定します。10 文字を超える名前は付けられません。
 - 「記述」フィールドに、CRG の説明を入力します。50 文字を超える説明は入力できません。

- 「配布情報ユーザー待ち行列」フィールドで、配布情報を受け取るユーザー待ち行列の名前を指示します。10文字を超える名前は付けられません。「ライブラリー」フィールドで、配布情報を受け取るユーザー待ち行列が含まれるライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBLなどをライブラリー名にすることはできません。10文字を超える名前は付けられません。

注: 「配布情報ユーザー待ち行列」の設定をブランクにした場合、「ライブラリー」名をブランクに、「フェイルオーバー待ち時間」を0に、「フェイルオーバー・デフォルト・アクション」を0に、それぞれ設定する必要があります。

- 「フェイルオーバー・メッセージ待ち行列」フィールドで、このクラスター・リソース・グループでフェイルオーバーが発生したときにメッセージを受け取る、メッセージ・キューの名前を指定します。このフィールドが設定されている場合、出口プログラムが完了した後に、指定されたメッセージ・キューがリカバリー・ドメイン内のすべてのノード上に存在していなければなりません。フェイルオーバー・メッセージ・キューを独立ディスク・プール内に置くことはできません。「ライブラリー」フィールドで、フェイルオーバー・メッセージを受け取るメッセージ・キューが含まれる、ライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBLなどをライブラリー名にすることはできません。
- 「フェイルオーバー待ち時間」フィールドで、クラスター・メッセージ・キューでフェイルオーバー・メッセージに対する応答を待つ分数を指定する。使用可能な値は、次のとおりです。

待機しない

ユーザー介入なしでフェイルオーバーが継続します。

無期限に待機

フェイルオーバー照会メッセージに対する応答が受け取られるまで、フェイルオーバーは無期限に待機します。

数値 フェイルオーバー照会メッセージに対する応答を待機する分数を指定します。指定した分数内に応答が受け取られない場合、「フェイルオーバー・デフォルト・アクション」フィールド内の値によって、継続方法が指定されます。

- 「出口プログラム」ページで、CRG 出口プログラムの情報を指定することができます。装置 CRG の例外を持つすべての CRG タイプで、出口プログラムが必要になります。CRG のクラスター関連イベントが発生し、そのイベントに対する応答が行われた後に、出口プログラムが呼び出されます。
- 「リカバリー・ドメイン」ページで、リカバリー・ドメインにノードを追加し、クラスター内での役割を指定します。

装置 CRG の作成:

装置クラスター・リソース・グループ (CRG) は、エンティティーとして切り替えることができる、ハードウェア・リソースのプールで構成されています。高可用性ソリューション内に切り替え可能装置を作成するには、これらの装置を使用するノードを装置 CRG の一部とする必要があります。

装置 CRG を作成する前に、切り替え可能なリソースを共有するすべてのノードをデバイス・ドメインに追加します。

装置 CRG を作成するには、以下の手順を実行します。

- Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
- ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。

4. 「クラスター・リソース・サービス」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、「**アクションの選択**」メニューをクリックします。
6. 「**新規装置 CRG**」を選択して、「**実行**」をクリックします。「**新規装置 CRG**」ウィザードが表示されます。「**新規装置 CRG**」タスクは、リカバリー・ドメイン内のすべてのノードが開始される場合にのみ使用可能です。
7. 「**新規装置 CRG**」ウィザードの指示に従い、新規の装置 CRG を作成します。このウィザードを実行しながら、新規の装置 CRG を作成することもできます。また、新規の独立ディスク・プールを作成したり、既存のディスク・プールを使用するよう指定したりすることもできます。

装置 CRG によって、すべてのリカバリー・ドメイン・ノードで同一のハードウェア・リソース情報が保持され、リソース名が同じであるかどうかを検査されます。また、クラスター管理可能ドメインを構成して、構成オブジェクトの登録済み属性 (リソース名が含まれる場合があります) を、クラスター管理可能ドメインを通じて一致させるすることができます。サイト間ミラーリングを使用している場合は、各サイトの独立ディスク・プールとその他のタイプの切り替え可能装置に対して、個別の装置 CRG を作成してください。

対等 CRG の作成:

対等 CRG を作成すると、ロード・バランシング環境でのノードの役割を定義することができます。

クラスターで対等 CRG を作成するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「クラスター・リソース・サービス」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、「**アクションの選択**」メニューをクリックします。
6. 「**新規対等 CRG**」を選択して、「**実行**」をクリックします。「**新規対等 CRG**」ページが表示されます。
7. 「**一般**」ページで、対等 CRG に関する以下の情報を指定します。
 - 「**名前**」フィールドで、CRG の名前を指定します。10 文字を超える名前は付けられません。
 - 「**記述**」フィールドに、CRG の説明を入力します。50 文字を超える説明は入力できません。
 - 「**アプリケーション ID**」フィールドで、対等クラスター・リソース・グループのアプリケーション ID を `[VendorName].[ApplicationName]` の形式で指定します。例: MyCompany.MyApplication。50 文字を超える ID は指定できません。
8. 「**出口プログラム**」ページで、CRG 出口プログラムの情報を指定することができます。装置 CRG の例外を持つすべての CRG タイプで、出口プログラムが必要になります。CRG のクラスター関連イベントが発生し、そのイベントに対する応答が行われた後に、出口プログラムが呼び出されます。
9. 「**リカバリー・ドメイン**」ページで、リカバリー・ドメインにノードを追加し、クラスター内での役割を指定します。

CRG の開始:

クラスター・リソース・グループ (CRG) を開始すると、i5/OS 高可用性環境内でクラスタリングが活動化状態になります。

CRG を開始するには、次のタスクを行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「**クラスター・リソース・グループ**」タブで、開始する CRG の名前を選択します。
6. 「**アクションの選択**」メニューから、「**開始**」を選択します。「**状況**」列に、CRG が開始されたことが表示されます。

関連情報

クラスター資源グループの開始 (STRCRG) コマンド

クラスター資源グループの作成 (QcstCreateClusterResourceGroup) API (英語)

メッセージ・キューの指定

クラスター・メッセージ・キューまたはフェイルオーバー・メッセージ・キューのいずれかを指定することができます。これらのメッセージ・キューは、i5/OS 高可用性環境で障害の原因を判別するのに役立ちます。

クラスター・メッセージ・キューは、クラスター・レベルのメッセージに使用され、特定のノードにフェイルオーバーするすべてのクラスター・リソース・グループ (CRG) を制御するメッセージを 1 つ提供します。フェイルオーバー・メッセージ・キューは、CRG レベル・メッセージに使用され、フェイルオーバーする各 CRG に対してメッセージを 1 つ提供します。

クラスター・メッセージ・キューの指定

注: クラスターの作成ウィザードを実行してメッセージ・キューを指定することにより、クラスター・メッセージ・キューを使用するようにクラスターを構成することもできます。

クラスター・メッセージ・キューを指定するには、次の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・プロパティの表示**」をクリックします。
5. 「**クラスター・プロパティ**」ページで、「**クラスター・メッセージ待ち行列**」をクリックします。
6. 以下の情報を指定して、クラスター・メッセージ・キューを作成します。
 - 「**名前**」フィールドで、クラスター・レベルまたはノード・レベルでフェイルオーバーを処理するメッセージを受け取るための、メッセージ・キューの名前を指定します。ノード・レベルのフェイルオーバーの場合、同じ新規プライマリー・ノードですべてのクラスター・リソース・グループのフェイルオーバーを制御するメッセージが 1 つ送信されます。クラスター・リソース・グループが個々にフェイルオーバーする場合、そのクラスター・リソース・グループのフェイルオーバーを制御するメッセージが 1 つ送信されます。このメッセージは、新規プライマリー・ノードで送信されます。こ

のフィールドが設定されている場合、指定されたメッセージ・キューは、開始時にクラスター内のすべてのノード上に存在していなければなりません。メッセージ・キューを独立ディスク・プール内に置くことはできません。

- 「**ライブラリー**」フィールドで、フェイルオーバー・メッセージを受け取るメッセージ・キューが含まれる、ライブラリーの名前を指定します。
*CURLIB、QTEMP、*LIBL、*USRLIBL、*ALL、*ALLUSRなどをライブラリー名にすることはできません。
- 「**フェイルオーバー待ち時間**」フィールドで、「**待機しない not wait**」または「**無期限に待機**」のいずれかを選択するか、クラスター・メッセージ・キューでフェイルオーバー・メッセージに対する応答を待つ分数を指定します。
- 「**フェイルオーバー・デフォルト・アクション**」フィールドで、フェイルオーバー・メッセージに対する応答がフェイルオーバー応答時間の値を超えたときに、クラスター・リソース・サービスが行うアクションを指定します。このフィールドは、「**フェイルオーバーの続行**」または「**フェイルオーバーの取り消し**」のいずれかに設定することができます。

フェイルオーバー・メッセージ・キューの指定

フェイルオーバー・メッセージ・キューを指定するには、次のようなステップを行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードで、システムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. クラスター・リソース・グループのリストから、処理を行うクラスター・リソース・グループを選択します。
6. 「**クラスター・リソース・グループ**」ページで、「**アクションの選択**」メニューをクリックし、「**プロパティ**」を選択します。
7. 「**一般**」ページで、以下の値を指定して、フェイルオーバー・メッセージ・キューを指定します。
 - 「**フェイルオーバー・メッセージ待ち行列**」フィールドで、このクラスター・リソース・グループでフェイルオーバーが発生したときにメッセージを受け取る、メッセージ・キューの名前を指定します。このフィールドが設定されている場合、出口プログラムが完了した後に、指定されたメッセージ・キューがリカバリー・ドメイン内のすべてのノード上に存在していなければなりません。フェイルオーバー・メッセージ・キューを独立ディスク・プール内に置くことはできません。
 - 「**ライブラリー**」フィールドで、フェイルオーバー・メッセージを受け取るメッセージ・キューが含まれる、ライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBLなどをライブラリー名にすることはできません。
 - 「**フェイルオーバー待ち時間**」フィールドで、フェイルオーバー・メッセージ・キューでフェイルオーバー・メッセージに対する応答を待つ分数を指定します。フェイルオーバー・メッセージに対する応答が指定されたフェイルオーバー待ち時間を超えた場合に、クラスター・リソース・サービスが行うアクションを指定することもできます。

切り替えの実行

切り替えを実行することで、高可用性ソリューションをテストしたり、プライマリー・ノードの計画停止（バックアップ操作、計画されたシステム保守など）を処理したりすることができます。

1 手動で切り替えを実行すると、現行のプライマリー・ノードがバックアップ・ノードに切り替わります。クラスタ・リソース・グループのリカバリー・ドメインは、これらの役割を定義します。切り替えが発生すると、リカバリー・ドメインで現在定義されているノードの役割が、以下のように変更されます。

- 現行プライマリー・ノードに、最後のアクティブ・バックアップの役割が割り当てられる。
- 現行の最初のバックアップが、プライマリー・ノードの役割に割り当てられる。
- それ以降のバックアップは、バックアップの順序が 1 つ上に移動する。

1 切り替えを実行できるのは、状況がアクティブになっているアプリケーション、データ、およびデバイス CRG においてのみです。

注: デバイス CRG で切り替えを行うには、パフォーマンス上の理由で、ユーザー・プロファイル名、UID、および GUID を同期化する必要があります。クラスタ管理可能ドメインを使うと、ユーザー・プロファイルの同期を簡単に行うことができます。

リソースで切り替えを実行するには、以下の手順のようにします。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。`mysystem` はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスタ・リソース・サービス」を選択します。
4. 「クラスタ・リソース・サービス」ページで、「クラスタ・リソース・グループの処理」を選択し、クラスタ内のクラスタ・リソース・グループのリストを表示します。
5. 切り替えを実行する CRG を選択します。アプリケーション CRG、データ CRG、またはデバイス CRG を選択して、切り替えを実行することができます。
6. 「アクションの選択」メニューで、「SWITCH」を選択します。
7. 確認パネルで、「はい」を選択します。

選択されたクラスタ・リソース・グループが、ここでバックアップ・ノードに切り替えられます。「状況」列が、新規ノード名に更新されます。

関連概念

クラスタ管理可能ドメイン

関連タスク

110 ページの『クラスタ管理可能ドメインの構成』

高可用性環境では、アプリケーションと作動環境が、高可用性に参加するノード間で整合している必要があります。クラスタ管理可能ドメインは、i5/OS における環境回復のインプリメンテーションであり、作動環境がノード間で整合していることを確認するものです。

関連情報

CRG 1 次の変更 (CHGCRGPRI) コマンド (英語)

切り替えの開始 (QcstInitiateSwitchOver) API (英語)

ノードの構成

ノードとは、i5/OS 高可用性ソリューションに属しているシステムまたは論理区画のことです。

ノード構成には、複数のタスクが関連しています。クラスタの作成ウィザードを使用すると、単純な 2 つのノードのクラスタを構成できます。追加ノードを合計で最大 128 まで追加することができます。高可用性ソリューションを構成するテクノロジーによっては、追加のノード構成タスクが必要になる場合があります。

ノードの開始:

クラスター・ノードを開始すると、i5/OS 高可用性環境のノードでクラスタリングおよびクラスター・リソース・サービスが開始されます。

ノードは、自分自身を開始することができ、クラスターにアクティブ・ノードを検出した場合には、現在アクティブなクラスターと再結合することができます。

ノードでクラスタリングを開始するには、以下の手順で行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「ノード」タブで、開始するノードを選択します。
5. 「アクションの選択」メニューをクリックして、「開始」を選択します。指定されたノードでクラスター・リソース・サービスが正常に開始されると、ノードの状況が「開始済み」に設定されます。

関連情報

クラスター・ノードの開始 (STRCLUNOD) コマンド

クラスター・ノードの開始 (QcstStartClusterNode) API

ノードをクラスターに追加できるようにする:

ノードをクラスターに追加するためには、クラスターへの追加可能 (ALWADDCLU) ネットワーク属性の値を設定する必要があります。

クラスター・ノードとして設定するサーバーに対して、ネットワーク属性変更 (CHGNETA) コマンドを使用します。CHGNETA コマンドは、システムのネットワーク属性を変更します。ALWADDCLU ネットワーク属性によって、クラスターのノードとしてノードを追加することが、他のシステムで可能かどうかを指定します。

注: ネットワーク属性 ALWADDCLU を変更するには、*IOSYSCFG 権限がなければなりません。

使用可能な値は、次のとおりです。

***SAME**

値は変更されません。システム出荷時の設定値は *NONE です。

***NONE**

他のシステムは、このシステムをクラスターのノードとして追加できません。

***ANY** 他のどんなシステムも、このシステムをクラスターのノードとして追加できます。

***RQSAUT**

クラスター追加要求が認証された後ならば、他のどんなシステムもこのシステムをクラスターのノードとして追加できます。

ALWADDCLU ネットワーク属性を検査すれば、追加しようとしているノードをクラスターの一部とすることが許可されているかどうか、および X.509 デジタル証明書を使用することによって、クラスター要求の妥当性を検査する必要があるかどうか分かります。デジタル証明書は、電子的に検証できる身分証明書のような形式になっています。妥当性検査が必要とされた場合、要求を出しているノードと追加しようとしているノードには、以下のものがシステムにインストールされていなければなりません。

- i5/OS オプション 34 (デジタル証明書マネージャー)
- i5/OS オプション 35 (CCA 暗号サービス・プロバイダー)

ALWADDCLU に対して *RQSAUT を選択した場合は、i5/OS クラスタ・セキュリティー・サーバー・アプリケーションの認証局信頼リストを正しく設定する必要があります。このサーバー・アプリケーションの ID は、QIBM_QCST_CLUSTER_SECURITY です。少なくとも、クラスタへの加入を許可するノードについては、認証局を追加してください。

ノードの追加:

クラスタ・リソース・サービスのグラフィカル・インターフェースを使用すると、最初にクラスタを作成するときに、単純な 2 つのノードのクラスタを作成することができます。i5/OS 高可用性ソリューション内で、クラスタにノードを追加することができます。

高可用性ソリューションの一部として新規クラスタを作成した場合、クラスタ内のアクティブ・ノードを介してノードを追加する必要があります。

既存クラスタへノードを追加するには、以下の手順で行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログインします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスタ・リソース・サービス」を選択します。
4. 「クラスタ・リソース・サービス」ページで、「クラスタ・ノードの処理」タスクを選択して、クラスタ内のノードのリストを表示します。
5. 「ノード」タブで、「アクションの選択」メニューをクリックし、「ノードの追加」アクションを選択します。「ノードの追加」ページが表示されます。
6. 「ノードの追加」ページで、新規ノードの情報を指定します。「OK」をクリックして、ノードを追加します。新規ノードが、ノードのリストに表示されます。1 つのクラスタには、最大 128 ノードを組み込むことができます。

関連情報

クラスタ・ノード項目の追加 (ADDCLUNODE) コマンド

クラスタ・ノード項目の追加 (QcstAddClusterNodeEntry) API (英語)

デバイス・ドメインへのノードの追加:

デバイス・ドメインとは、デバイス・リソースを共有するクラスタ内のノードのサブセットのことです。

- 1 切り替えディスク、サイト間ミラーリングなどといった、独立ディスク・プール・ベースのテクノロジーを含む高可用性ソリューションをインプリメントするには、ノードをデバイス・ドメインのメンバーとして定義する必要があります。ノードをデバイス・ドメインに追加すると、クラスタのリカバリー・ドメインを定義するデバイス・クラスタ・リソース・グループ (CRG) を作成することができます。装置 CRG のリカバリー・ドメインに入るすべてのノードは、同じデバイス・ドメインに所属する必要があります。クラスタ・ノードは、単一のデバイス・ドメインにしか所属することができません。

デバイス・ドメインの作成および管理を行うには、i5/OS オプション 41 (HA 切り替え可能リソース) をインストールする必要があります。デバイス・ドメインのすべてのクラスタ・ノード上に、有効なライセンス・キーが存在しなければなりません。

ノードをデバイス・ドメインに追加するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。`mysystem` はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで「**クラスター・ノードの処理**」タスクを選択して、クラスターにあるノードのリストを表示します。
5. 「**ノード**」タブで、デバイス・ドメインに追加するノードを選択します。
6. 「**アクションの選択**」メニューで、「**プロパティ**」を選択します。
7. 「**クラスター化**」タブで、「**装置ドメイン**」フィールドのノードの追加先となるデバイス・ドメインの名前を指定します。

関連情報

装置ドメイン項目の追加 (ADDDEVDMNE) コマンド

装置ドメイン項目の追加 (QcstAddDeviceDomainEntry) API (英語)

CRG の構成

クラスター・リソース・グループ (CRG) は、i5/OS 高可用性環境内のリソースを管理します。CRG を通じて、複数のタスクによる高可用性リソースの管理が可能になります。

CRG の開始:

クラスター・リソース・グループ (CRG) を開始すると、i5/OS 高可用性環境内でクラスタリングが活動化状態になります。

CRG を開始するには、次のタスクを行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。`mysystem` はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「**クラスター・リソース・グループ**」タブで、開始する CRG の名前を選択します。
6. 「**アクションの選択**」メニューから、「**開始**」を選択します。「**状況**」列に、CRG が開始されたことが表示されます。

関連情報

クラスター資源グループの開始 (STRCRG) コマンド

クラスター資源グループの作成 (QcstCreateClusterResourceGroup) API (英語)

クラスター・リソース・グループ (CRG) の作成:

クラスター・リソース・グループ (CRG) は、アプリケーション、データ、装置などといった、高可用性リソースを管理します。高可用性環境における特定のリソース・タイプを、それぞれの CRG タイプで管理します。

クラスター・リソース・サービスのグラフィカル・インターフェースを使うと、高可用性リソースの管理用にさまざまな CRG を作成することができます。CRG タイプは、それぞれ個別に使用することもでき、他の CRG と一緒に使用することもできます。例えば、ご使用のスタンドアロン・ビジネス・アプ

アプリケーションで、高可用性が必要になったとします。アプリケーションで高可用性を使用可能にした後に CRG を作成することで、そのアプリケーションの可用性の管理に役立てることができます。

データではなくアプリケーションのみを、停止時に使用できるようにしたい場合、アプリケーション CRG を作成することができます。ただし、データとアプリケーションの両方を使用可能にしたい場合には、独立ディスク・プールに両方とも保管することができます (これは装置 CRG で定義できます)。停止が発生した場合、独立ディスク・プール全体がバックアップ・ノードに切り替わり、アプリケーションとそのデータの両方が使用可能になります。

アプリケーション CRG の作成:

高可用性ソリューションのアプリケーションに高可用性を持たせたい場合、アプリケーション・クラスター・リソース・グループ (CRG) を作成して、そのアプリケーションのフェイルオーバーを管理することができます。

アプリケーション CRG の作成時に、アクティブなテークオーバー IP アドレスを許可するよう指定することができます。アクティブなテークオーバー IP アドレスが許可されたアプリケーション CRG を開始すると、その CRG の開始が許可されます。

アプリケーション CRG を作成するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「**クラスター・リソース・グループ**」ページで、「**アクションの選択**」メニューをクリックします。
6. 「**新規アプリケーション CRG**」を選択し、「**実行**」をクリックします。「**新規アプリケーション CRG**」ページが表示されます。
7. 「**一般**」ページで、アプリケーション CRG に関する以下の情報を指定します。
 - 「**名前**」フィールドで、CRG の名前を指定します。10 文字を超える名前は付けられません。
 - 「**テークオーバー IP アドレス**」フィールドで、アプリケーション CRG に関連付ける IP アドレスを指定します。この値は、小数点付き 10 進数の形式で、最大 15 文字までにします。テークオーバー IP アドレスを使うと、アプリケーションが現在実行されているシステムに関係なく、アプリケーションにアクセスすることができます。「**テークオーバー IP アドレスの構成**」フィールドは、ユーザーとクラスター・リソース・サービスのどちらが、IP アドレスの作成を担当するのかを判別します。
 - 「**記述**」フィールドに、CRG の説明を入力します。50 文字を超える説明は入力できません。
 - 「**再始動を許可**」を選択して、アプリケーション CRG に対する再始動の試行回数を指示します。この値によって、バックアップ・ノードへのフェイルオーバーが発生する前に、同一のノードでアプリケーションを再始動する試行回数が増えます。
 - 「**テークオーバー IP アドレスの構成**」フィールドで、アプリケーション CRG のテークオーバー IP アドレスの構成および管理を、クラスター・リソース・サービスとユーザーのどちらに担当させるかを選択します。使用可能な値は、次のとおりです。

クラスター・リソース・サービス

この値を指定する場合、CRG を作成する前に、リカバリー・ドメイン内のノード上にテーク

オーバー IP アドレスが存在してはなりません。すべてのリカバリー・ドメイン・ノード上に、これが作成されています。IP アドレスがすでに存在している場合、アプリケーション CRG の作成は失敗します。

ユーザー

この値を指定する場合、CRG を開始する前に、リカバリー・ドメインで定義されているすべてのプライマリー・ノードとバックアップ・ノード上に、テークオーバー IP アドレスを追加する必要があります。

- 「**アクティブ・テークオーバー IP アドレスの許可**」を選択して、テークオーバー IP アドレスがアプリケーション CRG に割り当てられるときにアクティブになるようにします。「テークオーバー IP アドレスの構成」フィールドが「クラスター・リソース・サービス」に設定されている場合にのみ、このフィールドは有効になります。
- 「**配布情報ユーザー待ち行列**」フィールドで、配布情報を受け取るユーザー待ち行列の名前を指示します。10 文字を超える名前は付けられません。「**ライブラリー**」フィールドで、配布情報を受け取るユーザー待ち行列が含まれるライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBL などをライブラリー名にすることはできません。10 文字を超える名前は付けられません。

注: 「配布情報ユーザー待ち行列」の設定をブランクにした場合、「ライブラリー」名をブランクに、「フェイルオーバー待ち時間」を 0 に、「フェイルオーバー・デフォルト・アクション」を 0 に、それぞれ設定する必要があります。

- 「**フェイルオーバー・メッセージ待ち行列**」フィールドで、このクラスター・リソース・グループでフェイルオーバーが発生したときにメッセージを受け取る、メッセージ・キューの名前を指定します。このフィールドが設定されている場合、出口プログラムが完了した後に、指定されたメッセージ・キューがリカバリー・ドメイン内のすべてのノード上に存在していなければなりません。フェイルオーバー・メッセージ・キューを独立ディスク・プール内に置くことはできません。「**ライブラリー**」フィールドで、フェイルオーバー・メッセージを受け取るメッセージ・キューが含まれる、ライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBL などをライブラリー名にすることはできません。
- 「**フェイルオーバー待ち時間**」フィールドで、クラスター・メッセージ・キューでフェイルオーバー・メッセージに対する応答を待つ分数を指定する。使用可能な値は、次のとおりです。

待機しない

ユーザー介入なしでフェイルオーバーが継続します。

無期限に待機

フェイルオーバー照会メッセージに対する応答が受け取られるまで、フェイルオーバーは無期限に待機します。

数値 フェイルオーバー照会メッセージに対する応答を待機する分数を指定します。指定した分数内に応答が受け取られない場合、「フェイルオーバー・デフォルト・アクション」フィールド内の値によって、継続方法が指定されます。

- 「**フェイルオーバー・デフォルト・アクション**」フィールドで、クラスター・メッセージ・キュー上のフェイルオーバー・メッセージに対する応答が、フェイルオーバー待ち時間の制限内に受け取られなかった場合に、どのようなクラスタリングを行うかを指定します。このフィールドは、「**フェイルオーバーの続行**」または「**フェイルオーバーの取り消し**」のいずれかに設定することができます。

8. 「**出口プログラム**」ページで、CRG 出口プログラムの情報を指定することができます。装置 CRG の例外を持つすべての CRG タイプで、出口プログラムが必要になります。CRG のクラスター関連イベントが発生し、そのイベントに対する応答が行われた後に、出口プログラムが呼び出されます。

9. 「リカバリー・ドメイン」 ページで、リカバリー・ドメインにノードを追加し、クラスター内での役割を指定します。

関連情報

クラスター資源グループの作成 (CRTCRG) コマンド

クラスター資源グループの作成 (QcstCreateClusterResourceGroup) API (英語)

データ CRG の作成:

データ・クラスター・リソース・グループ (CRG) は、主に論理複製アプリケーション (複数の高可用性ビジネス・パートナー各社によって提供されています) で使用されます。論理複製に基づいて高可用性ソリューションをインプリメントする場合、データ CRG を作成して、プライマリー・ノードとバックアップ・ノード間のデータの複製を支援することができます。

データ CRG を作成するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」 ページで、「クラスター・リソース・グループの処理」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」 ページで、「アクションの選択」メニューをクリックします。
6. 「新規データ CRG」を選択して、「実行」をクリックします。「新規データ CRG」 ページが表示されます。
7. 「一般」 ページで、データ CRG に関する以下の情報を指定します。
 - 「名前」 フィールドで、CRG の名前を指定します。10 文字を超える名前は付けられません。
 - 「記述」 フィールドに、CRG の説明を入力します。50 文字を超える説明は入力できません。
 - 「配布情報ユーザー待ち行列」 フィールドで、配布情報を受け取るユーザー待ち行列の名前を指示します。10 文字を超える名前は付けられません。「ライブラリー」 フィールドで、配布情報を受け取るユーザー待ち行列が含まれるライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBL などをライブラリー名にすることはできません。10 文字を超える名前は付けられません。

注: 「配布情報ユーザー待ち行列」 の設定をブランクにした場合、「ライブラリー」名をブランクに、「フェイルオーバー待ち時間」を 0 に、「フェイルオーバー・デフォルト・アクション」を 0 に、それぞれ設定する必要があります。

- 「フェイルオーバー・メッセージ待ち行列」 フィールドで、このクラスター・リソース・グループでフェイルオーバーが発生したときにメッセージを受け取る、メッセージ・キューの名前を指定します。このフィールドが設定されている場合、出口プログラムが完了した後に、指定されたメッセージ・キューがリカバリー・ドメイン内のすべてのノード上に存在していなければなりません。フェイルオーバー・メッセージ・キューを独立ディスク・プール内に置くことはできません。「ライブラリー」 フィールドで、フェイルオーバー・メッセージを受け取るメッセージ・キューが含まれる、ライブラリーの名前を指定します。*CURLIB、QTEMP、*LIBL などをライブラリー名にすることはできません。
- 「フェイルオーバー待ち時間」 フィールドで、クラスター・メッセージ・キューでフェイルオーバー・メッセージに対する応答を待つ分数を指定する。使用可能な値は、次のとおりです。

待機しない

ユーザー介入なしでフェイルオーバーが継続します。

無期限に待機

フェイルオーバー照会メッセージに対する応答が受け取られるまで、フェイルオーバーは無期限に待機します。

数値 フェイルオーバー照会メッセージに対する応答を待機する分数を指定します。指定した分数内に応答が受け取られない場合、「フェイルオーバー・デフォルト・アクション」フィールド内の値によって、継続方法が指定されます。

- 「**出口プログラム**」ページで、CRG 出口プログラムの情報を指定することができます。装置 CRG の例外を持つすべての CRG タイプで、出口プログラムが必要になります。CRG のクラスター関連イベントが発生し、そのイベントに対する応答が行われた後に、出口プログラムが呼び出されます。
- 「**リカバリー・ドメイン**」ページで、リカバリー・ドメインにノードを追加し、クラスター内での役割を指定します。

関連情報

クラスター資源グループの作成 (CRTCRG) コマンド

クラスター資源グループの作成 (QcstCreateClusterResourceGroup) API (英語)

装置 CRG の作成:

装置クラスター・リソース・グループ (CRG) は、エンティティとして切り替えることができる、ハードウェア・リソースのプールで構成されています。高可用性ソリューション内に切り替え可能装置を作成するには、これらの装置を使用するノードを装置 CRG の一部とする必要があります。

装置 CRG を作成する前に、切り替え可能なリソースを共有するすべてのノードをデバイス・ドメインに追加します。

装置 CRG を作成するには、以下の手順を実行します。

- Web ブラウザーで、`http://mysystem:2001` と入力します。`mysystem` はシステムのホスト名です。
- ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
- 「**クラスター・リソース・サービス**」ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
- 「**クラスター・リソース・グループ**」ページで、「**アクションの選択**」メニューをクリックします。
- 「**新規装置 CRG**」を選択して、「**実行**」をクリックします。「**新規装置 CRG**」ウィザードが表示されます。「**新規装置 CRG**」タスクは、リカバリー・ドメイン内のすべてのノードが開始される場合にのみ使用可能です。
- 「**新規装置 CRG**」ウィザードの指示に従い、新規の装置 CRG を作成します。このウィザードを実行しながら、新規の装置 CRG を作成することもできます。また、新規の独立ディスク・プールを作成したり、既存のディスク・プールを使用するよう指定したりすることもできます。

装置 CRG によって、すべてのリカバリー・ドメイン・ノードで同一のハードウェア・リソース情報が保持され、リソース名が同じであるかどうかを検査されます。また、クラスター管理可能ドメインを構成して、構成オブジェクトの登録済み属性 (リソース名が含まれる場合があります) を、クラスター管理可能ドメインを通じて一致させるすることができます。サイト間ミラーリングを使用している場合は、各サイトの独立ディスク・プールとその他のタイプの切り替え可能装置に対して、個別の装置 CRG を作成してください。

関連情報

- | クラスター資源グループの作成 (CRTCRG) コマンド
- | クラスター資源グループの作成 (QcstCreateClusterResourceGroup) API (英語)

| 対等 CRG の作成:

- | 対等 CRG を作成すると、ロード・バランシング環境でのノードの役割を定義することができます。
- | クラスターで対等 CRG を作成するには、以下の手順を実行します。
- | 1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
- | 2. ご使用のユーザー・プロファイルとパスワードでシステムにログインします。
- | 3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
- | 4. 「クラスター・リソース・サービス」ページで、「クラスター・リソース・グループの処理」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
- | 5. 「クラスター・リソース・グループ」ページで、「アクションの選択」メニューをクリックします。
- | 6. 「新規対等 CRG」を選択して、「実行」をクリックします。「新規対等 CRG」ページが表示されます。
- | 7. 「一般」ページで、対等 CRG に関する以下の情報を指定します。
 - | • 「名前」フィールドで、CRG の名前を指定します。10 文字を超える名前は付けられません。
 - | • 「記述」フィールドに、CRG の説明を入力します。50 文字を超える説明は入力できません。
 - | • 「アプリケーション ID」フィールドで、対等クラスター・リソース・グループのアプリケーション ID を `[VendorName].[ApplicationName]` の形式で指定します。例: MyCompany.MyApplication。50 文字を超える ID は指定できません。
- | 8. 「出口プログラム」ページで、CRG 出口プログラムの情報を指定することができます。装置 CRG の例外を持つすべての CRG タイプで、出口プログラムが必要になります。CRG のクラスター関連イベントが発生し、そのイベントに対する応答が行われた後に、出口プログラムが呼び出されます。
- | 9. 「リカバリー・ドメイン」ページで、リカバリー・ドメインにノードを追加し、クラスター内での役割を指定します。

| 関連情報

- | クラスター資源グループの作成 (CRTCRG) コマンド
- | クラスター資源グループの作成 (QcstCreateClusterResourceGroup) API (英語)

| クラスター管理可能ドメインの構成

- | 高可用性環境では、アプリケーションと作動環境が、高可用性に参加するノード間で整合している必要があります。クラスター管理可能ドメインは、i5/OS における環境回復のインプリメンテーションであり、作動環境がノード間で整合していることを確認するものです。

| クラスター管理可能ドメインの作成:

- | 高可用性ソリューションにおけるクラスター管理可能ドメインは、クラスター内のシステムおよび区画間における、リソースの同期化を保持するメカニズムを提供します。

- | クラスター管理可能ドメインを作成するには、*IOSYSCFG 権限と、QCLUSTER ユーザー・プロファイルに対する権限を、ユーザーが有する必要があります。クラスター管理可能ドメインを管理するには、クラスター管理可能ドメイン、QCLUSTER ユーザー・プロファイル、クラスター・リソース・グループ・コマンドをそれぞれ表す CRG に対する権限を、ユーザーが有する必要があります。

- クラスタ管理可能ドメインを作成するには、以下の手順で行います。
1. Web ブラウザーで、`http://mysystem:2001` と入力します。`mysystem` はシステムのホスト名です。
 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスタ・リソース・サービス**」を選択します。
 4. 「**クラスタ・リソース・サービス**」ページで、「**管理可能ドメインの処理**」をクリックし、クラスタ内のクラスタ管理可能ドメインをリストします。クラスタ管理可能ドメインが構成されていない場合、このリストは空になります。
 5. 「**管理可能ドメイン**」タブで、「**新規管理可能ドメイン**」を選択します。
 6. 「**新規管理可能ドメイン**」ページで、クラスタ管理可能ドメインに関する以下の情報を指定します。
 - 「**名前**」フィールドに、クラスタ管理可能ドメインの名前を入力します。10 文字を超える名前は付けられません。
 - 「**クラスタ**」フィールドに、クラスタの名前が表示されます。このフィールドの値は変更できません。
 - 「**同期オプション**」フィールドで、ノードがクラスタ管理可能ドメインを結合するときの同期動作を指定します。クラスタがバージョン 6 以降である場合にのみ、このフィールドを使用できます。使用可能な値は、次のとおりです。

最終変更オプション (デフォルト)

モニター対象リソースに対するすべての変更内容をクラスタ管理可能ドメインに適用する場合は、このオプションを選択します。モニター対象リソースに行われた最新の変更内容が、すべてのアクティブ・ノード上のリソースに適用されます。

アクティブ・ドメイン・オプション

モニター対象リソースに行った変更内容のみをアクティブ・ノードから許可する場合に、このオプションを選択します。クラスタ管理可能ドメインをノードが結合した場合、非アクティブ・ノード上でモニター対象リソースに行った変更は廃棄されます。ネットワーク・サーバー・ストレージ・スペース (*NWSSTG) またはネットワーク・サーバー構成 (*NWSCFG) には、アクティブ・ドメイン・オプションは適用されません。これらのリソースの同期化は、常に最後に変更された内容に基づいて行われます。

- 「**管理可能ドメイン内のノード**」リストから、クラスタ管理可能ドメインに追加するノードを選択し、「**追加**」を選択します。

関連概念

80 ページの『すべてのノード上のユーザー・プロファイルの保守』

クラスタ内のすべてのノードにあるユーザー・プロファイルを保守するために、2 種類のメカニズムを使用することができます。

関連情報

クラスタ管理可能ドメインの作成 (CRTCAD) コマンド

クラスタ管理可能ドメインの作成 (QcstCrtClusterAdminDomain) API (英語)

クラスタ管理可能ドメインへのノードの追加:

- 高可用性ソリューション内で、クラスタ管理可能ドメインにノードを追加することができます。

1 クラスター管理可能ドメインにノードを追加する前に、そのノードが、クラスター管理可能ドメインがある
1 クラスターの一部でもあることを確認します。そうでない場合、クラスター管理可能ドメインにノードを追
1 加することはできません。クラスター管理可能ドメインがアクティブである必要はありませんが、アクティ
1 ブになるまでリソースは整合されません。

1 管理可能ドメインにノードを追加すると、そのドメインの MRE が追加されるノードにコピーされます。
1 モニター対象リソースが新規ノードに存在しない場合は、クラスター管理可能ドメインによって作成されま
1 す。追加されるノード上にモニター対象リソースが既に存在している場合、ドメインがアクティブであれ
1 ば、残りのクラスター管理可能ドメインと同期化されます。つまり、結合するノード上の各モニター対象リ
1 ソースの属性値は、アクティブ・ドメインのモニター対象リソースのグローバル値と一致するように変更さ
1 れます。

1 クラスター管理可能ドメインにノードを追加するには、以下の手順で行います。

- 1 1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
- 1 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- 1 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選
1 択します。
- 1 4. 「**クラスター・リソース・サービス**」ページで、「**管理可能ドメインの処理**」をクリックして、クラス
1 ター内のクラスター管理可能ドメインのリストを表示します。
- 1 5. 「**管理可能ドメイン**」ページで、クラスター管理可能ドメインを選択します。
- 1 6. 「**アクションの選択**」メニューから「**プロパティ**」を選択します。
- 1 7. 「**プロパティ**」ページで、「**管理可能ドメイン内のノード**」のリストからクラスター管理可能ドメイ
1 ンに追加するノードを選択します。「**Add (追加)**」をクリックします。

1 関連情報

- 1 クラスター管理可能ドメイン・ノード項目の追加 (ADDCADNODE) コマンド (英語)
- 1 リカバリー・ドメインへのノードの追加 (QcstAddNodeToRcvyDomain) API (英語)

1 クラスター管理可能ドメインの開始:

1 クラスター管理可能ドメインにより、i5/OS 高可用性ソリューション内で、リソース環境面での回復力が得
1 られます。

1 クラスター管理可能ドメインが開始すると、クラスター管理可能ドメインの終了中にモニター対象リソースに加えら
1 れた変更は、クラスター管理可能ドメインのすべてのアクティブ・ノードに伝搬されます。

1 クラスター管理可能ドメインを開始するには、以下のステップを行います。

- 1 1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
- 1 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- 1 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選
1 択します。
- 1 4. 「**クラスター・リソース・サービス**」ページで、「**管理可能ドメインの処理**」をクリックして、クラス
1 ター内のクラスター管理可能ドメインのリストを表示します。
- 1 5. 「**管理可能ドメイン**」ページで、クラスター管理可能ドメインを選択します。
- 1 6. 「**アクションの選択**」メニューから、「**開始**」を選択します。

1 「**状況**」列にクラスター管理可能ドメインが開始されたことが表示されます。

関連概念

『モニター対象リソースの同期』

モニター対象リソースの同期は、モニター対象リソースがクラスター管理可能ドメインで定義されたノード上で変更されると行われます。

関連情報

クラスター管理可能ドメインの開始 (STRCAD) コマンド (英語)

モニター対象リソースの同期:

モニター対象リソースの同期は、モニター対象リソースがクラスター管理可能ドメインで定義されたノード上で変更されると行われます。

この同期化処理の間、クラスター管理可能ドメインは、リソースに保留中の変更がない限り、グローバル値と値が一致しない属性を持つ各リソースを変更しようと試みます。保留中の変更はドメイン内のすべてのアクティブ・ノードに配布され、各ノード上で影響を受けるリソースに適用されます。保留中の変更が配布される際にグローバル値は変更され、影響を受ける各リソースは、各ノード上での変更操作に基づいて整合または不整合に変更されます。影響を受けるリソースがドメイン内の各アクティブ・ノード上で正常に変更された場合、そのリソースのグローバル状況は整合していることとなります。ノード上で操作が失敗すると、グローバル状況は不整合に設定されます。

クラスター管理可能ドメインの非アクティブ時に複数ノードから同じリソースに対して変更が行われると、すべての変更は、同期化処理の一部としてドメインの始動時にすべてのアクティブ・ノードに伝搬されます。すべての保留中の変更はクラスター管理可能ドメインの活動化中に処理されますが、変更を処理する上での決まった順序はありません。クラスター管理可能ドメインの非アクティブ時に複数ノードから単一のリソースに対して変更を行った場合、活動化中に変更を処理する上で決まった順序はありません。

ノードが非アクティブのクラスター管理可能ドメインに結合される場合 (つまり、クラスター管理可能ドメインの終了中にノードが開始した場合)、モニター対象リソースはクラスター管理可能ドメインが始動するまで再同期されません。

注: クラスター管理可能ドメインおよび関連する出口プログラムは、IBM 提供のオブジェクトです。これらのオブジェクトは、QcstChangeClusterResourceGroup API またはクラスター資源グループの変更 (CHGCRG) コマンドを使用して変更しないでください。予測不能の結果が発生します。

クラスター管理可能ドメインの一部であるクラスター・ノードが終了しても、モニター対象リソースは非アクティブ・ノード上で変更可能です。ノードが再始動したときに、他のクラスター管理可能ドメインとの間で変更が再同期されます。再同期処理の間、ノードが非アクティブのときにアクティブ・ドメインでも変更が行われなかった場合に限り、クラスター管理可能ドメインは、ドメイン内の他のアクティブ・ノードに対して非アクティブであったノードからの変更をすべて適用します。アクティブ・ドメイン内および非アクティブ・ノード上の両方のモニター対象リソースに対して変更が行われた場合、アクティブ・ドメイン内で行われた変更は、結合ノードに適用されます。つまり、ノードの状況に関係なく、モニター対象リソースに加えられた変更は失われません。同期時の動作は、同期化オプションを指定することで制御できます。

クラスター管理可能ドメインの一部であるクラスター・ノードを終了し、ノード始動時に非アクティブ・ノード上の変更がアクティブ・ドメインに伝搬しないようにするには (例えば、クラスター・ノードを終了して、そのノードでテストする場合)、ノードを管理可能ドメイン・ピア CRG から削除してからクラスター・ノードを終了する必要があります。

関連概念

管理ドメイン・ノード項目の除去 (RMVCADNODE) コマンド

関連タスク

112 ページの『クラスター管理可能ドメインの開始』

クラスター管理可能ドメインにより、i5/OS 高可用性ソリューション内で、リソース環境面での回復力が得られます。

関連情報

CRG ノード項目の除去 (RMVCRGNODE) コマンド

モニター対象リソース項目の追加:

クラスター管理可能ドメインに、モニター対象リソース項目 (MRE) を追加することができます。モニター対象リソース項目はクリティカル・リソースを定義するため、これらのリソースに対して変更を行った場合、高可用性環境で矛盾がないように整合されます。

モニター対象リソース項目を追加するには、以下の手順で行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. 「クラスター・リソース・サービス」ページで、「管理可能ドメインの処理」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
4. 「管理可能ドメイン」ページで、クラスター管理可能ドメインの隣にあるコンテキスト・アイコンをクリックし、「モニター対象リソース・タイプの処理」を選択します。

注: 「モニター対象リソース・タイプの処理」アクションを使用できるのは、管理しているノードがクラスター管理可能ドメインの一部である場合のみです。モニター対象リソース・タイプの現行リストが表示されます。

5. モニター対象リソース・タイプのリストで、モニター対象リソース・タイプの隣にあるコンテキスト・アイコンをクリックし、「モニター対象リソース項目の追加」を選択します。「モニター対象リソース項目の追加」ページが表示されます。
6. モニター対象リソース項目としてモニターする属性を選択し、「OK」をクリックします。MRE オブジェクトがライブラリー内にある場合、そのオブジェクトの名前およびライブラリーを指定する必要があります。クラスター管理可能ドメインがモニターしているリソースのリストに、新規のモニター対象リソース項目が追加されます。ドメインがアクティブである場合に、モニター対象リソースに対して変更を行うと、クラスター管理可能ドメイン内のすべてのアクティブ・ノード間で同期化されます。デフォルトでは、モニター対象リソース・タイプに関連付けられたすべての属性がモニターされます。ただし、モニター対象とする属性を選択することにより、モニターする属性を制御することができます。

関連タスク

150 ページの『モニターする属性の選択』

モニター対象リソース項目を追加した後で、クラスター管理可能ドメインによってモニターされるそのリソースに関連付けられた属性を選択することができます。

関連情報

管理ドメイン MRE の追加 (ADDCADMRE) コマンド (英語)

モニター対象リソース項目の追加 (QfpadAddMonitoredResourceEntry) API (英語)

切り替えディスクの構成

切り替えディスクは、i5/OS クラスターの一部として構成された独立ディスク・プールです。切り替えディスクを使用すると、独立ディスク・プール内に保管されたデータおよびアプリケーションを、他のシステムに切り替えることができます。

独立ディスク・プールの作成

独立ディスク・プールを作成するために、「新規ディスク・プール」ウィザードを使用できます。新規ディスク・プールを作成して、それをディスク装置に追加する際に、このウィザードが役立ちます。

「新規ディスク・プール」ウィザードを使用すると、パリティ・セットに未構成のディスク装置を組み込んで、装置パリティ保護およびディスク圧縮を開始することができます。ディスク装置を追加する際に、同一のパリティ・セット内にあるディスク装置を、複数のディスク・プールに拡張しないでください。あるパリティ・セットで障害が発生した場合に、複数のディスク・プールに影響を及ぼす可能性があるためです。

IBM Systems Director Navigator for i5/OS の「新規ディスク・プール」ウィザードを使用して独立ディスク・プールを作成するには、以下の手順を実行します。

注: IBM Systems Director Navigator for i5/OS 内のディスクを処理するには、専用保守ツールに対して適切なパスワード構成を行う必要があります。

IBM Systems Director Navigator for i5/OS

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク装置」を選択します。
5. 「アクションの選択」メニューから、「新規ディスク・プール」を選択します。
6. ウィザードの指示に従って、ディスク装置を新規ディスク・プールに追加します。
7. ディスク構成をプリントして、リカバリー時に使用できるようにします。
8. 独立ディスク・プール名と番号との関係を記録します。

System i Navigator

System i ナビゲーターの「新規ディスク・プール」ウィザードを使用して独立ディスク・プールを作成するには、以下の手順を実行します。

1. System i ナビゲーターで、「ユーザー接続」（またはアクティブな環境）を展開します。
2. 調べたいシステムを展開し、「構成およびサービス」→「ハードウェア」→「ディスク装置」を展開します。
3. 「ディスク・プール」を右クリックして、「新規ディスク・プール」を選択します。
4. ウィザードの指示に従って、ディスク装置を新規ディスク・プールに追加します。
5. ディスク構成をプリントして、リカバリー時に使用できるようにします。
6. 独立ディスク・プール名と番号との関係を記録します。

注: システムが完全に再始動したら、独立ディスク・プールを追加します。専用保守ツール (DST) モードで「新規ディスク・プール」ウィザードを使用する必要がある場合、システムが完全に再始動するとき、独立ディスク・プールの関連装置記述を作成する必要があります。装置記述を作成するには、装置記述の作成 (ASP) (CRTDEVASP) コマンドを使用します。装置記述およびリソース名には、独立ディスク・プールに付けたのと同じ名前を付けてください。装置記述処理 (WRKDEVD) コマンドを使用すると、装置記述および独立ディスク・プール名が一致しているかどうかを検査できます。

ミラー保護の開始

「ディスク装置の追加 (Add Disk Unit)」および「新規ディスク・プール」ウィザードに従って、保護されたディスク・プールに同容量のディスク装置のペアを追加します。ディスクが正しく構成されると、ミラー保護されたミラーリングを開始する準備が整います。

ミラー保護は、単一システムに対する局所的なもので、サイト間ミラーリングとは区別されます。使用不可になっている独立ディスク・プール上でミラーリングを開始する場合は、システムの完全な再始動時に行います。その他のすべてのディスク・プールの場合は、システムを再始動して専用保守ツール (DST) モードにしてからミラー保護を開始します。

IBM Systems Director Navigator for i5/OS を使用してミラーリングを開始するには、以下のステップを行います。

1. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
2. 「ディスク・プール」を選択します。
3. ミラーリングするディスク・プールを選択します。
4. 「アクションの選択」メニューから、「ミラーリングの開始...」を選択します。

System i ナビゲーターを使用してミラーリングを開始するには、以下のステップを行います。

1. System i ナビゲーターで、「ユーザー接続」(またはアクティブな環境)を展開します。
2. 検査する System i を、「構成およびサービス」→「ハードウェア」→「ディスク装置」→「ディスク・プール」で展開します。
3. ミラーリングするディスク・プールを右クリックして、「ミラーリングの開始...」を選択します。

ミラー保護の停止

ミラー保護を停止すると、ミラー保護された各対のうち 1 つのディスク装置は未構成となります。ディスク・プールのミラー保護を停止する前に、ディスク・プール内で各ミラー保護された対のうち、少なくとも 1 つのディスク装置が存在しアクティブになっている必要があります。

ミラー保護されたディスク装置の各対のうち、未構成にするディスク装置をどれにするかを制御する場合、未構成にするディスク装置を中断します。中断状態にないディスク装置の場合は、自動的に選択が行われます。

使用不可になっている独立ディスク・プール上でミラーリングを停止する場合は、システムの完全な再始動時に行います。その他のすべてのディスク・プールの場合は、システムを再始動して専用保守ツール (DST) モードにしてからミラー保護を停止する必要があります。

ミラー保護は、単一システム専用で、サイト間ミラーリングとは区別されます。

IBM Systems Director Navigator for i5/OS を使用してミラーリングを停止するには、以下のステップを行います。

1. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
2. 「ディスク・プール」を選択します。
3. 停止するディスク・プールを選択します。
4. 「アクションの選択」メニューから、「ミラーリングを停止...」を選択します。

System i ナビゲーターを使用してミラー保護を停止するには、以下のステップを行います。

1. System i ナビゲーターで、「ユーザー接続」(またはアクティブな環境)を展開します。

2. 検査する System i を、「構成およびサービス」 → 「ハードウェア」 → 「ディスク装置」 → 「ディスク・プール」で展開します。
3. ミラー保護を停止するディスク装置を選択します。
4. 選択したディスク・プールを右クリックして、「ミラーリングを停止...」を選択します。
5. 確認ダイアログ・ボックスで、「ミラーリングを停止...」をクリックします。

ディスク装置またはディスク・プールを追加する

「ディスク装置の追加 (Add Disk Unit)」ウィザードで、既存のディスク・プールを使用して新規または未構成のディスク装置を追加することができます。

「ディスク装置の追加 (Add Disk Unit)」および「ディスク・プール」ウィザードは、時間のかかるいくつかの処理を 1 つの効率的な処理に組み込むことにより、時間を節約します。また、これらのウィザードによりディスク装置の構成を基に推量が行われますが、これはウィザードがシステムの能力を理解して有効な選択のみを提案できるためです。例えば、システムに圧縮の能力がなければ、圧縮を開始するオプションはウィザード上でリストされません。

保護されたディスク・プールにディスク装置を追加するように選択した場合、ウィザードに従い、ディスク装置をデバイス・パリティ保護に含めるか、同じ容量のディスク装置を追加してミラー保護を開始する必要がある場合があります。ウィザードには、それらがシステム構成で許容しうるアクションの場合、ディスク・プールを越えてデータのバランシングを行うか、ディスク圧縮を開始するオプションもあります。システムで操作が調整されるように、選択するオプションを決定します。

IBM Systems Director Navigator for i5/OS を使用してディスク装置またはディスク・プールを追加するには、以下のステップを行います。

1. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
2. 「ディスク装置」を選択します。
3. 「アクションの選択」メニューから、「ディスク装置の追加 (Add Disk Unit)」を選択します。
4. ウィザードの指示に従って、ディスク装置をディスク・プールに追加します。

System i ナビゲーターを使用してディスク装置またはディスク・プールを追加するには、以下のステップを行います。

1. System i ナビゲーターで、「ユーザー接続」(またはアクティブな環境)を展開します。
2. 「構成およびサービス」 → 「ハードウェア」 → 「ディスク装置」で、検査する System i を展開します。
3. ディスク装置を追加するには、「すべてのディスク装置 (All Disk Units)」を右クリックし、「ディスク装置の追加 (Add Disk Unit)」を選択します。
4. ウィザードの指示に従って、タスクを完了します。

現在の構成の評価

システムのディスク構成を変更する前に、ディスク・プール、IOA、およびフレームとの関連から、既存のディスク装置の正確な場所を知っておく必要があります。

System i ナビゲーターのグラフィカル・ビューは、システムの構成方法をグラフィカルに表示することにより、このすべての情報を収集する手間を省きます。グラフィカル・ビューを使用することで、System i ナビゲーターのディスク装置リスト・ビュー経由で可能な機能を、追加された利点であるビジュアル表示で

確認しながら実行できます。特定のディスク装置、ディスク・プール、パリティ・セット、またはフレームなど、表内のオブジェクトをどれか右クリックすると、System i ナビゲーターのウィンドウと同じオプションが表示されます。

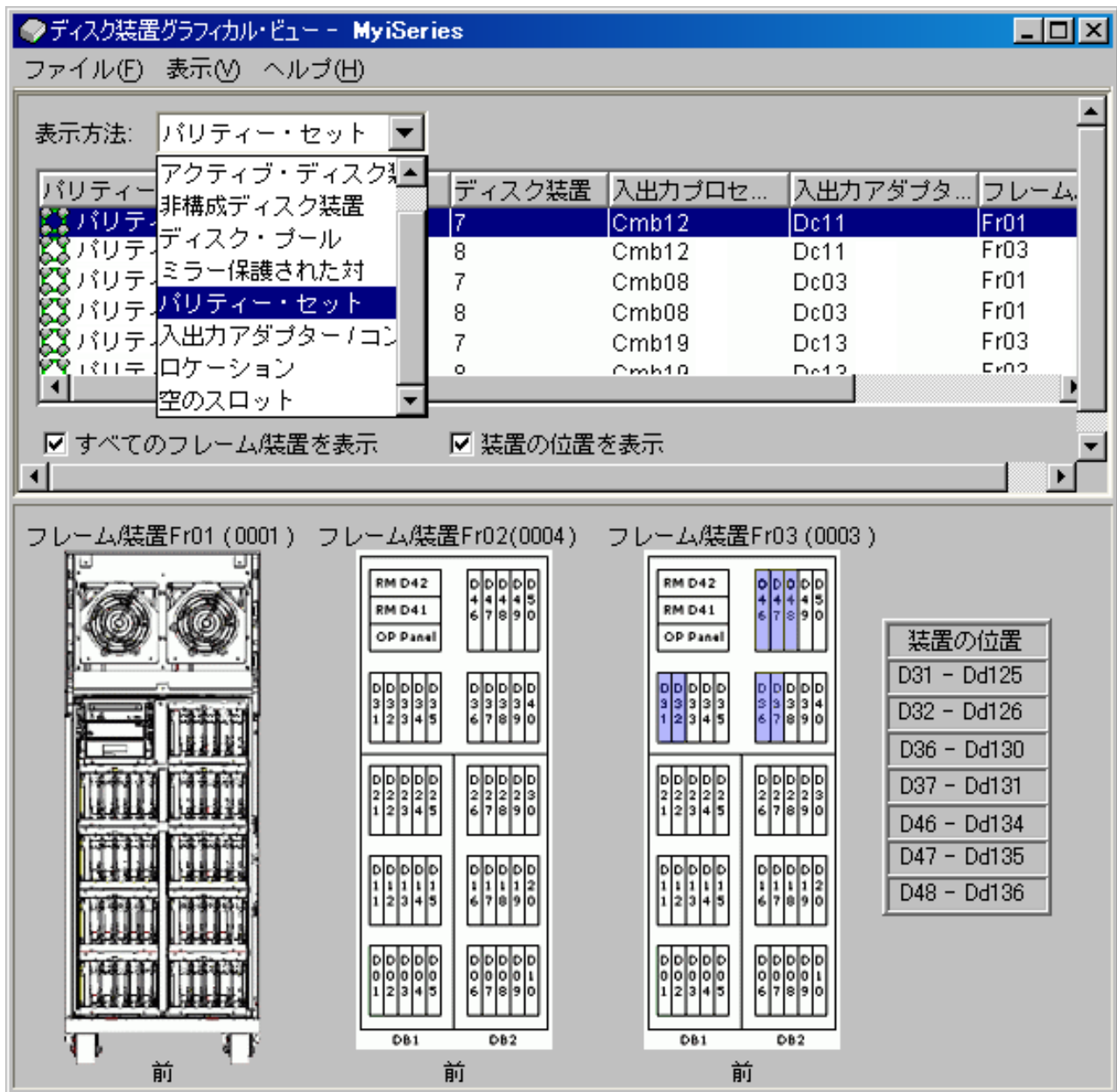
「ディスク装置のグラフィカル・ビュー (Disk Unit Graphical View)」ウィンドウでハードウェアの表示方法を選ぶことができます。例えば、ディスク・プール別の表示を選択してからリスト内のディスク・プールを選択し、選択したディスク・プールを構成するディスク装置が収容されているフレームのみを表示することができます。「すべてのフレームを表示 (Show all frames)」を選択して、選択したディスク・プールのディスク装置が収容されているかどうかに関係なく、すべてのフレームを表示することができます。また、「デバイス・オプションを表示 (Show device positions)」を選択して、ディスク装置名をデバイスが挿入された位置と関連付けることも可能です。

グラフィカル・ビューで強調表示された青色のディスク装置を右クリックして、ディスク装置上で実行するアクションを選択できます。例えば、ディスク装置上における圧縮の開始または停止、パリティ・セット内におけるディスク装置の組み込みまたは除外、あるいは、ディスク装置のリネームを選択することができます。ディスク装置がミラー保護されている場合は、ディスク装置上のミラーリングを中断または再開することが可能です。空のディスク装置スロットを右クリックすると、「ディスク装置の取り付け (Install Disk Unit)」ウィザードを開始します。

グラフィカル・ビューをアクティブにするには、次のステップを行います。

1. System i ナビゲーターで、「**ユーザー接続**」 (またはアクティブな環境) を展開します。
2. 「**構成およびサービス**」 → 「**ハードウェア**」 → 「**ディスク装置**」で検査対象を展開します。
3. 「**すべてのディスク装置 (All Disk Units)**」を右クリックし、「**グラフィカル・ビュー (Graphical View)**」を選択します。

以下は、System i ナビゲーターのグラフィカル・ビューの例です。「以下で表示 (View by)」メニューには、ディスク装置表示用のオプションがいくつかリストされています。



ディスク・プールを使用可能にする

独立ディスク・プールのディスク装置にアクセスするには、ディスク・プールを使用可能 (オンに変更) する必要があります。

独立ディスク・プールのディスク装置および対応するデータベースのオブジェクトにアクセスするには、ディスク・プールを使用可能 (オンに変更) する必要があります。リモート・ミラーリングを使用している場合は、ディスク・プールの実動コピーを使用可能にしなければなりません。実動コピーが切り離されている場合は、ミラー・コピーの作成のみが可能です。リモート・ミラーリングされたディスク・プールの場合、リモート・ミラーリングが中断状態でない限り、切り替え可能なハードウェア・グループを開始してからディスク・プールを使用可能にします。

複数システムがクラスター化された環境では、現行ノードまたはクラスター内の別のノードに対して、ディスク・プールを使用可能な状態にすることができます。独立ディスク・プールは、1 度に 1 つのノードに

対してのみ、オンにすることが可能です。別のノードから独立ディスク・プールにアクセスするときは、独立ディスク・プールをバックアップ・クラスター・ノードに切り替える必要があります。詳細は、『切り替えの実行』でデバイス CRG (System i ナビゲーターでは切り替え可能ハードウェアとして参照) のバックアップ・ノードへの切り替えについて参照してください。

注: 1 次または 2 次ディスク・プールを使用可能にすると、ディスク・プール・グループ内のすべてのディスク・プールも同時に使用可能になります。

ディスク・プールを使用可能にするか、別の独立ディスク・プール上でディスク構成の変更を行う場合、処理が停止しているように見えます。これは、別の装置記述アクティビティを実行中の場合にディスク・プールを使用可能にすると、ディスク構成の変更が待機するためです。

リモート・ミラーリングされたディスク・プールの使用可能処理の最初で障害が発生すると、次回の使用可能または再開時に完全同期を行わなければならない場合があります。

独立ディスク・プールを使用可能にするには、以下の手順を行います。

1. System i ナビゲーターで、「**ユーザー接続**」(またはアクティブな環境)を展開します。
2. 「**構成およびサービス**」→「**ハードウェア**」→「**ディスク装置**」で検査対象を展開します。
3. 「**ディスク・プール**」を展開します。
4. 使用不可になっているディスク・プールを右クリックして、「**使用可能化**」を選択します。同時に複数のディスク・プールを選択して使用可能にすることができます。
5. 表示されるダイアログ・ボックスから、「**使用可能化**」を選択して、ディスク・プールを使用可能な状態にします。

文字ベース・インターフェースの構成の変更 (VRYCFG) コマンドを使用して、ディスク・プールを使用可能にすることができます。

ステップが処理のどの段階にあるかを判断するには、ASP 状況の表示 (DSPASPSTS) コマンドを使用します。

サイト間ミラーリングの構成

サイト間ミラーリングとは、リモート・ミラーリング、メトロ・ミラー、グローバル・ミラーなど、さまざまな高可用性テクノロジーを総称したものです。これらの各テクノロジーには、構成に関連する具体的なタスクがあります。

リモート・ミラーリングの構成

リモート・ミラーリングは、サイト間ミラーリングの副次機能です。リモート・ミラーリングを使用して高可用性ソリューションを構成するには、実動システムとバックアップ・システムの間で、ミラーリング・セッションを構成する必要があります。

リモート・ミラーリングを構成するには、アクティブなクラスター、ノード、および CRG が必要になります。リモート・ミラーリングに使用する予定の独立ディスク・プールは、構成を完了するためにオフ (使用不可) に変更しておく必要があります。トピック「シナリオ: リモート・ミラーリングを使用するサイト間ミラーリング」では、リモート・ミラーリングに基づく高可用性ソリューションをセットアップするための、段階的な手順を説明しています。

IBM Systems Director Navigator for i5/OS

IBM Systems Director Navigator for i5/OS を使用してリモート・ミラーリングを構成するには、以下の手順を実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 実動 (ソース) コピーとして使用するディスク・プールを選択します。
6. 「アクションの選択」メニューから、「新規セッション」を選択します。
7. ウィザードの指示に従って、作業を完了します。

System i Navigator

System i Navigator を使用してリモート・ミラーリングを構成するには、以下の手順を実行します。

1. System i Navigator で、「ユーザー接続」 (または、アクティブな環境) を展開します。
2. 実動コピーとして使用するシステムを展開します。
3. 「構成およびサービス」 → 「ハードウェア」 → 「ディスク装置」 → 「ディスク・プール」を展開します。
4. 実動コピーとして使用するディスク・プールを右クリックして、「セッション」 → 「新規」と選択します。
5. ウィザードの指示に従って、作業を完了します。

関連概念

84 ページの『シナリオ: リモート・ミラーリングによる切り替えディスク』

このシナリオでは、3 つのノード・クラスターにおいてリモート・ミラーリングによる切り替えディスクを使用する、i5/OS 高可用性ソリューションについて説明します。このソリューションは、災害時回復と高可用性の両方を提供します。

メトロ・ミラー・セッションの構成

IBM System Storage のメトロ・ミラー・テクノロジーを使用した i5/OS 高可用性ソリューションの場合、System i マシンと、メトロ・ミラーが構成されている IBM System Storage の外部ストレージ・ユニットとの間に、セッションを構成する必要があります。i5/OS のメトロ・ミラー・セッションは、外部ストレージ・ユニットのミラーリングをセットアップするのではなく、i5/OS システムと外部ストレージ・ユニット上の既存のメトロ・ミラー構成との間での関係をセットアップします。

i5/OS 上でメトロ・ミラー・セッションを作成する前に、IBM System Storage の外部ストレージ・ユニット上で、メトロ・ミラーを構成しておく必要があります。IBM System Storage DS6000 でのメトロ・ミラーの使用について詳しくは、IBM System Storage DS6000 Information Center を参照してください。IBM System Storage DS8000 でのメトロ・ミラーの使用について詳しくは、IBM System Storage DS8000 Information Center を参照してください。

メトロ・ミラー・セッションを構成するには、以下の手順で行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 実動 (ソース) コピーとして使用するディスク・プールを選択します。

6. 「アクションの選択」メニューから、「新規セッション」を選択します。
7. ウィザードの指示に従って、作業を完了します。

関連情報

Add コピー説明の追加 (ADDASPCPYD) コマンド (英語)

ASP セッションの開始 (STRASPSSN) コマンド (英語)

グローバル・ミラー・セッションの構成

IBM System Storage のグローバル・ミラー・テクノロジーを使用する i5/OS 高可用性ソリューションの場合、System i マシンと、グローバル・ミラーが構成されている IBM System Storage の外部ストレージ・ユニットとの間で、セッションを構成する必要があります。i5/OS のグローバル・ミラー・セッションは、外部ストレージ・ユニットのミラーリングをセットアップするのではなく、i5/OS システムと外部ストレージ・ユニット上の既存のグローバル・ミラー構成との間での関係をセットアップします。

IBM System Storage グローバル・ミラー・テクノロジーでは、すべてのユーザーが 1 つのグローバル・ミラー接続を共有しなければなりません。i5/OS の高可用性グローバル・ミラーでは、所定の System Storage サーバー上でグローバル・ミラーを構成することができる System i 区画は 1 つだけです。他の System i 区画や、他のプラットフォームのサーバーは、同時にグローバル・ミラーを使用することはできません。グローバル・ミラー・セッションに複数のユーザーを追加すると、予想不能の結果が生じます。

i5/OS でグローバル・ミラー・セッションを作成する前に、IBM System Storage の外部ストレージ・ユニット上で、グローバル・ミラーを構成しておく必要があります。IBM System Storage DS6000 でのグローバル・ミラーの使用について詳しくは、IBM System Storage DS6000 Information Center を参照してください。IBM System Storage DS8000 でのグローバル・ミラーの使用について詳しくは、IBM System Storage DS8000 Information Center を参照してください。

グローバル・ミラーリングを構成するには、以下の手順で行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 実動 (ソース) コピーとして使用するディスク・プールを選択します。
6. 「アクションの選択」メニューから、「新規セッション」を選択します。
7. ウィザードの指示に従って、作業を完了します。

関連情報

Add コピー説明の追加 (ADDASPCPYD) コマンド (英語)

ASP セッションの開始 (STRASPSSN) コマンド (英語)

高可用性の管理

i5/OS 高可用性ソリューションの構成後、高可用性に関連するいくつかのインターフェースを使用してソリューションを管理することができます。

シナリオ: 高可用性ソリューションの管理

高可用性ソリューションのシステム・オペレーターまたは管理者にとって、バックアップやシステム保守などの一般的なタスクを高可用性環境で実行することは不可欠です。

以下のシナリオでは、バックアップやアップグレードなどの一般的なシステム・タスクに加えて、クラスター区画やフェイルオーバーなどの高可用性イベントを実行する上での手順を説明します。各シナリオでは、モデル環境が既に選択されています。各シナリオでの説明は、特定の高可用性ソリューションに対応したものになっていますが、あくまでも一例として使用するようになっています。

シナリオ: 高可用性環境でのバックアップの実行

ご使用の高可用性ソリューションおよびバックアップ戦略によって、データのバックアップ方法は異なります。しかし、高可用性環境でシステムのバックアップ操作を実行する場合に、共通して使える一連のタスクがあります。

一部の高可用性ソリューションでは、バックアップ・システムに保管されたデータの 2 次コピーから、リモート・バックアップを実行できる場合があります。リモート・バックアップを使用することで、2 次システムでバックアップを行いながら、実動システムを操作可能にしておくことができます。バックアップ・システムでバックアップがリモートに実行される 2 つの高可用性ソリューションの例を、各シナリオで説明します。

最初のシナリオでは、リモート・ミラーリング技術を使用する高可用性ソリューションで、リモート・バックアップを実行します。2 番目のシナリオでは、メトロ・ミラー、グローバル・ミラーなどといった、IBM System Storage ソリューションを使用する高可用性環境で、FlashCopy を使用する方法を紹介します。

シナリオ: リモート・ミラーリング環境でのバックアップの実行:

このシナリオでは、リモート・ミラーリングを使用する i5/OS 高可用性ソリューションで、リモート・バックアップを実行する際に必要なタスクの概要を説明します。

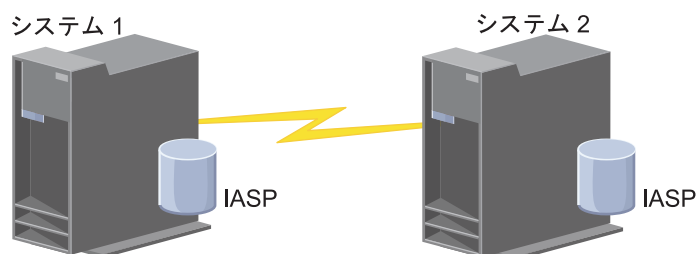
概説

この例では、リモート・ミラーリング・テクノロジーに基づく高可用性ソリューションで使用される独立ディスク・プールに保管されたデータのバックアップを、システム管理者が実行したいと考えています。バックアップを実行するために、実動システムをオフラインにしてしまうと、システムに影響が出てしまうため、管理者としては避けたいところです。そこで管理者は、ミラーリングされたコピーを一時的に切り離してから、リモート・ロケーションの独立ディスク・プールにあるデータの 2 次コピーからバックアップを実行するよう、計画を立てます。

注: 基本的には、ミラーリングされたコピーを切り離すと、コピーが実動に再接続されるまでに、リモート・ミラーリングが終了します。切り離されている間は、高可用性および災害時回復は使用できなくなります。この処理中に実動システムで停止が発生した場合、データが失われてしまうことがあります。

詳細

次の図は、この環境を示しています。



構成ステップ

1. 独立ディスク・プールの静止
2. 169 ページの『ミラー・コピーの切り離し』
3. ディスク・プールを使用可能にする
4. 独立ディスク・プールのバックアップ
5. 166 ページの『独立ディスク・プールの再開』
6. 170 ページの『ミラー・コピーの再接続』

シナリオ: FlashCopy の実行:

この例では、バックアップ・サイトの外部ストレージ・ユニットに保管されたデータのリモート・コピーから、管理者がバックアップを実行します。IBM Storage Solutions で利用できる FlashCopy 機能を管理者が使用することで、バックアップの所要時間が大幅に削減されます。

概説

この例では、IBM System Storage の外部ストレージ・ユニットに保管されているバックアップ・データを、システム管理者が実行したいと考えています。バックアップを実行するために、実動システムをオフラインにしてしまうと、システムに影響が出てしまうため、管理者としては避けたいところです。代わりに FlashCopy を実行して、データのポイント・イン・タイム取り込みを行うよう、管理者が計画します。このデータを基に、管理者は外部メディアにデータをバックアップします。FlashCopy はわずか数秒で完了するため、バックアップ処理全体の時間を削減できます。

この例では、バックアップ操作用に FlashCopy を使用していますが、実際にはさまざまな目的で FlashCopy を使用することができます。例えば、FlashCopy をデータウェアハウジングに使用して、実動システム上の照会ワークロードを減らしたり、実動データの複製に使用して、テスト環境を作成したりすることができます。

構成ステップ

1. 166 ページの『独立ディスク・プールの静止』
2. 175 ページの『FlashCopy セッションの構成』
3. IBM System Storage の外部ストレージ・ユニットで、FlashCopy を実行します。IBM System Storage DS6000 での FlashCopy の使用について詳しくは、IBM System Storage DS6000 Information Center を参照してください。IBM System Storage DS8000 での FlashCopy の使用について詳しくは、IBM System Storage DS8000 Information Center を参照してください。
4. 166 ページの『独立ディスク・プールの再開』
5. ディスク・プールを使用可能にする
6. 独立ディスク・プールのバックアップ

シナリオ: 高可用性環境でのオペレーティング・システムのアップグレード

この例では、システム管理者が高可用性ソリューションにある 2 つの i5/OS システムのオペレーティング・システムを、リモート・ミラーリングに基づいてアップグレードしています。

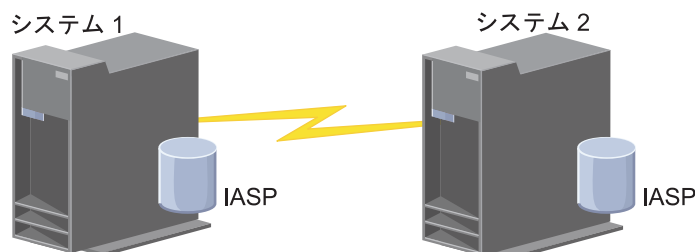
概説

システム管理者は、高可用性環境で 2 つのシステムに対してオペレーティング・システムをアップグレードする必要があります。この例では、システム 1 および システム 2 の、2 つのノードがあります。システム 1 は実動コピーで、システム 2 はミラー・コピーです。いずれのシステムも i5/OS V5R4 です。独

立ディスク・プールはオンラインであり、リモート・ミラーリングはアクティブ、そしてシステムは同期化されています。システム管理者はこれらのシステムを両方とも i5/OS V6R1 にアップグレードしたいと考えています。

詳細

以下の図形は、環境を図解しています。



構成ステップ

1. ミラー・コピーのデタッチ (システム 2)。
2. CRG の終了 (システム 2)。
3. ノードの停止 (システム 2)。
4. システム 2 を新規リリースにアップグレードします。詳しくは、『i5/OS および関連ソフトウェアのアップグレードまたは取り替え』を参照してください。
5. IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム (5761-HAS) のインストール。
6. ディスク・プールを使用できるようにして、システム 2 でアプリケーションをテストします。アプリケーションをテストすることにより、それらが新規リリース内で期待どおりに作動することを確認できます。アプリケーション・テストが完了したら、次の残りのステップを完了して、アップグレードを終了できます。
7. デタッチされたミラー・コピー (システム 2) でのディスク・プールを使用不可にします。
8. ミラー・コピーの再接続を実行します。これにより、ミラー・データの再同期が開始されます。再同期化が完了したら、アップグレード処理を続けることができます。
9. 101 ページの『切り替えの実行』。これにより、ミラー・コピー (システム 2) が新しい実動コピーになり、実動コピー (システム 1) が新しいミラー・コピーになります。

注: V6R1 から V5R4 へのリモート・ミラーリングは実行できないので、リモート・ミラーリングは中断します。V5R4 から V6R1 へのリモート・ミラーリングを問題なく実行できます。このシナリオでは、リモート・ミラーリングは切り替えが完了した後で中断します。アップグレード残存処理を実行中の際、データは保護されません。これは、有効なバックアップ・システムがない状態にあるからです。

10. CRG の終了 (システム 1)。
11. ノードの停止 (システム 1)。
12. システム 1 を新しいリリースにアップグレードします。詳しくは、『i5/OS および関連ソフトウェアのアップグレードまたは取り替え』を参照してください。
13. IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム (5761-HAS) のインストール。
14. ノードの開始 (システム 1)。

15. CRG の開始 (システム 1)。
16. ミラーリングの再開
17. 切り替えの実行。これで、現在のミラー・コピー (システム 1) が実動コピーに切り替えられ、実動コピー (システム 2) がミラー・コピーになります。これは、アップグレード前のオリジナルの構成です。

例: オペレーティング・システムのアップグレード:

高可用性環境では、オペレーティング・システムのアップグレードに先立って、特定のアクションを行う必要があります。

以下の例は、クラスター環境でアップグレードする際に行うべき必要なアクションを判断するのに役立ちます。アップグレードやアクションを行う前に、クラスターのクラスター・バージョンを最初に調べる必要があります。

例 1: アップグレードするノードは i5/OS V5R4。クラスター内のその他のすべてのノードは i5/OS V5R4 以降。現行クラスター・バージョンは 5。

アクション: ノードを i5/OS V6R1 にアップグレードします。ノードのアップグレード後に、アップグレードされたノード上でクラスタリングを始動します。

例 2: アップグレードされるノードは i5/OS V5R4。クラスター内のその他のすべてのノードは i5/OS V5R4。現行クラスター・バージョンは 4。

アクション: 現行クラスター・バージョンを 5 にアップグレードします。ノードは i5/OS にアップグレードします。アップグレードしたノード上でクラスタリングを始動します。

例 3: アップグレードするノードは i5/OS V5R3。クラスター内のその他のすべてのノードは i5/OS V5R4。現行クラスター・バージョンは 4。

アクション: i5/OS V6R1 にアップグレード中のノードをクラスターから削除してからアップグレードします。現行クラスター・バージョンを 5 に変更します。ノードを i5/OS V6R1 にアップグレードし、クラスターに追加して戻します。

例 4: アップグレードするノードは i5/OS V5R4。現在クラスター内にあるのは i5/OS V5R3 と i5/OS V5R4 のノードのみ。現行クラスター・バージョンは 4。i5/OS V5R4 ノードの i5/OS V6R1 へのアップグレードは、ノードが i5/OS V5R3 のままよりも重要度は低い。

アクション:

1. クラスターからアップグレード中のノードの除去を行います。
2. ノードを i5/OS V6R1 にアップグレードを行います。
3. 残りの i5/OS V5R3 ノードを最低でも i5/OS V5R4 にアップグレードを行います。
4. 『クラスター・バージョンの変更』にあるように、クラスター・バージョンを 5 にします。
5. アップグレードしたノードをクラスターに追加して戻します。

シナリオ 5: アップグレードするノードは i5/OS V5R4。現在クラスター内にあるのは i5/OS V5R3 と i5/OS V5R4 のノードのみ。現行クラスター・バージョンは 4。i5/OS V5R4 ノードの i5/OS V6R1 へのアップグレードは、ノードが i5/OS V5R3 のままよりも重要度は高い。

アクション:

1. i5/OS V5R3 ノードをクラスターからすべて除去します。
2. 『クラスター・バージョンの変更』にあるように、クラスター・バージョンを 5 にします。
3. ノードを i5/OS V6R1 にアップグレードを行います。
4. アップグレードしたノードを開始します。

5. 残りの i5/OS V5R3 ノードが i5/OS V5R4 にアップグレードされると、クラスターに追加して戻されます。

シナリオ 6: アップグレードするノードは i5/OS V5R3。クラスター内のノードのうち、他の少なくとも 1 つが i5/OS V5R3。現行クラスター・バージョンは 3 以下。

アクション: すべてのノードを i5/OS V5R4 にアップグレードします。クラスター・バージョンを 4 に変更します。すべてのノードを i5/OSV6R1 にアップグレードします。

以下の表に、クラスター環境でアップグレードを行う際に必要なアクションを示します。

表7. ノードの i5/OS V6R1 へのアップグレード

アップグレードするノードの現行リリース	現行クラスター・バージョン	アクション
V5R4	5	<ol style="list-style-type: none"> 1. ノードを i5/OS V6R1 にアップグレードを行います。 2. アップグレードしたノードを開始します。
V5R4	4	<ol style="list-style-type: none"> 1. 『クラスター・バージョンの変更』にあるように、クラスター・バージョンを 5 にします。 2. ノードを i5/OS V6R1 にアップグレードを行います。 3. アップグレードしたノードを開始します。 <p>注: クラスター内のその他のノードが i5/OS V5R3 の場合、シナリオ 4 および 5 で詳細を確認してください。</p>
V5R3	4 以下	<p>オプション A</p> <ol style="list-style-type: none"> 1. クラスターからアップグレード中のノードの除去を行います。 2. 『クラスター・バージョンの変更』にあるように、クラスター・バージョンを 5 にします。 3. ノードを V6R1 にアップグレードを行います。 4. ノードを追加してクラスターに戻します。 <p>オプション B</p> <ol style="list-style-type: none"> 1. すべてのノードで V5R4 にアップグレードを行います。 2. 『クラスター・バージョンの変更』にあるように、クラスター・バージョンを 5 にします。 3. すべてのノードで V6R1 にアップグレードを行います。

シナリオ: 装置を高可用性にする

独立ディスク・プールに加えて、サポートされている他の装置用にも高可用性を提供することができます。この状況において、高可用性アドミニストレーターはイーサネット回線へ高可用性を提供したいと考えます。

概説

システム管理者は高可用性ソリューション内で使用されているイーサネット回線の高可用性を提供したいと考えます。現在の構成では、切り替えディスク・テクノロジーを使用する 2 つのシステムによって、計画された停止に対する高可用性を提供しています。このソリューションはまた、クラスター管理可能ドメインを使用して、高可用性ソリューションの作動環境への変更を管理し、同期化します。この例では、これらのステップを終了する前に、すべての高可用性構成およびイーサネット構成が正常に完了しているものと想定しています。また、高可用性の現在の状態はアクティブで、すべてのモニター対象リソースが環境内で整合しているとも想定しています。この例は、イーサネット回線に対する高可用性を構成する際のステップを提供します。

構成ステップ

1. 139 ページの『切り替え可能装置の作成』
2. 114 ページの『モニター対象リソース項目の追加』
3. 150 ページの『モニターする属性の選択』

クラスターの管理

クラスター・リソース・サービスのグラフィカル・インターフェースを使用すると、i5/OS 高可用性ソリューションの基礎となるクラスター・テクノロジーに関連した、さまざまなタスクを実行することができます。これらのタスクは、クラスターの管理および保守に役立ちます。

クラスターを構成した後に、以下のような変更をクラスターに加えることができます。

クラスター・タスク

- クラスターへのノードの追加
- クラスターからのノードの除去
- クラスター・ノードの開始
- クラスター・ノードの終了
- 最新レベルへのクラスターのクラスター・バージョンの調整
- クラスターの削除
- クラスター・ノードの変更

クラスター・リソース・グループ・タスク

- 新規クラスター・リソース・グループの作成
- 既存クラスター・リソース・グループの削除
- クラスター・リソース・グループの開始
- クラスター・リソース・グループへのノードの追加
- クラスター・リソース・グループからのノードの除去
- クラスター・リソース・グループの終了
- クラスター・リソース・グループのリカバリー・ドメインの変更

- 切り替えの実行
- デバイス・ドメインへのノードの追加
- デバイス・ドメインからのノードの除去

クラスター管理可能ドメインのタスク

- クラスター管理可能ドメインの作成
- モニター対象リソースの追加
- クラスター管理可能ドメインの削除

クラスターのクラスター・バージョンの調整

クラスター・バージョンは、クラスター内のすべてのノードがアクティブに相互通信するレベルを定義します。

1 つのクラスターの中に、複数のリリース・レベルのシステムを組み込み、使用可能な通信プロトコルのレベルを判別することによって、完全な相互運用を実現するための技法が、このクラスター・バージョン設定です。

クラスター・バージョンを変更するには、クラスター内のすべてのノードは同じ潜在バージョンでなければなりません。そうすると、クラスター・バージョンを潜在バージョンと一致するように変更することができます。これにより、新機能が使用できるようになります。バージョンは 1 つずつしか増やすことができません。これを減らすには、クラスターを削除して、古いバージョンで再作成しなくてはなりません。現行のクラスター・バージョンは、クラスター内で定義されている最初のノードを基準にして初期設定されます。それ以降クラスターに追加されるノードは、現行のクラスター・バージョンと同等か、またはその次のレベルのバージョンにする必要があります。そうしないと、クラスターには追加できません。

- | ノードを新規リリースにアップグレードする場合は、ノードのクラスター・バージョンが適切であることを
- | 確認する必要があります。クラスターは、1 つ違いのバージョンしかサポートしません。クラスターにある
- | すべてのノードが同じリリースである場合、クラスター・バージョンを変更する前に、ノードを新規リリー
- | スにアップグレードする必要があります。これによって、新規リリースに関連した機能がすべて使用可能に
- | なります。新規リリースへのアップグレードを行うためのアクションについては、124 ページの『シ
- | ナリオ: 高可用性環境でのオペレーティング・システムのアップグレード』のトピックを参照してくださ
- | い。

ノードのクラスター・バージョンの検証および変更を行うには、以下の指示に従ってください。

- | 1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
- | 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- | 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選
- | 択します。
- | 4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・プロパティーの表示**」タスクを選択し
- | ます。
- | 5. 「**クラスター・プロパティー**」ページで、「**一般**」タブをクリックします。
- | 6. クラスター・バージョンの設定を検証するか、またはクラスター・バージョンを希望する設定に変更し
- | ます。

関連概念

クラスター・バージョン

関連情報

クラスター・バージョンの変更 (CHGCLUVER) コマンド

クラスター・バージョン調整 (QcstAdjustClusterVersion) API (英語)

クラスターの削除

クラスターを削除すると、アクティブなすべてのクラスター・ノード上でクラスター・リソース・サービスが終了し、クラスターから除去されます。

- | アクティブ・ノードが少なくとも 1 つ存在しないと、クラスターを削除することはできません。切り替え
- | ディスクまたはその他の切り替え可能装置がクラスターにある場合、クラスターを削除する前に、まず各ノ
- | ードをデバイス・ドメインから除去する必要があります。これを行わないと、別のクラスターにディスクを
- | 戻せない場合があります。

クラスターを削除するには、以下の手順で行います。

- | 1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
- | 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- | 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選
- | 択します。
- | 4. 「**クラスター・リソース・サービス**」ページで、「**クラスターの削除**」をクリックします。
- | 5. 「**クラスターの削除**」確認ウィンドウが表示されます。「はい」を選択すると、クラスターが削除され
- | ます。クラスターを削除すると、「**クラスター・リソース・サービス**」ページが変更されて、「**新規ク**
- | **ラスタ**」タスクが表示されます。

関連タスク

136 ページの『デバイス・ドメインからのノードの除去』

デバイス・ドメインとは、装置リソースを共有するクラスター内のノードのサブセットです。

関連情報

クラスターの削除 (DLTCLU) コマンド

クラスターの削除 (QcstDeleteCluster) API

クラスター構成の表示

- | クラスタ構成に関する情報を提供する詳細レポートを、表示することができます。クラスタ構成レポー
- | トは、クラスタ、ノード・メンバーシップ・リスト、構成パラメーター、調整パラメーター、クラスタ
- | 内の各クラスター・リソース・グループなどに関する詳細情報を提供します。

- | クラスタ構成を表示するには、以下の手順を実行します。

- | 1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
- | 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- | 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選
- | 択します。
- | 4. 「**クラスター・リソース・サービス**」ページで、「**構成情報の表示**」タスクを選択します。「クラスタ
- | 構成およびプロパティ
- | ー」ページが表示されます。このページをファイルとして保存することもでき
- | ますし、印刷することもできます。

関連情報

- | クラスタ情報の表示 (DSPCLUINF) コマンド

クラスター構成の保管および復元

システムにクラスタリングを使用する場合でも、データを保護するためのバックアップ戦略および回復戦略を計画することは依然として重要です。

バックアップ戦略の一環としてクラスタリングの使用を計画し、バックアップ時に一方のシステムがダウンしたとき、もう一方のシステムが稼働中であるようにする場合は、クラスター内に最低でも 3 つ以上のシステムを含めるようお勧めします。クラスター内に 3 つのシステムがあれば、常に 1 つのシステムは切り替えが可能で、いずれかのシステムで障害が発生した場合に安心です。

クラスター・リソース・グループの保管および復元

クラスターがアクティブかどうかに関係なく、クラスター・リソース・グループはいつでも保管できます。クラスター・リソース・グループの復元については、以下の制限があてはまります。

- 稼働しているクラスターに認識されていないクラスター・リソース・グループは、復元できません。
- ノードがクラスターに構成されていない場合、クラスター・リソース・グループは復元できません。

アクティブなクラスターにクラスター・リソース・グループが認識されておらず、ノードがクラスター・リソース・グループのリカバリー・ドメイン内に含まれており、かつそのクラスター名がクラスター・リソース・グループ内のクラスター名と一致している場合、そのクラスター・リソース・グループは復元できます。クラスターが構成されているものの、そのノードではアクティブではなく、しかしそのノードがそのクラスター・リソース・グループのリカバリー・ドメイン内に含まれている場合には、そのクラスター・リソース・グループを復元できます。

災害のための準備

災害が発生した際には、クラスターの再構成が必要になる場合があります。そのようなシナリオに備えるために、クラスター構成情報を保管して、その情報のハードコピー印刷出力を取っておくことをお勧めします。

1. クラスター構成に変更を加えた後、構成の保管 (SAVCFG) コマンドまたはシステム保管 (SAVSYS) コマンドを使用して、復元された内部クラスター情報を現行のものとし、クラスター内のその他のノードと整合させます。SAVCFG または SAVSYS 操作の実行について詳しくは、『構成情報を保管する』を参照してください。
2. クラスター構成情報を変更するたびに、そのコピーをプリントします。クラスター情報の表示 (DSPCLUINF) コマンドを使用して、クラスター構成をプリントできます。バックアップ・テープと一緒にコピーを保持しておいてください。災害が発生した際に、全クラスターの再構成が必要になる場合があります。

関連情報

構成情報を保管する

構成の保管 (SAVCFG) コマンド

システム保管 (SAVSYS) コマンド

クラスター情報の表示 (DSPCLUINF) コマンド

1 クラスター状況のモニター

1 クラスター・リソース・サービスのグラフィカル・インターフェースは、クラスター状況をモニターし、高可用性ソリューションに関係するノードに不整合が発生すると警告メッセージを表示します。

1 クラスター・リソース・サービスのグラフィカル・インターフェースは、クラスターに不整合が生じると「ノード」ページに警告メッセージ HAI0001W を表示します。不整合に関するメッセージは、このノード

から取得される情報が、クラスター内の他のアクティブ・ノードと不整合を生じている可能性があることを意味します。クラスター内で非アクティブになったノードには、不整合が生じます。

整合された情報を取得するには、クラスター内のアクティブ・ノードからクラスター情報にアクセスするか、またはこのノードを開始して要求を再試行してください。

クラスター状況をモニターするには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。

2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。

3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。

4. 「ノード」ページには、ノードに不整合が生じた際に HAI0001W が表示されます。ローカル・クラスター・ノードがアクティブではありません。ローカル・ノードが開始されるまでは、クラスター情報は正確ではない可能性があります。

関連タスク

103 ページの『ノードの開始』

クラスター・ノードを開始すると、i5/OS 高可用性環境のノードでクラスタリングおよびクラスター・リソース・サービスが開始されます。

関連情報

クラスター情報の表示 (DSPCLUINF) コマンド

CRG 情報の表示 (DSPCRGINF) コマンド

クラスター情報のリスト (QcstListClusterInfo) API (英語)

装置ドメイン情報リスト (QcstListDeviceDomainInfo) API (英語)

クラスター資源サービス情報検索 (QcstRetrieveCRSInfo) API (英語)

クラスター情報の検索 (QcstRetrieveClusterInfo) API (英語)

クラスター資源グループのリスト (QcstListClusterResourceGroups) API (英語)

クラスター資源グループ情報検索 (QcstListClusterResourceGroupInf) API (英語)

メッセージ・キューの指定

クラスター・メッセージ・キューまたはフェイルオーバー・メッセージ・キューのいずれかを指定することができます。これらのメッセージ・キューは、i5/OS 高可用性環境で障害の原因を判別するのに役立ちます。

クラスター・メッセージ・キューは、クラスター・レベルのメッセージに使用され、特定のノードにフェイルオーバーするすべてのクラスター・リソース・グループ (CRG) を制御するメッセージを 1 つ提供します。フェイルオーバー・メッセージ・キューは、CRG レベル・メッセージに使用され、フェイルオーバーする各 CRG に対してメッセージを 1 つ提供します。

クラスター・メッセージ・キューの指定

注: クラスターの作成ウィザードを実行してメッセージ・キューを指定することにより、クラスター・メッセージ・キューを使用するようにクラスターを構成することもできます。

クラスター・メッセージ・キューを指定するには、次の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。

2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。

3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・プロパティの表示**」をクリックします。
5. 「**クラスター・プロパティ**」ページで、「**クラスター・メッセージ待ち行列**」をクリックします。
6. 以下の情報を指定して、**クラスター・メッセージ・キュー**を作成します。
 - 「**名前**」フィールドで、**クラスター・レベル**または**ノード・レベル**で**フェイルオーバー**を処理する**メッセージ**を受け取るための、**メッセージ・キュー**の名前を指定します。ノード・レベルの**フェイルオーバー**の場合、同じ**新規プライマリー・ノード**ですべての**クラスター・リソース・グループ**の**フェイルオーバー**を制御する**メッセージ**が 1 つ送信されます。**クラスター・リソース・グループ**が個々に**フェイルオーバー**する場合、その**クラスター・リソース・グループ**の**フェイルオーバー**を制御する**メッセージ**が 1 つ送信されます。この**メッセージ**は、**新規プライマリー・ノード**で送信されます。このフィールドが設定されている場合、指定された**メッセージ・キュー**は、開始時に**クラスター**内のすべての**ノード**上に存在していなければなりません。**メッセージ・キュー**を**独立ディスク・プール**内に置くことはできません。
 - 「**ライブラリー**」フィールドで、**フェイルオーバー・メッセージ**を受け取る**メッセージ・キュー**が含まれる、**ライブラリー**の名前を指定します。
*CURLIB、*QTEMP、*LIBL、*USRLIBL、*ALL、*ALLUSR などを**ライブラリー**名にすることはできません。
 - 「**フェイルオーバー待ち時間**」フィールドで、「**待機しない not wait**」または「**無期限に待機**」のいずれかを選択するか、**クラスター・メッセージ・キュー**で**フェイルオーバー・メッセージ**に対する**応答**を待つ**分数**を指定します。
 - 「**フェイルオーバー・デフォルト・アクション**」フィールドで、**フェイルオーバー・メッセージ**に対する**応答**が**フェイルオーバー応答時間**の値を超えたときに、**クラスター・リソース・サービス**が行う**アクション**を指定します。このフィールドは、「**フェイルオーバーの続行**」または「**フェイルオーバーの取り消し**」のいずれかに設定することができます。

フェイルオーバー・メッセージ・キューの指定

フェイルオーバー・メッセージ・キューを指定するには、次のようなステップを行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードで、システムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・リソース・グループの処理**」を選択し、**クラスター**内の**クラスター・リソース・グループ**のリストを表示します。
5. **クラスター・リソース・グループ**のリストから、**処理**を行う**クラスター・リソース・グループ**を選択します。
6. 「**クラスター・リソース・グループ**」ページで、「**アクションの選択**」メニューをクリックし、「**プロパティ**」を選択します。
7. 「**一般**」ページで、以下の値を指定して、**フェイルオーバー・メッセージ・キュー**を指定します。
 - 「**フェイルオーバー・メッセージ待ち行列**」フィールドで、この**クラスター・リソース・グループ**で**フェイルオーバー**が発生したときに**メッセージ**を受け取る、**メッセージ・キュー**の名前を指定します。このフィールドが設定されている場合、**出口プログラム**が完了した後に、指定された**メッセー**

- ジ・キューがリカバリー・ドメイン内のすべてのノード上に存在していなければなりません。フェイルオーバー・メッセージ・キューを独立ディスク・プール内に置くことはできません。
- 「ライブラリー」フィールドで、フェイルオーバー・メッセージを受け取るメッセージ・キューが含まれる、ライブラリーの名前を指定します。 *CURLIB、QTEMP、*LIBLなどをライブラリー名にすることはできません。
- 「フェイルオーバー待ち時間」フィールドで、フェイルオーバー・メッセージ・キューでフェイルオーバー・メッセージに対する応答を待つ分数を指定します。フェイルオーバー・メッセージに対する応答が指定されたフェイルオーバー待ち時間を超えた場合に、クラスター・リソース・サービスが行うアクションを指定することもできます。

クラスター構成解除チェックリスト

クラスターの構成解除を完全に行うには、さまざまなクラスター・コンポーネントを体系的に除去する必要があります。

表8. クラスターの独立ディスク・プールの構成解除チェックリスト

独立ディスク・プールの要件	
—	切り替えディスク・プールを使用している場合、クラスター・リソース・グループの構成解除の前に、SPCN所有者であるノードにタワーを切り替える必要があります。切り替えの開始 (QcstInitiateSwitchOver) API または クラスター資源グループ 1 次の変更 (CHGCRGPRI) コマンドを使用して、CRG を SPCN 所有者に移し戻すことができます。この手順が実行されない場合、このシステム専用としてタワーをマークすることはできなくなります。
—	独立ディスク・プール・グループのサブセットの除去、または切り替え可能装置内の最後の独立ディスク・プールの除去を行う予定の場合、まず CRG を終了する必要があります。クラスター・リソース・グループの終了 (ENDCRG) コマンドを使用します。
—	クラスターに属している独立ディスク・プールを削除したい場合には、先に装置クラスター・リソース・グループ (CRG) を削除することを強くお勧めします。詳しくは、139 ページの『CRG の削除』を参照してください。 また、CRG 装置項目の除去 (RMVCRGDEVE) コマンドを使って、独立ディスク・プールの構成オブジェクトを CRG から除去することもできます。
—	独立ディスク・プールの構成オブジェクトをクラスター切り替え可能装置から除去し終わってから、独立ディスク・プールを削除することができます。
—	以下のタスクを実行して、独立ディスク・プールの装置記述を削除します。 1. コマンド行インターフェースで WRKDEVD DEVD(*ASP) と入力し、Enter キーを押します。 2. ページ送りして、削除したい独立ディスク・プールの装置記述を見つけ出します。 3. 装置記述の名前のそばのオプション 4 (「削除」) を選択し、Enter キーを押します。

表9. クラスターのクラスター・リソース・グループの構成解除のチェックリスト

クラスター・リソース・グループの要件	
—	以下の手順のいずれかを実行して、クラスター・リソース・グループを削除します。 1. ノード上でクラスタリングがアクティブになっていない場合、コマンド行インターフェースに DLTCRG CRG(CRGNAME) と入力します。CRGNAME は、削除しようとしている CRG の名前です。Enter を押します。 2. ノード上でクラスタリングがアクティブになっている場合、コマンド行インターフェースに DLTCRGCLU CLUSTER(CLUSTERNAME) CRG(CRGNAME) と入力します。CLUSTERNAME は、クラスターの名前です。CRGNAME は、削除しようとしている CRG の名前です。Enter を押します。

ノードの管理

i5/OS 高可用性環境の一部であるシステム区画と論理区画のことを、ノードと呼びます。ノードに関連したさまざまな管理タスクを実行することができます。

ノード・プロパティの表示:

高可用性環境の一部として構成されているノードに関連付けられたプロパティについて、クラスター・リソース・サービスのグラフィカル・インターフェースを使用して表示および管理を行うことができます。

ノード・プロパティを表示するには、以下の作業を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・ノードの処理**」タスクを選択して、クラスター内のノードのリストを表示します。
5. 「**ノード**」タブで、「**アクションの選択**」メニューをクリックし、「**プロパティ**」を選択します。「**実行**」をクリックします。ノードのプロパティ・ページが表示されます。
 - 「**一般**」ページに、ノードの名前およびそのノードのシステム IP アドレスが表示されます。
 - 「**クラスタリング**」ページに、以下の情報が表示されます。
 - クラスター・インターフェースの IP アドレスは、クラスター内の他のノードと通信するために、クラスタリングで使用されます。
 - ノードの潜在的バージョンは、クラスター内のノードがアクティブに相互通信するバージョンとモディフィケーション・レベルを指定します。
 - 選択されたクラスターで構成されるデバイス・ドメインが表示されます。デバイス・ドメインをリスト内で選択すると、選択されたデバイス・ドメインに属するノードも表示されます。

ノードの停止:

ノードを停止または終了すると、そのノードのクラスタリングおよびクラスター・リソース・サービスが終了します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**ノード**」タブで、停止するノードを選択します。
5. 「**アクションの選択**」メニューをクリックして、「**停止**」を選択します。指定されたノードでクラスター・リソース・サービスが正常に停止されると、ノードの状況は「**停止済み**」に設定されます。

関連情報

クラスター・ノードの終了 (ENDCLUNOD) コマンド (英語)

クラスター・ノードの終了 (QcstEndClusterNode) API

ノードの除去:

- 1 ノードをアップグレードする場合、またはノードを i5/OS 高可用性環境に参加させる必要がなくなった場合には、クラスターからノードを除去しなくてはならない場合があります。

既存のクラスターからノードを除去するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・ノードの処理**」タスクを選択して、クラスター内のノードのリストを表示します。
5. 「**ノード**」ページで、「**アクションの選択**」メニューを選択し、「**除去**」を選択します。
6. 「**クラスター・ノードの除去**」確認ウィンドウで、「**はい**」をクリックします。

関連タスク

170 ページの『リモート・ミラーリングの構成解除』

特定のディスク・プールまたはディスク・プール・グループに対して、リモート・ミラーリングを使用する機能がなくなっただけの場合は、「**リモート・ミラーリングの構成解除 (Deconfigure Geographic Mirroring)**」を選択してください。リモート・ミラーリングを構成解除すると、システムはリモート・ミラーリングを停止し、ミラー・コピー・サイトのノード上にある、ディスク・プールのミラー・コピーを削除します。

関連情報

クラスター・ノード項目の除去 (RMVCLUNODE) コマンド

クラスター・ノード項目の除去 (QcstRemoveClusterNodeEntry) API (英語)

デバイス・ドメインからのノードの除去:

デバイス・ドメインとは、装置リソースを共有するクラスター内のノードのサブセットです。

重要:

デバイス・ドメインからノードを除去するときは、注意が必要です。デバイス・ドメインからノードを除去する場合、そしてそのノードがすべての独立ディスク・プールに対する現行の 1 次アクセス・ポイントになっている場合、ノードは除去されますがそれらの独立ディスク・プールは残りません。つまり、デバイス・ドメイン内の残りのノードから、これらの独立ディスク・プールにはアクセスできなくなるということです。

ノードをデバイス・ドメインから除去した後でも、同じデバイス・ドメイン内に既存のクラスター・ノードが 1 つ以上属している場合には、そのデバイス・ドメインにノードを再度追加することはできません。ノードをデバイス・ドメインに再度追加するには、以下の手順を実行します。

1. デバイス・ドメインに追加されるノードが現在所有している独立ディスク・プールを削除する。
2. IPL をノード上で実行して、システムを再始動する。
3. デバイス・ドメインにノードを追加する。
4. ステップ 1 で削除した独立ディスク・プールを再作成する。

デバイス・ドメインからノードを除去するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。

- 4. 「**クラスター・リソース・サービス**」 ページで、「**クラスター・ノードの処理**」タスクを選択して、クラスター内のノードのリストを表示します。
- 5. 「**ノード**」タブで「**アクションの選択**」メニューを選択し、「**プロパティ**」を選択します。「**実行**」をクリックします。「**ノード・プロパティ**」シートが表示されます。
- 6. 「**クラスター化**」タブで「**装置ドメイン**」フィールドからノード名を削除して、「**OK**」をクリックします。

関連タスク

130 ページの『**クラスターの削除**』

クラスターを削除すると、アクティブなすべてのクラスター・ノード上でクラスター・リソース・サービスが終了し、クラスターから除去されます。

関連情報

装置ドメイン項目の除去 (RMVDEVDMNE) コマンド

装置ドメイン項目の除去 (QcstRemoveDeviceDomainEntry) API (英語)

クラスター・リソース・グループ (CRG) の管理

クラスター・リソース・グループ (CRG) は、i5/OS 高可用性環境内で回復力のあるリソースを管理します。これらは、障害時におけるバックアップ・システムへのリソースの切り替えを定義および制御するクラスター・テクノロジーです。

CRG 状況の表示:

高可用性環境におけるクラスター・リソース・グループ (CRG) の状況をモニターすることができます。これらの状況メッセージを使用することで、CRG の変更内容を検証し、CRG における問題点を判別できます。

CRG 状況を表示するには、以下の手順を実行します。

- 1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
- 2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
- 3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
- 4. 「**クラスター・リソース・サービス**」 ページで、「**クラスター・リソース・グループの処理**」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
- 5. 「**クラスター・リソース・グループ**」 ページで、「**状況**」列に CRG の現在の状況を表示します。

想定される CRG の状況値は、以下のとおりです。

表 10. CRG の状況値

想定される値	説明
開始済み	CRG は現在開始されています。
停止済み	CRG は現在停止されています。
未確定	高可用性ソリューション内のこの CRG に関する情報は、正確ではない可能性があります。この状況が発生するのは、取り消しアクションによって CRG 出口プログラムが呼び出されて、正常に完了しなかった場合です。

表 10. CRG の状況値 (続き)

想定される値	説明
復元済み	CRG はノード上に復元され、クラスター内の他のノードにはコピーされていません。クラスタリングがノードで開始されると、CRG は他のノードと同期化され、状況が非アクティブに設定されます。
非アクティブ	CRG のクラスター・リソース・サービスが、ノード上でアクティブになっていません。ノードが失敗したか、ノードが終了したか、あるいはそのノード上で CRG が実行されていないか、いずれかが考えられます。
削除	CRG をクラスターから削除しているところです。
変更	CRG を変更しているところです。この変更が正常に完了すると、CRG は前の状況にリセットされます。
停止	CRG を停止しているところです。
追加	CRG をクラスターに追加しているところです。
開始	CRG を開始しているところです。
切り替え中	CRG を他のノードに切り替えているところです。
ノードの追加中	新規ノードをクラスターに追加しているところです。ノードが正常に追加されると、CRG は前の状況にリセットされます。
ノードの除去中	ノードが CRG から除去されているところです。ノードが正常に除去されると、CRG は前の状況にリセットされます。
ノード状況の変更中	CRG のリカバリー・ドメイン内のノードの状況が、現在変更されているところです。

CRG の停止:

クラスター・リソース・グループ (CRG) は、i5/OS 高可用性環境内の回復力のあるリソースを管理します。これらのクラスター・テクノロジーによって、停止時における回復力のあるリソースのバックアップ・システムへの切り替えの定義および制御を行います。

高可用性環境で自動フェイルオーバー機能を終了するために、CRG を停止することができます。例えば、CRG で定義されるシステムのいずれかで IPL を実行している場合が、これにあたります。

CRG を停止するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログインします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで、「クラスター・リソース・グループの処理」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、停止する CRG を選択します。
6. 「アクションの選択」メニューから「停止」を選択して、「実行」をクリックします。

関連情報

クラスター資源グループの終了 (ENDCRG) コマンド

| クラスター資源グループの終了 (QcstEndClusterResourceGroup) API (英語)

CRG の削除:

クラスター・リソース・サービスのインターフェースを使用して、クラスター・リソース・グループを削除できます。

CRG を削除するには、以下の手順を実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで、「クラスター・リソース・グループの処理」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、削除する CRG を選択します。
6. 「アクションの選択」メニューから「削除」を選択し、「実行」をクリックします。
7. 「クラスター・リソース・グループの削除」確認ウィンドウで、「はい」を選択します。

関連情報

クラスター資源グループをクラスターから削除 (DLTCRGCLU) コマンド (英語)

クラスター資源グループの削除 (QcstDeleteClusterResourceGroup) API (英語)

| 切り替え可能装置の作成:

| 独立ディスク・プール装置のほかに、他の複数のデバイスが高可用性のためにサポートされています。イー
| サネット回線や光ディスク・デバイス、およびネットワーク・サーバーなどの装置が、今では高可用性の一
| 部となることが可能です。

| 装置クラスター・リソース・グループには切り替え可能な装置のリストが含まれています。リストの各装置
| は、切り替え可能な独立ディスク・プールや他のタイプの切り替え可能装置 (磁気テープ装置、回線記述、
| コントローラー、ネットワーク・サーバーなど) を識別します。障害が起こると、装置の集合全体が、バッ
| クアップ・ノードに切り替わります。切り替え処理やフェイルオーバー処理の間に、装置をオンに変更する
| こともできます。

| 切り替え可能装置を作成するには、以下のステップを実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで、「クラスター・リソース・グループの処理」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、コンテキスト・アイコン (既存の切り替え可能装置を追加する対象の装置クラスター・リソース・グループの隣) をクリックして、コンテキスト・メニューから「既存の装置の追加」を選択します。
6. 「切り替え可能装置の追加」リストで、「追加」をクリックします。

- 7. 「切り替え可能装置の追加」ウィンドウで、切り替え可能装置の構成オブジェクト・タイプおよびオブジェクト名を入力します。「OK」をクリックして、新規切り替え可能装置をリストに追加します。例えば、切り替え可能なイーサネット回線を追加していた場合は、リストの「イーサネット」を選択します。
- 8. リスト・ウィンドウで「OK」をクリックし、新しい装置を装置クラスター・リソース・グループに追加します。

CRG のリカバリー・ドメインの変更:

リカバリー・ドメインは、クラスター・リソース・グループ (CRG) で定義されたノードのサブセット用のリカバリー・アクションを制御します。

- デバイス・クラスター・リソース・グループ、アプリケーション・クラスター・リソース・グループ、またはデータ・クラスター・リソース・グループのリカバリー・ドメインを変更するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで、「クラスター・リソース・グループの処理」を選択し、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」ページで、変更する CRG を選択します。
6. 「アクションの選択」メニューで「プロパティ」を選択して、「実行」をクリックします。
7. 「リカバリー・ドメイン」ページをクリックして、リカバリー・ドメインの既存の値を変更します。このページで、クラスターのリカバリー・ドメイン内にあるノードの役割の変更、リカバリー・ドメインへのノードの追加、およびリカバリー・ドメインからのノードの削除を行うことができます。装置クラスター・リソース・グループの場合は、リカバリー・ドメインにあるノードのサイト名とデータ・ポート IP アドレスの変更も行えます。

関連情報

CRG ノード項目の追加 (ADDCRGNODE) コマンド

クラスター資源グループの変更 (CHGCRG) コマンド

CRG ノード項目の除去 (RMVCRGNODE) コマンド

リカバリー・ドメインへのノードの追加 (QcstAddNodeToRcvyDomain) API (英語)

クラスター資源グループの変更 (QcstChangeClusterResourceGroup) API (英語)

リカバリー・ドメインからのノードの除去 (QcstRemoveNodeFromRcvyDomain) API (英語)

サイト名およびデータ・ポート IP アドレスの作成:

- リモート・ミラーリングを使用している場合、装置クラスター・リソース・グループのリカバリー・ドメイン・ノードで定義されているノードには、データ・ポート IP アドレスとサイト名が必要になります。

- サイト名は、装置クラスター・リソース・グループのリカバリー・ドメイン内のノードに関連しており、リモート・ミラーリングにのみ適用されます。高可用性のためにリモート・ミラーリング環境を構成している場合、異なるサイトの各ノードを異なるサイト名に割り当てる必要があります。

- リカバリー・ドメインでノードのデータ・ポート IP アドレスおよびサイト名を作成するには、以下の手順を実行します。

1. Web ブラウザーで、http://mysystem:2001 と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「クラスター・リソース・サービス」ページで、「**クラスター・リソース・グループの処理**」タスクをクリックして、クラスター内のクラスター・リソース・グループのリストを表示します。
5. 「クラスター・リソース・グループ」タブで、装置クラスター・リソース・グループの隣にあるコンテキスト・アイコンをクリックして、「**プロパティ**」を選択します。
6. 「リカバリー・ドメイン」ページで、「**編集**」を選択します。
7. 既存のデータ・ポート IP アドレスを使用するには、希望のアドレスをリストから選択して、「**OK**」をクリックします。データ・ポート IP アドレスを新たに追加するには、「**追加**」をクリックします。「データ・ポート IP アドレスの追加」ウィンドウで、IP アドレスを入力します。
8. 「**編集**」ウィンドウで、サイト名を指定します。

フェイルオーバー停止イベントの管理

通常、フェイルオーバーはノード停止によって起こりますが、その他の理由でも起こることがあります。システムの違いやユーザー処置によって、フェイルオーバー状態が発生する可能性があります。

ある問題によって単一のクラスター・リソース・グループが影響を受けると、その CRG だけでなく、他の CRG でもフェイルオーバーが発生する可能性があります。

- クラスター内で発生する停止には 4 つのカテゴリーがあります。これらのイベントにはノードが停止している真のフェイルオーバーもありますが、原因を判別して適切な対応を行うために調査が必要な場合があります。以下の表に、停止の各カテゴリー、各カテゴリーに該当する停止イベントの種類、および回復のために講じる必要がある適切な回復処置を示します。

カテゴリー 1 の停止: フェイルオーバーの原因となるノード停止

- ノード・レベルのフェイルオーバーが発生し、以下のことが起きます。
 - 各 CRG で、プライマリー・ノードに非アクティブのマークが付けられ、最後のバックアップ・ノードにされます。
 - 最初のバックアップだったノードは新しいプライマリー・ノードになります。
- フェイルオーバーは以下の順序で発生します。
 1. すべてのデバイスの CRG
 2. すべてのデータの CRG
 3. すべてのアプリケーションの CRG

注:

1. CRG のフェイルオーバーにおいて、アクティブなバックアップ・ノードがないことが検出される場合、その CRG の状況は未確定に設定され、CRG リカバリー・ドメインは変更されません。
2. クラスター・リソース・サービスのすべてに障害が発生した場合、クラスター・リソース・サービスによって管理されるリソース (CRG) にはフェイルオーバー・プロセスが実行されます。

表 11. カテゴリー 1 の停止: フェイルオーバーの原因となるノード停止

フェイルオーバー停止イベント
ENDTCP(時間制限付きの *IMMED または *CNTRLD) が発行された。

表 11. カテゴリー 1 の停止: フェイルオーバーの原因となるノード停止 (続き)

フェイルオーバー停止イベント	
	ENDSYS (*IMMED または *CNTRLD) が発行された。
	PWRDWN SYS(*IMMED または *CNTRLD) が発行された。
	システム上でクラスター・リソース・サービスがアクティブの場合に初期プログラム・ロード (IPL) ボタンが押される。
	クラスター・ノードの終了 (API またはコマンド) が CRG リカバリー・ドメイン内の 1 次ノードに呼び出される。
	クラスター・ノードの解除 (API またはコマンド) が CRG リカバリー・ドメイン内の 1 次ノードに呼び出される。
	区画の HMC 遅延電源遮断またはパネル・オプション 7 が発行された。
	ENDSBS QSYSWRK(*IMMED または *CNTRLD) が発行された。
	QCSTCTL ジョブの取り消しジョブ (時間制限付きの *IMMED または *CNTRLD) が発行された。
	QCSTCRGM ジョブの取り消しジョブ (時間制限付きの *IMMED または *CNTRLD) が発行された。

カテゴリー 2 の停止: 区画の原因となるノード停止

- | これらの停止によって以下のことが起こります。
- |
 - | • クラスター・メッセージングによる通信を行っていないノードの状況は、「区画」に設定されます。区画についての詳細は、『クラスター区画』を参照してください。
 - | • 区画のメンバーとしてプライマリー・ノードを持たないクラスター区画内の全ノードは、アクティブなクラスター・リソース・グループを終了します。

| 注:

- | 1. ノードで実際に障害が発生しても、単なる区画問題として検出され、障害の発生したノードがプライマリー・ノードであれば、そのノード上のデータおよびアプリケーション・サービスはすべて失われ、フェイルオーバーは自動的に開始されません。
- | 2. ノードに対して失敗と宣言するか、またはノード・バックアップを起動してそのノードに対して再びクラスタリングを開始する必要があります。詳細については、『区画化ノードを障害ノードに変更する』を参照してください。

表 12. カテゴリー 2 の停止: 区画の原因となるノード停止

フェイルオーバー停止イベント	
	CEC ハードウェア停止 (例えば、CPU) が発生。
	オペレーティング・システムのソフトウェアによるマシン・チェックが発生。
	HMC 即時パワーオフまたはパネル・オプション 8 が発行された。
	HMC 区画が再始動またはパネル・オプション 3 が発行された。
	CEC の停電が発生。

カテゴリー 3 の停止: フェイルオーバーの原因となる CRG 障害

- | CRG 障害がフェイルオーバーの原因の場合、以下のことが発生します。
- |
 - | • 単一の CRG のみが影響を受ける場合、フェイルオーバーは個々の CRG ベースで発生します。これは、CRG がお互いに独立しているためです。
 - | • いくつかのクラスター・リソース・ジョブが取り消されて複数の CRG が同時に影響を受けた場合、フェイルオーバーの調整は CRG 間で行われません。

- 1 • プライマリー・ノードは、個々の CRG 内で「非アクティブ」のマークが付けられ、最後のバックアップ・ノードにされます。
- 1 • 最初のバックアップ・ノードだったノードは新しいプライマリー・ノードになります。
- 1 • アクティブなバックアップ・ノードがない場合、CRG の状況は「未確定」に設定され、リカバリー・ドメインは変更されないままとなります。

表 13. カテゴリー 3 の停止: フェイルオーバーの原因となる CRG 障害

フェイルオーバー停止イベント
CRG ジョブに、そのジョブを異常終了させるソフトウェア・エラーがあります。
アプリケーション CRG のアプリケーション出口プログラムの障害です。

カテゴリー 4 の停止: 区画の原因となる通信停止

- 1 このカテゴリーはカテゴリー 2 に似ています。以下のイベントが発生します。
- 1 • クラスター・メッセージングによる通信を行っていないノードの状況は、「区画」に設定されます。区画についての詳細は、『クラスター区画』を参照してください。
- 1 • すべてのノードおよびノード上のクラスター・リソース・サービスは引き続き操作可能ですが、すべてのノードが相互に通信できるわけではありません。
- 1 • クラスターは区分されますが、CRG のプライマリー・ノード (複数も可) は引き続きサービスを提供します。
- 1 この区画状態に対しては、通常のリカバリーとして、クラスター区画の原因となった通信の問題を修理します。クラスターにより区画状態は解決されますが、その際、手動による操作は必要ありません。
- 1 注: CRG で新規プライマリー・ノードのフェイルオーバーを行う場合、ノードに失敗のマークが付けられる前に、古いプライマリー・ノードがリソースを使用していないことを確認してください。詳細については、『区画化ノードを障害ノードに変更する』を参照してください。

表 14. カテゴリー 4 の停止: 区画の原因となる通信停止

フェイルオーバー停止イベント
クラスター・ハートビート IP アドレス上の通信アダプター、回線、またはルーター障害が発生しています。
ENDTCPIFC はクラスター・ノード上のすべてのクラスター・ハートビート IP アドレスに影響を及ぼしています。

アクティブな CRG での停止

- 1 • CRG がアクティブのときに障害発生ノードがプライマリー・ノードではないと、以下のような結果になります。
 - 1 – フェイルオーバーによって、CRG のリカバリー・ドメイン内で障害が発生したりリカバリー・ドメイン・メンバーの状況が更新されます。
 - 1 – 障害発生ノードがバックアップ・ノードの場合、アクティブ・ノードがリストの先頭になるよう、バックアップ・ノードのリストはリオーダーされます。
- 1 • CRG がアクティブで、リカバリー・ドメイン・メンバーがプライマリー・ノードである場合、発生した障害の種類に応じて、以下のアクションがシステムで実行されます。
 - 1 – カテゴリー 1 の停止: フェイルオーバーの原因となるノード停止
 - 1 – カテゴリー 2 の停止: 区画の原因となるノード停止
 - 1 – カテゴリー 3 の停止: フェイルオーバーの原因となる CRG 障害
 - 1 – カテゴリー 4 の停止: 区画の原因となる通信停止

非アクティブな CRG での停止

- | CRG が停止していると、以下のことが起こります。
- | • クラスタ・リソース・グループのリカバリー・ドメインで障害が発生したノードのメンバーシップ状況は、非アクティブまたは区画状態に変更されます。
- | • ノードの役割は変更されず、バックアップ・ノードは自動的にリオーダーされません。
- | • クラスタ・リソース・グループの開始 (STRCRG) コマンドまたはクラスタ・リソース・グループ開始 (QcstStartClusterResourceGroup) API が呼び出されると、非アクティブの CRG 内のバックアップ・ノードの再配列が行われます。
- | 注: プライマリー・ノードがアクティブでない場合、クラスタ・リソース・グループ開始 API は失敗します。クラスタ・リソース・グループの変更 (CHGCRG) コマンドまたはクラスタ・リソース・グループ変更 (QcstChangeClusterResourceGroup) API を発行することによってアクティブ・ノードをプライマリー・ノードと指定してから、クラスタ・リソース・グループ開始 API をもう一度呼び出します。

クラスタ管理可能ドメインの管理

クラスタ管理可能ドメインを作成して、適切なモニター対象リソース項目 (MRE) を追加したら、クラスタ管理者は、管理可能ドメイン内のアクティビティをモニターして、モニター対象リソースの整合性が保たれるようにする必要があります。クラスタ・リソース・サービスのグラフィカル・インターフェースを使用すると、クラスタ管理可能ドメインの管理およびモニターを行うことができます。

このグラフィカル・インターフェースを使うと、各リソースの全体的な状況とともに MRE をリストすることができます。MRE を選択すると、詳細情報を表示できます。この情報には、MRE に関連付けられた各属性のグローバル値と、その属性がドメインと整合しているか、あるいは不整合であることを示す標識が含まれています。モニター対象リソース状況のグローバル状況に不整合が生じた場合、管理者は、リソースが不整合になった理由の判別、問題の解決、およびリソースの再同期を行うために必要な処置をとる必要があります。

1 つ以上のノード上で更新が失敗したことが原因でリソースに不整合が生じた場合は、MRE に保存された情報を利用して、障害の原因を特定することができます。障害が発生したノードでは、更新の失敗の原因に関するメッセージが MRE に記録されます。それ以外のノードでは、障害の発生を知らせる情報メッセージが、更新が失敗したノードのリストと一緒に内部に記録されます。これらのメッセージは、クラスタ・リソース・サービスのグラフィカル・インターフェースを通じて利用することもできますし、はモニター対象リソース情報の検索 (QfpadRtvMonitoredResourceInfo) API を呼び出して利用することもできます。障害メッセージは、対等 CRG ジョブのジョブ・ログに記録されます。

不整合の原因を特定したら、障害の発生場所であるノードで更新操作を行うか、または管理可能ドメインを終了してから再始動することで、リソースを再同期することができます。例えば、管理可能ドメインのあるノードでユーザー・プロファイルの UID を変更したが、指定された UID がすでに別のノードの別のユーザー・プロファイルによって使用されていたために、ユーザー・プロファイルの MRE で不整合が生じることがあります。この UID の値を再度変更して、管理可能ドメイン内の他のユーザー・プロファイルで使用されていないものにすると、すべてのノードのクラスタ管理可能ドメインでその変更が行われ、そのユーザー・プロファイル MRE のグローバル状況は「整合」に設定されます。ユーザー・プロファイル MRE を再同期するために、さらにアクションを行う必要はありません。

不整合のリソースを再同期するために、クラスタ管理可能ドメイン CRG を終了して、再始動しなければならない場合もあります。例えば、MRE が関連付けられているユーザー・プロファイルの UID を変更したが、管理可能ドメインの他のクラスタ・ノードのいずれかのジョブにおいて、そのユーザー・プロフ

ファイルがアクティブになっている場合、そのユーザー・プロファイルに関連付けられている MRE のグローバル値は「不整合」に設定されます。これは、ユーザー・プロファイルがアクティブになっていたジョブのノードで、変更操作が失敗したためです。この状況を訂正するには、ジョブの終了を待って、クラスター管理可能ドメインを終了する必要があります。管理可能ドメインが再始動する際に、不整合になっている各属性のグローバル値が使用されて、リソースが整合状態に変更されます。

ドメイン内のいずれかのノード上でリソースの削除、名前変更、または移動が行われると、モニター対象リソースのグローバル状況は必ず「失敗」に設定されます。そのような場合には、クラスター管理可能ドメインによるリソースの同期は行われなくなるので、MRE を除去する必要があります。

クラスター管理可能ドメインの一部である任意のシステム上にモニター対象リソースを復元すると、クラスター管理可能ドメインを表す対等 CRG がアクティブであれば、リソースは、クラスター管理可能ドメイン内で現在既知であるグローバル値に再同期されます。

復元コマンドの RSTLIB、RSTOBJ、RSTUSRPRF、RSTCFG を実行すると、システム・オブジェクトが再同期されます。RSTSYSINF および UPDSYSINF を実行すると、システム値とネットワーク属性が再同期されます。RSTSYSINF コマンドまたは UPDSYSINF コマンドを実行した後でシステム環境変数を再同期するには、クラスター管理可能ドメインを表す対等 CRG を終了して再始動する必要があります。

モニター対象リソースを以前の状態に復元するには、復元したいリソースを表す MRE を削除します。次に、リソースを復元した後で、復元操作を行ったシステムのリソースの MRE を追加します。クラスター管理可能ドメインは、復元されたリソースの値を使用して、ドメイン全体のモニター対象リソースを同期します。

クラスター管理可能ドメインをモニターするには、以下の手順で行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. 「クラスター・リソース・サービス」ページで、「管理可能ドメインの処理」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
4. 「管理可能ドメイン」タブで、「新規管理可能ドメイン」を選択します。
5. 「新規管理可能ドメイン」ページで、クラスター管理可能ドメインの情報を指定します。

クラスター管理可能ドメインの停止:

クラスター管理可能ドメインにより、i5/OS 高可用性ソリューション内で、リソース環境面での回復力が得られます。モニター対象リソースの同期を一時的に終了する場合に、クラスター管理可能ドメインを停止する必要があります。

クラスター管理可能ドメインは、停止時に非アクティブになります。クラスター管理可能ドメインが非アクティブになっている間、すべてのモニター対象リソースに不整合が生じることを考慮する必要がありますが、これは、リソースに加えられる変更内容の同期が行われなためです。モニター対象リソースに加えられる変更内容は引き続き追跡されますが、グローバル値は変更されず、変更内容は残りの管理可能ドメインに伝搬されません。クラスター管理可能ドメインが非アクティブになっている間にモニター対象リソースに加えられる変更内容は、クラスター管理可能ドメインの再開時にすべてのアクティブ・ノードで同期が行われます。

注: クラスター管理可能ドメインおよび関連する出口プログラムは、IBM 提供のオブジェクトです。これらのオブジェクトは、QcstChangeClusterResourceGroup API または CHGCRG コマンドを使用して変更しないでください。これらの変更を行うことで、予測不能の結果が生じます。

クラスター管理可能ドメインを停止するには、以下のステップを行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. 「クラスター・リソース・サービス」ページで、「**管理可能ドメインの処理**」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
4. 「管理可能ドメイン」ページで、クラスター管理可能ドメインを選択します。
5. 「**選択アクション**」メニューから、「**停止**」を選択します。
6. 「管理可能ドメインの停止を確認 (Stop Administrative Domain Confirmation)」ページで、「**はい**」をクリックします。

関連情報

クラスター管理可能ドメインの終了 (ENDCAD) コマンド (英語)

クラスター管理可能ドメインの削除:

クラスター・リソース・サービスのインターフェースを使用して、クラスター管理可能ドメインを削除することができます。クラスター管理可能ドメインが削除されると、クラスター管理可能ドメインに定義されていたモニター対象リソースの同期化が終了します。

クラスター管理可能ドメインを削除するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. 「クラスター・リソース・サービス」ページで、「**管理可能ドメインの処理**」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
4. 「管理可能ドメイン」ページで、クラスター管理可能ドメインを選択します。
5. 「**アクションの選択**」メニューから、「**削除**」を選択します。
6. 「管理可能ドメインの削除」確認ページで「**はい**」をクリックします。

クラスター管理可能ドメインのプロパティーの変更:

クラスター・リソース・サービスのグラフィカル・インターフェースを使用して、プロパティーを既存のクラスター管理可能ドメインに変更することができます。これらのプロパティーは、クラスター管理可能ドメインで定義されるモニター対象リソース項目の同期化を制御します。

クラスター管理可能ドメインのプロパティーを変更するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. 「クラスター・リソース・サービス」ページで、「**管理可能ドメインの処理**」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
4. 「管理可能ドメイン」ページで、クラスター管理可能ドメインを選択します。
5. 「**アクションの選択**」メニューから「**プロパティー**」を選択します。
6. 「プロパティー」ページで、クラスター管理可能ドメインに関する以下の情報を変更できます。
 - 「名前」フィールドに、クラスター管理可能ドメインの名前を入力します。10文字を超える名前は付けられません。

- 「同期オプション」フィールドで、ノードがクラスター管理可能ドメインを結合するときの同期動作を指定します。クラスターがバージョン 6 以降である場合にのみ、このフィールドを使用できます。使用可能な値は、次のとおりです。

最終変更オプション (デフォルト)

モニター対象リソースに対するすべての変更内容をクラスター管理可能ドメインに適用する場合は、このオプションを選択します。アクティブなクラスター管理可能ドメインとノードが結合している場合、結合するノード上のモニター対象リソースに対して行われた変更 (ノードがアクティブだったときのもの) は、ドメイン内の他のアクティブ・ノード上のモニター対象リソースに適用されます。ただし、アクティブなドメインのリソースに対して新たに変更が行われている場合は除きます。モニター対象リソースに行われた最新の変更内容が、すべてのアクティブ・ノード上のリソースに適用されます。

アクティブ・ドメイン・オプション

モニター対象リソースに行った変更内容のみをアクティブ・ノードから許可する場合に、このオプションを選択します。クラスター管理可能ドメインをノードが結合した場合、非アクティブ・ノード上でモニター対象リソースに行った変更は廃棄されます。ネットワーク・サーバー・ストレージ・スペース (*NWSSTG) またはネットワーク・サーバー構成 (*NWSCFG) には、アクティブ・ドメイン・オプションは適用されません。これらのリソースの同期化は、常に最後に変更された内容に基づいて行われます。

- 「管理可能ドメイン内のノード」リストにおいて、「追加」を選択すると、クラスター管理可能ドメインにノードを追加できます。また、「除去」を選択すると、ドメインからノードを除去することができます。

モニター対象リソース項目の管理:

クラスター・リソース・サービスのグラフィカル・インターフェースを使用すると、ご使用のクラスター管理可能ドメインのモニター対象リソース項目を管理することができます。これらのモニター対象リソースに変更を行った場合に、クラスター管理可能ドメインによって、高可用性環境内の各ノード内での整合性が保たれます。

モニター対象リソース項目の状況の処理:

クラスター・リソース・サービスのグラフィカル・インターフェースを使うと、クラスター管理可能ドメイン内のモニター対象リソース項目の状況メッセージを表示できます。

MRE がクラスター管理可能ドメインに追加されると、すべての管理可能ドメイン・ノードでのリソース変更がモニターされるため、リソース属性の値をクラスター管理可能ドメインの全ノードで同期することができます。同期の振る舞いは、さまざまな要因によって変化します。

- クラスターの状況
- クラスター管理可能ドメインの状況
- ノードの状況
- リソースに対する特定のアクション

モニター対象リソース項目の状況を処理するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。

4. 「クラスター・リソース・サービス」ページで、「管理可能ドメインの処理」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
5. 「管理可能ドメイン」ページで、クラスター管理可能ドメインの隣にあるコンテキスト・アイコンをクリックし、「モニター対象リソース・タイプの処理」を選択します。

注: 「モニター対象リソース・タイプの処理」アクションを使用できるのは、管理しているノードがクラスター管理可能ドメインの一部である場合のみです。モニター対象リソース・タイプの現行リストが表示されます。

6. 「モニター対象リソース・タイプの処理」ページで、リソース・タイプの隣にあるコンテキスト・アイコンをクリックし、「属性」を選択します。
7. モニター対象リソース項目の属性リストが表示されます。「グローバル状況」列には、アクティブなクラスター管理可能ドメインにおける、この属性の現在の状況が表示されます。

これらの値によって、クラスター全体のモニター対象リソースの状況を判別できます。

グローバル値

すべての管理可能ドメイン・ノードにおいてリソースが持っているとして想定される、各モニター対象属性の値です。グローバル値は、すべてのアクティブ・ノードにおいて同一であり、ドメイン内で同期された最新の変更内容を表します。

グローバル状況

クラスター管理可能ドメイン全体のリソース状況であり、リソースが完全に同期されているかどうかを表します。想定されるグローバル状況値は、次のとおりです。

整合 システムでモニターされるすべてのリソースの属性の値は、クラスター管理可能ドメイン内のどのアクティブ・ノードでも同じです。通常の操作環境において、クラスター、クラスター管理可能ドメイン、およびすべてのノードが作動可能で、クラスター内でアクティブになっている場合に、この状況が発生します。この環境では、モニター対象リソースの値に対するすべての変更が、クラスター管理可能ドメイン内の他のすべてのノードに伝搬されます。この処理は、元の変更に対しては非同期ですが、管理可能ドメイン全体の登録リソースの値とは整合することになります。この状態では、グローバル状況は「整合」になり、各ノードで変更が正常に行われ、各ノード上のリソースの値はリソースのグローバル値と一致します。

不整合 システムでモニターされているすべてのリソースの属性の値のうち、クラスター管理可能ドメイン内のアクティブ・ノードと一致しないものがあります。状況が「不整合」になった理由を説明するメッセージが記録されます。例えば、クラスター管理可能ドメインが非アクティブになっているときに、モニター対象リソースに対して変更が行われると、モニター対象リソースの状況は「不整合」になります。

保留中 モニター対象の属性の値は、クラスター管理可能ドメイン全体において同期処理の進行中にあります。

追加済み

モニター対象リソース項目がクラスター管理可能ドメインに追加されましたが、まだ同期は行われていません。

終了済み

クラスター管理可能ドメインが終了しており、リソースへの変更が処理されなくなっているため、モニター対象リソースの状況が不明です。クラスター管理可能ドメインが終了すると、現在「整合」に設定されている MRE のグローバル状況は、すべて「終了」に設定されます。

失敗 リソースはクラスター管理可能ドメインにモニターされなくなり、MRE は除去されます。リソースがクラスター管理可能ドメインにより同期されているときに、行わないほうがよいリソース・アクションがあります。MRE によって表されるリソースがシステム・オブジェクトである場合、そのリソースの削除や名前変更を行ったり、MRE を除去する前に異なるライブラリーに移動したりすることは避けてください。リソースが削除されたり、名前変更されたり、異なるライブラリーに移動されたりすると、MRE のグローバル状況が「失敗」になり、その後にノードのリソースに対して行った変更については、クラスター管理可能ドメインにあるどのノードにも伝搬されなくなります。

クラスター管理可能ドメイン内のノードにモニター対象リソースを復元すると、モニター対象リソースの値が変更されて、クラスター管理可能ドメインによって同期されたグローバル値と同じになります。

モニター対象リソース項目の除去:

モニター対象リソース項目 (MRE) は、高可用性環境で現在使用されているリソースで、クラスター管理可能ドメインを介して変更をモニターされています。モニターを行う必要がない場合は、MRE を除去してもかまいません。モニター対象リソース項目 (MRE) を除去するには、クラスター・リソース・サービスのグラフィカル・インターフェースを使用してください。

モニター対象リソース項目を除去するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで、「管理可能ドメインの処理」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
5. 「管理可能ドメイン」ページで、クラスター管理可能ドメインの隣にあるコンテキスト・アイコンをクリックし、「モニター対象リソース・タイプの処理」を選択します。

注: 「モニター対象リソース・タイプの処理」アクションを使用できるのは、管理しているノードがクラスター管理可能ドメインの一部である場合のみです。モニター対象リソース・タイプの現行リストが表示されます。

6. モニター対象リソース・タイプのリストで、モニター対象リソース・タイプの隣にあるコンテキスト・アイコンをクリックして、「モニター対象リソース項目の処理」を選択します。MRE オブジェクト・リストが表示されます。
7. 除去する MRE オブジェクトの隣にあるコンテキスト・アイコンをクリックして、「モニター対象リソース項目の除去」を選択します。
8. 「MRE オブジェクトの除去」確認ウィンドウで「はい」をクリックします。モニター対象リソース項目が、クラスター管理可能ドメインから除去されます。

関連情報

管理ドメイン MRE の除去 (RMVCADMRE) コマンド (英語)

モニター対象リソース項目の除去 (QfpadRmvMonitoredResourceEntry) API (英語)

モニター対象リソース・タイプのリスト表示:

モニター対象リソース・タイプとは、ユーザー・プロファイルや環境変数などのシステム・オブジェクトのことで、クラスター管理可能ドメインによってモニターすることができます。クラスター管理可能ドメイン内で現在指定されているモニター対象リソース・タイプを、リスト表示することができます。

モニター対象リソース・タイプをリスト表示するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**管理可能ドメインの処理**」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
5. 「**管理可能ドメイン**」ページで、クラスター管理可能ドメインの隣にあるコンテキスト・アイコンをクリックし、「**モニター対象リソース・タイプの処理**」を選択します。

注: 「**モニター対象リソース・タイプの処理**」アクションを使用できるのは、管理しているノードがクラスター管理可能ドメインの一部である場合のみです。モニター対象リソース・タイプの現行リストが表示されます。

6. 選択したモニター対象リソース・タイプに対して、必要なアクションを実行します。
 - モニター対象リソース属性の処理
 - 新規モニター対象リソース項目の追加

1 | **モニター対象リソース項目のリスト表示:**

1 | モニター対象リソース項目とは、クラスター管理可能ドメインで定義されたりソース (ユーザー・プロファイル、環境変数など) のことです。クラスター・リソース・サービスのグラフィカル・インターフェースを使用して、クラスター管理可能ドメインで現在定義されているモニター対象リソース項目をリスト表示することができます。

1 | モニター対象リソース項目をリスト表示するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**管理可能ドメインの処理**」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
5. 「**管理可能ドメイン**」ページで、クラスター管理可能ドメインの隣にあるコンテキスト・アイコンをクリックし、「**モニター対象リソース・タイプの処理**」を選択します。

1 | **注:** 「**モニター対象リソース・タイプの処理**」アクションを使用できるのは、管理しているノードがクラスター管理可能ドメインの一部である場合のみです。モニター対象リソース・タイプの現行リストが表示されます。

6. モニター対象リソース・タイプのリストで、モニター対象リソース・タイプの隣にあるコンテキスト・アイコンをクリックして、「**モニター対象リソース項目の処理**」を選択します。
7. 登録されたモニター対象リソース項目のリストを表示して、操作します。

モニターする属性の選択:

モニター対象リソース項目を追加した後で、クラスター管理可能ドメインによってモニターされるそのリソースに関連付けられた属性を選択することができます。

モニター対象リソース項目 (MRE) について、モニターする属性を選択するには、以下のステップを実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**管理可能ドメインの処理**」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
5. 「**管理可能ドメイン**」ページで、クラスター管理可能ドメインの隣にあるコンテキスト・アイコンをクリックし、「**モニター対象リソース・タイプの処理**」を選択します。

注: 「**モニター対象リソース・タイプの処理**」アクションを使用できるのは、管理しているノードがクラスター管理可能ドメインの一部である場合のみです。モニター対象リソース・タイプの現行リストが表示されます。

6. モニター対象リソース・タイプのリストで、モニター対象リソース・タイプの隣にあるコンテキスト・アイコンをクリックして、「**モニター対象リソース項目の処理**」を選択します。MRE オブジェクト・リストが表示されます。
7. ユーザー・プロファイルやシステム値などの MRE オブジェクトの隣のコンテキスト・アイコンをクリックし、「**属性の処理 (Work with Attributes)**」を選択します。MRE 属性リストが表示されます。
8. 「**MRE 属性リスト**」ウィンドウで、モニターする属性を選択し、「**閉じる**」をクリックします。例えば、リソース名属性への変更について、イーサネット回線記述をモニターする場合は、属性としてリソース名を選択します。

関連タスク

114 ページの『モニター対象リソース項目の追加』

クラスター管理可能ドメインに、モニター対象リソース項目 (MRE) を追加することができます。モニター対象リソース項目はクリティカル・リソースを定義するため、これらのリソースに対して変更を行った場合、高可用性環境で矛盾がないように整合されます。

モニター可能な属性:

モニター対象リソース項目は、さまざまなタイプのリソースのクラスター管理可能ドメインに追加できます。このトピックには各リソース・タイプをモニターすることができる属性がリストされています。

リソース・タイプ

- クラス (*CLS)
- l • イーサネット回線基準 (*ETHLIN)
- 独立ディスク・プール・デバイスの記述 (*ASPDEV)
- ジョブ記述 (*JOBDEV)
- ネットワーク属性 (*NETA)
- l • 接続セキュリティー用ネットワーク・サーバーの構成 (*NWSCFG)
- l • リモート・システム用ネットワーク・サーバーの構成 (*NWSCFG)
- l • サービス・プロセッサ用ネットワーク・サーバーの構成 (*NWSCFG)
- l • iSCSI 接続用ネットワーク・サーバーの記述 (*NWSD)
- l • 統合ネットワーク・サービス用ネットワーク・サーバーの記述 (*NWSD)
- l • ネットワーク・サーバー・ストレージ・スペース (*NWSSTG)

- | • ネットワーク・サーバー・ホスト・アダプター装置の記述 (*NWSHDEV)
- | • 光ディスク装置記述 (*OPTDEV)
- | • サブシステム記述 (*SBSD)
 - システム環境変数 (*ENVVAR)
 - システム値 (*SYSVAL)
- | • 磁気テープ装置記述 (*TAPDEV)
- | • トークンリング回線記述 (*TRNLIN)
 - TCP/IP 属性 (*TCPA)
 - ユーザー・プロファイル (*USRPRF)

表 15. クラスでモニター可能な属性

属性名	説明
CPUTIME	最大 CPU 時間
DFTWAIT	デフォルト待ち時間
MAXTHD	最大スレッド数
MAXTMPSTG	最大一時ストレージ
RUNPTY	実行優先順位
TEXT	テキスト記述
TIMESLICE	タイム・スライス

表 16. イーサネット回線記述でモニター可能な属性

属性名	説明
ASSOCPORT	関連ポート・リソース名
AUTOCTRL	自動作成コントローラー
AUTODLCTRL	自動削除コントローラー
CMNRCYLMT	回復限界
COSTBYTE	回線でのデータの送受信の 1 バイト当たりの相対コスト
COSTCNN	回線に接続していることで生じる相対コスト
DUPLEX	二重
GENTSTFRM	テスト・フレームの生成
GRPADR	グループ・アドレス
LINESPEED	回線速度
MAXFRAME	最大フレーム・サイズ
MAXCTL	最大コントローラー
MSGQ	メッセージ・キュー
ONLINE	IPL 時のオンライン
PRPDLY	伝搬遅延
RSRCNAME	リソース名
SECURITY	物理回線のセキュリティー・レベル
SSAP	ソース・サービス・アクセス・ポイント (SSAP) 情報リスト
TEXT	テキスト記述

表 16. イーサネット回線記述でモニター可能な属性 (続き)

属性名	説明
USRDFN1	最初のユーザー定義
USRDFN2	2 番目のユーザー定義
USRDFN3	3 番目のユーザー定義
VRYWAIT	「待ち」をオンに変更

表 17. 独立ディスク・プールの装置記述でモニター可能な属性

属性名	説明
MSGQ	メッセージ・キュー
RDB	リレーショナル・データベース
RSRCNAME	リソース名
TEXT	テキスト記述

表 18. ジョブ記述でモニター可能な属性

属性名	説明
ACGCDE	アカウント・コード
ALWMLTTHD	マルチスレッドを許可
DDMCNV	DDM 会話
DEVRCYACN	装置回復アクション
ENDSEV	重大度終了
HOLD	ジョブ・キューの保留
INLASGRP	初期 ASP グループ
INQMSGRPY	照会メッセージ応答
JOBMSGQFL	ジョブ・メッセージ・キュー・フル・アクション
JOBMSGQMX	ジョブ・メッセージ・キュー最大サイズ
JOBPTY	ジョブ優先順位 (JOBQ に対する)
JOBQ	ジョブ・キュー
LOG	メッセージ・ロギング
LOGCLPGM	制御言語プログラムのログ
OUTPTY	出力優先順位 (OUTQ に対する)
OUTQ	出力キュー
PRTDEV	印刷装置
PRTTXT	印刷テキスト
RQSDTA	データまたはコマンドの要求
RTGDTA	ルーティング・データ
SPLFACN	スプール・ファイル・アクション
SWS	ジョブ切り替え
SYNTAX	CL 構文検査
TEXT	テキスト記述
TSEPOOL	タイム・スライス・エンド・プール
ユーザー	ユーザー

表 19. ネットワーク属性でモニター可能な属性

属性名	説明
ALWADDCLU	クラスターへの追加を許可
DDMACC	DDM/DRDA 要求アクセス
NWSDOMAIN	ネットワーク・サーバー・ドメイン
PCSACC	クライアント要求アクセス
注: 各ネットワーク属性は、それぞれのモニター対象リソース項目として扱われます。このため、リソース・タイプと属性名は同一です。	

表 20. サービス・プロセッサのネットワーク・サーバー構成でモニター可能な属性

属性名	説明
EID	筐体識別子
ENBUNICAST	ユニキャストの使用可能化
INZSP	サービス・プロセッサの初期化
SPAUT	サービス・プロセッサ権限
SPCERTID	サービス・プロセッサ証明書識別子
SPINTNETA	サービス・プロセッサのインターネット・アドレス
SPNAME	サービス・プロセッサ名
TEXT	テキスト記述

表 21. リモート・システムのネットワーク・サーバー構成でモニター可能な属性

属性名	説明
BOOTDEVID	ブート・デバイス識別子
CHAPAUT	ターゲット CHAP 認証
DELIVERY	配信方式
DYNBOOTOPT	動的ブート・オプション
INRCHAPAUT	イニシエーター CHAP 認証
RMTIFC	リモート・インターフェース
RMTSYSID	リモート・システム識別子
SPNWSCFG	リモート・サーバーの管理に使用されるサービス・プロセッサ・ネットワーク・サーバー構成
TEXT	テキスト記述

表 22. 接続の安全保護のネットワーク・サーバー構成でモニター可能な属性

属性名	説明
IPSECRULE	IP セキュリティ規則
TEXT	テキスト記述

表 23. 統合ネットワーク・サーバーのネットワーク・サーバー記述でモニターできる属性

属性名	説明
CFGFILE	構成ファイル

表 23. 統合ネットワーク・サーバーのネットワーク・サーバー記述でモニターできる属性 (続き)

属性名	説明
CODEPAGE	このネットワーク・サーバーで使用される文字セットを表す ASCII コード・ページ
EVTLOG	イベント・ログ
MSGQ	メッセージ・キュー
NWSSTGL	ストレージ・スペースのリンク
PRPDMNUSR	ドメイン・ユーザーの伝搬
RSRCNAME	リソース名
RSTDDEVRSC	制限付き装置リソース
SHUTDTIMO	シャットダウン・タイムアウト
SYNCTIME	日時の同期
TCPDMNNAME	TCP/IP ローカル・ドメイン・ネーム
TCPHOSTNAM	TCP/IP ホスト名
TCPPORTCFG	TCP/IP ポート構成
TCPNAMSVR	TCP/IP ネーム・サーバー・システム
TEXT	テキスト記述
VRYWAIT	「待ち」をオンに変更
WINDOWSNT	Windows® ネットワーク・サーバー記述

表 24. iSCSI 接続用ネットワーク・サーバー記述でモニター可能な属性

属性名	説明
ACTTMR	アクティベーション・タイマー
CFGFILE	構成ファイル
CMNMSGQ	通信メッセージ・キュー
CODEPAGE	このネットワーク・サーバーで使用される文字セットを表す ASCII コード・ページ
DFTSECRULE	デフォルト IP セキュリティ規則
DFTSTGPTH	デフォルト・ストレージ・パス
EVTLOG	イベント・ログ
MLTPHGRP	マルチパス・グループ
MSGQ	メッセージ・キュー
NWSCFG	ネットワーク・サーバー構成
NWSSTGL	ストレージ・スペースのリンク
PRPDMNUSR	ドメイン・ユーザーの伝搬
RMVMEDPTH	取り外し可能メディア・パス
RSRCNAME	リソース名
RSTDDEVRSC	制限付き装置リソース
SHUTDTIMO	シャットダウン・タイムアウト
STGPTH	ネットワーク・サーバーの iSCSI ストレージ・パス
SYNCTIME	日時の同期
TCPDMNNAME	TCP/IP ローカル・ドメイン・ネーム

表 24. iSCSI 接続用ネットワーク・サーバー記述でモニター可能な属性 (続き)

属性名	説明
TCPHOSTNAM	TCP/IP ホスト名
TCPNAMSVR	TCP/IP ネーム・サーバー・システム
TCPPORTCFG	TCP/IP ポート構成
TEXT	テキスト記述
VRTETHCTLP	仮想イーサネット制御ポート
VRTETHPTH	仮想イーサネット・パス
VRYWAIT	「待ち」をオンに変更

表 25. ネットワーク・サーバー・ストレージ・スペースでモニター可能な属性

属性名	説明
SIZE	サイズ
TEXT	テキスト記述
TOTALFILES	ファイル合計

表 26. ネットワーク・サーバー・ホスト・アダプター装置記述でモニター可能な属性

属性名	説明
CMNRCYLMT	回復限界
LCLIFC	関連ローカル・インターフェース
MSGQ	メッセージ・キュー
ONLINE	IPL 時のオンライン
RSRCNAME	リソース名
TEXT	テキスト記述

表 27. 光ディスク装置記述でモニター可能な属性

属性名	説明
MSGQ	メッセージ・キュー
ONLINE	IPL 時のオンライン
RSRCNAME	リソース名
TEXT	テキスト記述

表 28. サブシステムジョブ記述でモニター可能な属性

属性名	説明
AJE	自動開始ジョブ項目
CMNE	IPL 時のオンライン
JOBQE	ジョブ・キュー
MAXJOBS	ジョブの最大数
PJE	事前開始ジョブ項目
RMTLOCNAME	リモート・ロケーション名
RTGE	ルーティング項目
SGNDSPF	サインオン表示

表 28. サブシステムジョブ記述でモニター可能な属性 (続き)

属性名	説明
SYSLIBLE	サブシステム・ライブラリー
TEXT	テキスト記述
WSNE	ワークステーション名前項目
WSTE	ワークステーション・タイプ項目

表 29. システム環境変数でモニター可能な属性

すべての *SYS レベルの環境変数をモニターできます。属性およびリソース名はどちらも環境変数の名前と同じです。
注: 各環境変数は、それぞれのモニター対象リソース項目として扱われます。このため、リソース・タイプと属性名は同一です。

表 30. システム値でモニター可能な属性

属性名	説明
QACGLVL	会計レベル
QACTJOBITP	ジョブの中断を許可
QALWOBJRST	第三者がシステム状態のオブジェクトまたは権限を持つオブジェクトを復元するのを防ぎます。
QALWUSRDMN	ユーザー・ドメイン・オブジェクトを許可します
QASTLVL	操作援助レベル
QATNPGM	アテンション・プログラム
QAUDCTL	監査制御
QAUDENDACN	監査ジャーナル・エラー・アクション
QAUDFRCLVL	監査強制レベル
QAUDLVL	監査レベル
QAUDLVL2	監査レベルの拡張
QAUTOCFG	自動装置構成
QAUTORMT	リモート・コントローラーおよび装置
QAUTOVRT	自動仮想装置構成
QCCSID	コード化文字セット ID
QCFGMSGQ	回線、コントローラー、および装置のメッセージ・キュー
QCHRID	データの表示または印刷に使用されるデフォルトの図形文字セットおよびコード・ページ
QCHRIDCTL	ジョブの文字 ID 制御
QCMNRCYLMT	自動通信エラー回復
QCNTYID	国 ID または地域 ID
QCRTAUT	新規オブジェクトの権限
QCRTOBJAUD	新規オブジェクトの監査
QCTLBSD	サブシステムまたはライブラリーの制御
QCURSYM	通貨記号
QDATFMT	日付形式
QDATSEP	日付区切り記号

表 30. システム値でモニター可能な属性 (続き)

属性名	説明
QDBRCVYWT	再始動の終了前にデータベースの回復を待機
QDECFMT	10 進数フォーマット
QDEVNAMING	装置名の命名規則
QDEVRCYACN	装置回復アクション
QDSCJOBITV	切断ジョブのタイムアウト間隔
QDSPSGNINF	サインオン情報の表示を制御する
QENDJOB LMT	即時終了の最大時間
QFRCCVNRST	復元時の強制変換
QHSTLOGSIZ	履歴・ログのファイル・サイズ
QIGCCDEFNT	コード化フォントの名前
QIGCFNTSIZ	コード化フォントのポイント・サイズ
QINACTITV	非アクティブ・ジョブのタイムアウト間隔
QINACTMSGQ	タイムアウト間隔アクション
QIPLTYPE	再始動のタイプ
QJOBMSGQFL	ジョブ・メッセージ・キュー・フル・アクション
QJOBMSGQMX	ジョブ・メッセージ・キュー最大サイズ
QJOBMSGQSZ	ジョブ・メッセージ・キューの初期サイズ (キロバイト (KB) 単位)
QJOBMSGQTL	ジョブ・メッセージ・キューの最大サイズ (KB)
QJOBSPLA	スプール・コントロール・ブロックの初期サイズ (バイト)
QKBDBUF	キーボード・バッファ
QKBDTYPE	キーボード言語文字セット
QLANGID	デフォルトの言語 ID
QLIBLCKLVL	ユーザー・ジョブのライブラリー検索リストのライブラリーをロック
QLMTDEVSSN	装置セッションの制限
QLMTSECOFR	セキュリティ・オフィサー装置アクセスの制限
QLOCALE	ロケール
QLOGOUTPUT	ジョブ・ログのプリンター出力の作成
QMAXACTLVL	システムの最大アクティビティ・レベル
QMAXJOB	システムに許可される最大ジョブ数
QMAXSGNACN	QMAXSIGN システム値で指定された制限に達したときのシステムの応答
QMAXSIGN	許可される無効なサインオンの最大試行数
QMAXSPLF	最大プリンター出力ファイル数
QMLTTHDACN	マルチスレッド・ジョブの機能がスレッド・セーフでない場合
QPASTHRSVR	使用可能なディスプレイ装置パススルー・サーバー・ジョブ
QPRBFTR	問題ログ・フィルター

表 30. システム値でモニター可能な属性 (続き)

属性名	説明
QPRBHLDITV	最小保存
QPRTDEV	デフォルト・プリンター
QPRTKEYFMT	印刷キーのフォーマット
QPRTTXT	リストおよびセパレーター・ページの下部には最大 30 文字のテキストを印刷できます。
QPWDCHGBLK	パスワード変更から次の変更までの最小時間
QPWDEXPITV	パスワードが有効である日数
QPWDEXPWPN	パスワード有効期限の警告間隔システム
QPWDLMTACJ	パスワードにおける隣接回数の使用の制限
QPWDLMTCHR	パスワードにおける指定文字の使用の制限
QPWDLMTREP	パスワードにおける繰り返し文字の使用の制限
QPWDLVL	パスワード・レベル
QPWDMAXLEN	パスワードの最大文字数
QPWDMINLEN	パスワードの最小文字数
QPWDPOSDF	新規パスワードの文字位置の制御
QPWDRQDDGT	新規パスワードに数字が必要
QPWDRQDDIF	パスワードが前のパスワードと違っている必要があるかどうかの制御
QPWDRULES	パスワード・ルール
QPWDVLDPGM	パスワード承認プログラム
QPWRDWNLMT	即時シャットダウンの最大時間
QRCLSPLSTG	未使用のプリンター出カストレージの自動クリーンアップ
QRETSVRSEC	サーバー・セキュリティー・データ標識の保存
QRMTSIGN	リモート・サインオン
QRMTSRVATR	リモート・サービス属性
QSCANFS	ファイル・システムの走査
QSCANFCTL	走査制御
QSCPFCONS	コンソールの問題発生時の処置
QSECURITY	システム・セキュリティー・レベル
QSETJOBATR	ジョブ属性の設定
QSFWERRLOG	ソフトウェア・エラー・ログ
QSHRMEMCTL	書き込み機能を使用した共用またはマップ済みメモリーの使用を許可
QSPCENV	デフォルト・ユーザー環境
QSPLFACN	スプール・ファイル・アクション
QSRTSEQ	ソート・シーケンス
QSRVDMP	モニターされていないエスケープ・メッセージのサービス・ログ
QSSLCSL	Secure Sockets Layer 暗号化仕様のリスト
QSSLCSLCTL	Secure Sockets Layer 暗号制御

表 30. システム値でモニター可能な属性 (続き)

属性名	説明
QSSLPCL	Secure Sockets Layer プロトコル
QSTRUPGM	開始プログラムの設定
QSTSMMSG	状況メッセージの表示
QSYSLIBL	システム・ライブラリー・リスト
QTIMSEP	時間区切り記号
QTSEPOOL	対話式ジョブがタイム・スライス終了に達したときに別の主記憶域プールに移動するかどうかを示します。
注: 各システム値は、独自のモニター対象リソース項目として扱われます。このため、リソース・タイプと属性名は同一です。	

表 31. 磁気テープ装置でモニター可能な属性

属性名	説明
ASSIGN	オンに変更時に装置を割り当てる
MSGQ	メッセージ・キュー
ONLINE	IPL 時のオンライン
RSRCNAME	リソース名
TEXT	テキスト記述
UNLOAD	オフに変更時に装置をアンロード

表 32. トークンリング記述でモニター可能な属性

属性名	説明
ACTLANMGR	アクティブ LAN マネージャー
ADPTADR	ローカル・アダプター・アドレス
AUTOCTCTL	自動作成コントローラー
AUTODLCTL	自動削除コントローラー
CMNRCYLMT	回復限界
COSTBYTE	回線でのデータの送受信の 1 バイト当たりの相対コスト
COSTCNN	回線に接続していることで生じる相対コスト
DUPLEX	二重
ELYTKNRLS	早期トークン解放
FCNADR	機能アドレス
LINESPEED	回線速度
LINKSPEED	リンク速度
LOGCFGCHG	ログ構成変更
MAXCTL	最大コントローラー
MAXFRAME	最大フレーム・サイズ
MSGQ	メッセージ・キュー
ONLINE	IPL 時のオンライン
PRPDLY	伝搬遅延
RSRCNAME	リソース名

表 32. トークンリング記述でモニター可能な属性 (続き)

属性名	説明
SECURITY	回線のセキュリティー
SSAP	ソース・サービス・アクセス・ポイント (SSAP) 情報リスト
TRNINFBDN	トークンリングのビーコンの通知
TRNLOGLVL	TRLAN マネージャー・ロギング・レベル
TRNMGRMODE	TRLAN マネージャー・モード
TEXT	トークンリング回線のテキスト記述
USRDFN1	最初のユーザー定義
USRDFN2	2 番目のユーザー定義
USRDFN3	3 番目のユーザー定義
VRYWAIT	「待ち」をオンに変更

表 33. TCP/IP 属性でモニター可能な属性

属性名	説明
ARPTIMO	アドレス解決プロトコル (ARP) キャッシュ・タイムアウト
ECN	明示的輻輳通知の使用可能化 (ECN)
IPDEADGATE	IP デッド・ゲートウェイ検出
IPDTGFWD	IP データグラム転送
IPPATHMTU	パスの最大伝送単位 (MTU) ディスカバリー
IPQOSBCH	IP QoS データグラムのバッチ処理
IPQOSEN	IP QoS イネーブルメント
IPQOSTMR	IP QoS タイマー分解能
IPRSBTIMO	IP 再アセンブリー・タイムアウト
IPSRCRTG	IP ソース・ルーティング
IPTTL	IP 存続時間 (ホップ限界)
LOGPCLERR	プロトコル・エラーのログ
NFC	ネットワーク・ファイル・キャッシュ
TCPCLOTIMO	TCP 時間待ちタイムアウト
TCPCNNMSG	TCP 接続クローズ・メッセージ
TCPKEEPALV	TCP キープアライブ
TCPMINRTM	TCP 最小再送時間
TCPR1CNT	TCP R1 再送信数
TCPR2CNT	TCP R2 再送信数
TCPRCVBUF	TCP 受信バッファ・サイズ
TCPSNDBUF	TCP 送信バッファ・サイズ
TCPURGPTR	TCP 緊急ポインター
UDPCKS	UDP チェックサム

注: 各 TCP/IP 属性は、それぞれのモニター対象リソース項目として扱われます。このため、リソース・タイプと属性名は同一です。

表 34. ユーザー・プロフィールでモニター可能な属性

属性名	説明
ACGCDE	アカウント・コード
ASTLVL	操作援助レベル
ATNPGM	アテンション・プログラム
CCSID	コード化文字セット ID
CHRIDCTL	文字 ID の制御
CNTRYID	国 ID または地域 ID
CURLIB	現行ライブラリー
DLVRY	配信
DSPSGNINF	サインオン情報の表示
GID	グループ ID 番号
GRPAUT	グループ権限
GRPAUTTYP	グループ権限タイプ
GRPPRF	グループ・プロフィール
HOMEDIR	ホーム・ディレクトリー
INLMNU	初期メニュー
INLPGM	呼び出す初期プログラム
JOBID	ジョブ記述
KBDBUF	キーボード・バッファリング
LANGID	言語 ID
LCLPDMGT	ローカル・パスワード管理
LMTCPB	機能の制限
LMTDEVSSN	装置セッションの制限
LOCALE	ロケール
MAXSTG	最大許可ストレージ
MSGQ	メッセージ・キュー
OUTQ	出力キュー
OWNER	所有者
PASSWORD	ユーザー・パスワード
PRTDEV	印刷装置
PTYLMT	最優先スケジューリング優先順位
PWDEXP	パスワードを有効期限切れに設定
PWDEXPITV	パスワード有効期限間隔
SETJOBATR	ロケール・ジョブ属性
SEV	重大度コード・フィルター
SPCAUT	特殊権限
SPCENV	特殊環境
SRTSEQ	ソート・シーケンス
STATUS	状況
SUPGRPPRF	補助グループ
TEXT	テキスト記述

表 34. ユーザー・プロファイルでモニター可能な属性 (続き)

属性名	説明
UID	ユーザー ID 番号
USRCLS	ユーザー・クラス
USROPT	ユーザー・オプション

モニター対象リソース項目メッセージの表示:

クラスター・リソース・サービスのグラフィカル・インターフェースを使用して、モニター対象リソース項目に関連付けられたメッセージを表示できます。

モニター対象リソース項目メッセージを表示して確認するには、以下を行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログインします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「クラスター・リソース・サービス」を選択します。
4. 「クラスター・リソース・サービス」ページで、「管理可能ドメインの処理」をクリックして、クラスター内のクラスター管理可能ドメインのリストを表示します。
5. 「管理可能ドメイン」ページで、クラスター管理可能ドメインの隣にあるコンテキスト・アイコンをクリックし、「モニター対象リソース・タイプの処理」を選択します。

注: 「モニター対象リソース・タイプの処理」アクションを使用できるのは、管理しているノードがクラスター管理可能ドメインの一部である場合のみです。モニター対象リソース・タイプの現行リストが表示されます。

6. モニター対象リソース・タイプのリストで、モニター対象リソース・タイプの隣にあるコンテキスト・アイコンをクリックして、「モニター対象リソース項目の処理」を選択します。MRE オブジェクト・リストが表示されます。
7. ユーザー・プロファイルやシステム値など、MRE オブジェクトの横にあるコンテキスト・アイコンをクリックし「メッセージの表示...」を選択します。

切り替えディスクの管理

切り替えディスクは、デバイス・クラスター・リソース・グループ (CRG) の一部として構成された、独立ディスク・プールのことです。切り替えディスクに保管されたデータおよびアプリケーションの所有権は、デバイス CRG で定義された他のシステムに切り替えることができます。切り替えディスクのテクノロジーは、計画停止および計画外の停止時における高可用性を提供します。

ディスク・プールを使用不可にする

独立ディスク・プールを選択して使用不可 (オフに変更) にすることができます。再び使用可能 (オンに変更) するまで、独立ディスク・プールまたは対応するデータベースのディスク装置またはオブジェクトにアクセスすることはできません。プールは、同じシステムまたはクラスター・リソース・グループのリカバリー・ドメイン内の別のシステムで使用可能にできます。

重要: 独立ディスク・プールが使用不可になるまでは、ジョブはディスク・プール上で予約を保留できません。ジョブが独立ディスク・プールを使用中かどうかの判断、およびジョブ予約の保留解除の方法に関しては、独立ディスク上のジョブの予約の保留解除を参照してください。

System i ナビゲーターを使用して UDFS ディスク・プールを使用不可にするときに、文字ベース・インターフェイスでの応答が必要なメッセージが生成される場合があります。System i ナビゲーターには、メッセージの待機表示は示されません。

独立ディスク・プールを使用不可にするには、以下の手順を行います。

1. System i ナビゲーターで、「**ユーザー接続**」(またはアクティブな環境)を展開します。
2. 「**構成およびサービス**」 → 「**ハードウェア**」 → 「**ディスク装置**」で、検査する System i を展開します。
3. 「**ディスク・プール**」を展開します。
4. 使用不可にするディスク・プールを右クリックして、「**使用不可にする (Make Unavailable)**」を選択します。
5. 表示されるダイアログ・ボックスから、「**使用不可にする (Make Unavailable)**」を選択して、ディスク・プールを使用不可の状態にします。

文字ベース・インターフェイスの構成の変更 (VRYCFG) コマンドを使用して、ディスク・プールを使用不可にすることができます。

ステップが処理のどの段階にあるかを判断するには、ASP 状況の表示 (DSPASPSTS) コマンドを使用します。

ASP に対するアクセス権限を所有する処理を制限するには、ASP アクセスの制御 API (QYASPCTLAA) を使用します。

DASD 管理操作の開始 (QYASSDMO) API を使用して、ディスク・プールを使用不可にするためにかかる時間を短縮します。

ハードウェアを切り替え可能にする

i5/OS 高可用性環境では、外部拡張装置を切り替え可能にする必要があります。

切り替え可能な環境で独立ディスク・プールを使用している場合、関連したハードウェアでも切り替えが許可されるようにする必要があります。ご使用の環境によっては、フレーム、装置、または IOP やその関連リソースが、これに含まれる場合があります。以下のステップを参照して、切り替え可能な環境に適用してください。

フレームまたは装置を切り替え可能にする

独立ディスク・プールでは、複数の拡張装置内にディスク装置を組み込むことができます。独立ディスク・プールに組み込まれたディスクを持つ、スタンドアロンの拡張装置がある場合、他のシステムへのアクセスを許可するように拡張装置に権限を与える必要があります。このことを、「拡張装置を切り替え可能にする」と呼びます。他のシステムからスタンドアロンの拡張装置にアクセスさせない場合は、拡張装置をプライベートにする必要があります。

フレームまたは装置を切り替え可能にするには、以下の手順を実行します。

1. System i ナビゲーターで、「**ユーザー接続**」(またはアクティブな環境)を展開します。
2. 調べたいシステムを展開して、「**構成およびサービス**」 → 「**ハードウェア**」 → 「**ディスク装置**」 → 「**ロケーションごと (By Location)**」とクリックして展開し、切り替え可能にするフレームまたはディスク装置を選択します。
3. ハイライトされたフレームまたはディスク装置を右クリックして、「**切り替え可能にする (Make Switchable)**」を選択します。

4. 表示されたダイアログ・ボックスの指示に従います。

IOP を切り替え可能にする

IOP を切り替え可能にするには、切り替えられるディスク装置を制御する IOP を含むバスが、プライマリ・ノードにより共有所有 されている必要があります。このバスは、バックアップ・ノードとの共有バスを使用 する必要もあります。詳しくは、『区画間での IOP の動的切り替え』を参照してください。

このタスクを実行するには、専用保守ツールのシステム区画機能に対する管理者権限を持つ、保守ツールのユーザー・プロファイルが必要になります。論理区画特権の取得について詳しくは、『論理区画権限』を参照してください。

マネージメント・セントラルを使用してバスの所有権タイプを変更するには、以下の手順を実行します。

1. System i ナビゲーターで、「**ユーザー接続**」（またはアクティブな環境）を展開します。
2. システムの 1 次区画を選択します。
3. 「**構成およびサービス**」を展開して、「**論理区画**」を選択します。
4. 「**論理区画**」を右クリックして、「**区画の構成 (Configure Partitions)**」を選択します。
5. 「**論理区画の構成 (Configure Logical Partitions)**」ウィンドウで、所有権を変更するバスを右クリックして「**プロパティ**」を選択します。
6. 「**区画**」ページを選択します。
7. 「**論理区画の所有 (Owning logical partition)**」でバスを所有する区画を選択し、「**共用**」で所有権タイプを選択します。所有権タイプが共用の場合、バスを共用する区画がリストに表示されます。これらのオプションについて詳しく知りたい場合には、「**ヘルプ**」をクリックしてください。
8. 「**OK**」をクリックします。

ハードウェア管理コンソールによって入出力プールを切り替え可能にする

ハードウェア管理コンソールを使用して論理区画を管理する場合には、IOP、IOA、および接続済みの全リソースから成る入出力プールを構成して、単一の独立ディスク・プールを複数の区画の間で切り替えられるようにしてください。各区画プロファイルに入出力プールを割り当てることにより、独立ディスク・プールの所有者となる各区画へのアクセスを許可する必要があります。

区画間で切り替え可能な入出力プールを作成するには、以下の手順を実行します。

1. 「**論理区画プロファイル・プロパティ (Logical Partition Profile Properties)**」ウィンドウを開き、区画プロファイル・プロパティを変更して、入出力プールにリソースを割り当てます。
2. 「**物理 I/O**」タブをクリックします。
3. 「**プロファイル入出力装置 (Profile I/O devices)**」列で、切り替え可能にする入出力を含むバスを拡張します。
4. 入出力プールに割り当てる IOP を選択します。IOP は、**推奨**（「**必須**」列にチェック・マークがない）にする必要があります。
5. 「**入出力プール (I/O pool)**」列をクリックします。入出力プールに割り当てる入出力の行にカーソルが表示されたら、入出力プールの番号を入力してください。
6. これらのステップを繰り返して、IOP の制御下にある IOA とリソースを、それぞれ入出力プールに追加します。
7. 「**OK**」をクリックします。

入出力プールに区画を関連付ける

入出力プールにリソースを追加したら、以下の手順を実行して、切り替え可能な環境にある独立ディスク・プールの所有者になることができる追加の区画を、それぞれ入出力プールに関連付けます。

1. 「論理区画プロファイル・プロパティ (Logical Partition Profile Properties)」ウィンドウを開いて、独立ディスク・プールにアクセスする必要がある追加の各区画の区画プロファイル・プロパティを変更します。
2. 「物理 I/O」タブをクリックします。
3. 「詳細設定 (Advanced)」をクリックします。
4. 「入出力プール (I/O pool)」ウィンドウの、「追加する入出力プール (I/O pools to add)」フィールドで、独立ディスク・プールと切り替えるリソースを割り当てた入出力プールの番号を入力します。
5. 「追加」 → 「OK」とクリックします。

入出力プールの変更内容を有効にするには、区画プロファイルが変更された各区画に対して、以下の手順を実行します。

1. 区画をシャットダウンします。『論理区画における i5/OS の再始動とシャットダウン』を参照してください。
2. 変更内容が反映されるように区画プロファイルを活動化して、論理区画を開始します。

関連概念

区画間での IOP の動的切り替え

論理区画権限

I/O プール (英語)

関連タスク

区画プロファイル・プロパティの変更 (英語)

区画プロファイルの活動化 (英語)

論理区画における i5/OS™ の再始動とシャットダウン

独立ディスク・プールの静止

i5/OS 高可用性ソリューションでは、独立ディスク・プールを使用して回復力の高いデータおよびアプリケーションを保存します。一部のシステム機能 (バックアップの実行など) は、操作の発生している間そのデータへの変更を一時的に中断することが必要です。

独立ディスク・プールを静止するのにかかる時間を短くするためには、バッチ・ジョブ・キューを保留し、一部のサブシステムを終了するか、またはインタラクティブ・ユーザー宛てに新規作業を後にするように勧める中断メッセージを送信します。

独立ディスク・プールを静止するには、以下のステップを実行します。

コマンド行インターフェースで次のコマンドを入力します。CHGASPACT ASPDEV(name) OPTION(*SUSPEND) SSPTIMO(30) SSPTIMOACN(*CONT), ここで、*name* は中断する独立ディスク・プールの名前です。このコマンドで、ユーザーは 30 秒のタイムアウトを持つ独立ディスク・プールを中断し、タイムアウト制限が超えても次のステップを続けるよう、指定します。

独立ディスク・プールの再開

バックアップ操作のために i5/OS 高可用性環境の独立ディスク・プールを静止した後で、独立ディスク・プールを再開し、静止中にデータに加えられた変更が更新されていることを確認する必要があります。

独立ディスク・プールを再開するには、以下のステップを実行します。

コマンド行インターフェースで次のコマンドを入力します。CHGASPACT ASPDEV(name) OPTION(*RESUME), ここで、name は、再開する独立ディスク・プールの名前です。

サイト間ミラーリングの管理

リモート・ミラーリング、メトロ・ミラー、グローバル・ミラーの、3つのサイト間ミラーリング・テクノロジーを管理することができます。これらのサイト間ミラーリングのテクノロジーでは、実動場所のディスク装置からバックアップ・ロケーションのディスク装置へ重要なデータをコピーすることにより、災害時回復を可能にします。

リモート・ミラーリングの管理

以下の情報を使用して、リモート・ミラーリングの管理に役立ててください。リモート・ミラーリングは、サイト間ミラーリングの副次機能の一種で、i5/OS 環境の独立ディスク・プールにデータをミラーリングするものです。

リモート・ミラーリングの中断:

システムを制限状態に置くなど、何らかの理由で TCP 通信を終了する必要がある場合は、最初にリモート・ミラーリングを中断してください。このアクションにより、高可用性ソリューション内のシステム間でミラーリングが一時的に停止します。

ミラーリングを中断すると、独立ディスク・プール内の実動コピー元に加えられた変更は、ミラー・コピーに伝送されません。

注: リモート・ミラーリングの再開時に、実動コピーとミラー・コピー間で同期が必要となります。トラッキングをしないでリモート・ミラーリングを中断すると、完全同期が行われます。この処理には時間がかかります。

IBM Systems Director Navigator for i5/OS を使用したリモート・ミラーリングの中断

IBM Systems Director Navigator for i5/OS を使用してリモート・ミラーリングを中断するには、以下のステップを行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 中断する実動コピー元の「ディスク・プール」を選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 中断するセッションを選択します。
8. 「アクションの選択」メニューから、「トラッキングありで中断 (Suspend with tracking)」または「トラッキングなしで中断 (Suspend without tracking)」を選択します。

System i ナビゲーターを使用したリモート・ミラーリングの中断

System i ナビゲーターを使用してリモート・ミラーリングを中断するには、以下のステップを行います。

1. System i ナビゲーターで、「ユーザー接続」(またはアクティブな環境)を展開します。
2. 中断対象のリモート・ミラーリングされた実動コピー元のディスク・プールを所有するシステムを展開します。

3. 「構成およびサービス」 → 「ハードウェア」 → 「ディスク装置」 → 「ディスク・プール」を展開します。
4. 中断する実動コピー元の「ディスク・プール」を右クリックし、「リモート・ミラーリング」 → 「リモート・ミラーリングの中断 (Suspend Geographic Mirroring)」を選択します。

トラッキングありで中断する場合は、システムはこれらのディスク・プールに加えられる変更をトラッキングします。これにより、リモート・ミラーリングの再開時に部分的な同期が行われ、同期化処理の長さが短縮します。トラッキング・スペースがいっぱいになると、リモート・ミラーリングの再開時に完全な同期が必要です。

注: 変更をトラッキングなしでリモート・ミラーリングを中断すると、リモート・ミラーリングの再開時に、実動コピーとミラー・コピーの間で完全な同期が必要です。リモート・ミラーリングを中断して変更をトラッキングした場合は、部分的な同期のみで済みます。完全な同期処理には時間がかかります (1 時間から数時間またはそれ以上)。同期にかかる時間の長さは、同期を行うデータの数量、TCP/IP 接続の速度、およびリモート・ミラーリングで使用する回線数により異なります。

リモート・ミラーリングの再開:

リモート・ミラーリングを中断した場合、実動およびミラーリングされたコピー間とのミラーリングを再アクティブ化するために、これを再開する必要があります。

注: リモート・ミラーリングを再開するときに、実動およびミラー・コピーが、リモート・ミラーリングと同時に同期化されます。同期化のプロセスは長時間かかることがあります。ディスク・プールが使用不可になって同期化が中断された場合は、ディスク・プールが再び使用可能になったときに同期化は中断したところから継続します。中断した同期化が継続する際に、最初のメッセージ (CPI0985D) で、同期化が 0% 完了していることが出ます。

IBM Systems Director Navigator for i5/OS によるリモート・ミラーリングの再開

IBM Systems Director Navigator for i5/OS を使用してリモート・ミラーリングを再開するには、以下のステップを実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 再開する「ディスク・プール」の実動コピーを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 再開するセッションを選択します。
8. 「アクションの選択」メニューから、「再開」を選択します。

System i ナビゲーターによるリモート・ミラーリングの再開

System i ナビゲーターを使用してリモート・ミラーリングを再開するには、以下のステップを実行します。

1. System i ナビゲーターで、「ユーザー接続」 (またはアクティブな環境) を展開します。
2. リモート・ミラーリングを再開するディスク・プールの実動コピーを所有するシステムを展開します。
3. 「構成およびサービス」 → 「ハードウェア」 → 「ディスク装置」 → 「ディスク・プール」を展開します。

4. 再開する「ディスク・プール」を右クリックし、「リモート・ミラーリング」 → 「リモート・ミラーリングの再開 (Resume Geographic Mirroring)」を選択します。

DASD 管理操作の開始 (QYASSDMO) API を使用して、ディスク・プールを使用不可にするためにかかる時間を短縮します。

ミラー・コピーの切り離し:

リモート・ミラーリングを使用しているときに、ミラー・コピーにアクセスして保存操作またはデータ・マイングを実行したり、レポートを作成したりするには、実動コピーからミラー・コピーを切り離す必要があります。

ミラー・コピーを切り離すには、ディスク・プールの実動コピーにアクセスします。

注: 切り離したミラー・コピーに再接続するためには、実動コピーとミラー・コピーを完全に同期化させることが必要となります。同期化のプロセスは長時間かかることがあります。

| IBM Systems Director Navigator for i5/OS を使用したミラー・コピーの切り離し

| IBM Systems Director Navigator for i5/OS を使用してミラー・コピーを切り離すには、以下の手順を実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 切り離す「ディスク・プール」の実動コピーを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 切り離すセッションを選択します。
8. 「アクションの選択」メニューから、「トラッキングを使用した切り離し (Detach with tracking)」または「トラッキングを使用しない切り離し (Detach without tracking)」を選択します。

| System i ナビゲーターを使用したミラー・コピーの切り離し

切り離しの実行中に実動コピーが変更されないようにするために、独立ディスク・プールを使用不可にしておくことをお勧めします。

| System i ナビゲーターを使用してミラー・コピーを切り離すには、以下の手順を実行します。

1. System i ナビゲーターで、「ユーザー接続」 (またはアクティブな環境) を展開します。
2. ミラー・コピーを切り離すディスク・プールの実動コピーを所有するシステムを展開します。
3. 「構成およびサービス」 → 「ハードウェア」 → 「ディスク装置」 → 「ディスク・プール」を展開します。
4. 切り離す「ディスク・プール」の実動コピーを右クリックして、「リモート・ミラーリング (Geographic Mirroring)」 → 「ミラー・コピーの切り離し (Detach Mirror Copy)」と選択します

「リモート・ミラーリング (Geographic Mirroring)」 → 「ミラー・コピーの切り離し (Detach Mirror Copy)」が使用不可のためにクリックできない場合、ミラー・コピーは実動コピーと同期していません。この場合は、リモート・ミラーリングを再開して、ディスク・プールをオンに変更し、実動コピーとミラー・コピーを同期化してから、ミラー・コピーを切り離してください。

切り離れたミラー・コピーを使用可能にする前に、実動コピーと区別するために、独立ディスク・プールに固有の装置記述を改めて作成してください。ミラー・コピーに対して装置記述を個別に作成することで、ネットワーク内の同じデータベースに 2 つのインスタンスが存在しないようにします。また、これによって System i ナビゲーター外部での作業も単純化されます。切り離れたミラー・コピーを使用可能にするには、切り離れたミラー・コピーの装置記述を使用します。

ミラー・コピーの再接続:

ミラー・コピーをデタッチし、デタッチしたミラー・コピーの処理を終了した場合は、デタッチしたミラー・コピーを再接続してリモート・ミラーリングの使用を再開する必要があります。

ディスク・プールの実動コピーにアクセスして、デタッチされたミラー・コピーを再接続します。デタッチされたミラー・コピーは、これを実動コピーに再接続したときに使用不可になる必要があります。

注: デタッチされたミラー・コピーに再接続するときには、実動コピーとミラー・コピーとの間で完全な同期化が必要です。同期化のプロセスは長時間かかることがあります。

IBM Systems Director Navigator for i5/OS を使用したミラー・コピーの再接続

IBM Systems Director Navigator for i5/OS を使用してミラー・コピーを再接続するには、以下のステップを実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 中断する実動コピー元の「ディスク・プール」を選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 中断するセッションを選択します。
8. 「アクションの選択」メニューから「接続」を選択します。

System i ナビゲーターを使用したミラー・コピーの再接続

System i ナビゲーターを使用してミラー・コピーを再接続するには、以下のステップを実行します。

1. System i ナビゲーターで、「ユーザー接続」(またはアクティブな環境)を展開します。
2. デタッチされたミラー・コピーを再接続するディスク・プールの実動コピーを所有するシステムを展開します。
3. 「構成およびサービス」 → 「ハードウェア」 → 「ディスク装置」 → 「ディスク・プール」を展開します。
4. 再接続する「ディスク・プール」の実動コピーを右クリックして、「リモート・ミラーリング」 → 「ミラー・コピーの再接続 (Reattach Mirror Copy)」を選択します。

リモート・ミラーリングの構成解除:

特定のディスク・プールまたはディスク・プール・グループに対して、リモート・ミラーリングを使用する機能がなくなった場合は、「リモート・ミラーリングの構成解除 (Deconfigure Geographic Mirroring)」を選択してください。リモート・ミラーリングを構成解除すると、システムはリモート・ミラーリングを停止し、ミラー・コピー・サイトのノード上にある、ディスク・プールのミラー・コピーを削除します。

リモート・ミラーリングを構成解除するには、ディスク・プールをオフラインにする必要があります。

リモート・ミラーリングを構成解除するには、以下の手順で行います。

1. System i ナビゲーターで、「**ユーザー接続**」（またはアクティブな環境）を展開します。
2. 調べたいシステムを展開して、「**構成およびサービス**」 → 「**ハードウェア**」 → 「**ディスク装置**」 → 「**ディスク・プール**」を展開します。
3. 構成解除する「**ディスク・プール**」の実動コピーを右クリックして、「**リモート・ミラーリング (Geographic Mirroring)**」 → 「**リモート・ミラーリングの構成解除 (Deconfigure Geographic Mirroring)**」と選択する。
4. 次の手順で、クラスター構成を更新します。
 - a. 装置クラスター・リソース・グループ (CRG) のリカバリー・ドメインから、ミラー・コピーに関連付けられたノードを除去します。
 - b. クラスター内の残りのノードから、サイト名およびデータ・ポート IP アドレスを除去します。

関連タスク

135 ページの『ノードの除去』

- ノードをアップグレードする場合、またはノードを i5/OS 高可用性環境に参加させる必要がなくなった場合には、クラスターからノードを除去しなくてはならない場合があります。

リモート・ミラーリング・プロパティーの変更:

リモート・ミラーリングに関連する情報を変更したり、関連するコピー記述を編集したりすることができません。

IBM Systems Director Navigator for i5/OS を使用したリモート・ミラーリング・プロパティーの変更

IBM Systems Director Navigator for i5/OS を使用してリモート・ミラーリングのセッションを編集するには、以下の手順を実行します。

1. Web ブラウザーで、http://mysystem:2001 と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「**構成およびサービス**」を選択します。
4. 「**ディスク・プール**」を選択します。
5. セッションに関連付けられたディスク・プールを選択します。
6. 「**アクションの選択**」メニューから、「**セッション**」を選択します。
7. セッションを選択します。
8. 「**アクションの選択**」メニューから、「**プロパティー**」を選択します。関連したコピー記述を変更するには、コピー記述を選択して、「**編集**」を選択します。

System i Navigator を使用したリモート・ミラーリング・プロパティーの変更

System i Navigator を使用してリモート・ミラーリングのセッションを変更するには、以下の手順を実行します。

1. System i ナビゲーターで、「**ユーザー接続**」（またはアクティブな環境）を展開します。
2. 属性を編集するリモート・ミラー・セッションに関連付けられた、リモートでミラーリングされたディスク・プールの実動コピーを所有するシステムを展開して、「**構成およびサービス**」 → 「**ハードウェア**」 → 「**ディスク装置**」 → 「**ディスク・プール**」を展開します。

3. 属性を編集する「ディスク・プール」の実動コピーを右クリックして、「セッション」 → 「開く」と選択します。
4. 属性を編集する「セッション」の実動コピーを右クリックして、「プロパティ」を選択します。関連したコピー記述を変更するには、コピー記述を選択して、「編集」を選択します。

メトロ・ミラー・セッションの管理

IBM System Storage のメトロ・ミラー・テクノロジーを使用する i5/OS 高可用性環境では、メトロ・ミラーを構成された外部ディスク装置と i5/OS システムとの間に、メトロ・ミラーリング・セッションを構成する必要があります。これらのセッションをシステムで管理することができます。

メトロ・ミラー・セッションの中断:

システムのメンテナンスを行うために、メトロ・ミラー・セッションの中断が必要な場合があります。

メトロ・ミラー・セッションを中断するには、以下のステップを行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 中断するディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 中断するセッションを選択します。
8. 「アクションの選択」メニューから、「中断 (Suspend)」を選択します。

メトロ・ミラー・セッションの再開:

システムの保守の実行などのルーチン操作を完了した後で、中断されたメトロ・ミラー・セッションを再開して、高可用性を再び使用できるようにする必要があります。

中断されたメトロ・ミラーリング・セッションを再開するには、以下のステップを実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 中断しているディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 中断しているセッションを選択します。
8. 「アクションの選択」メニューから、「再開」を選択します。

メトロ・ミラー・セッションの削除:

メトロ・ミラー・セッションを削除して、高可用性用および災害時回復用のセッションを使用しないようにすることができます。

メトロ・ミラー・セッションを削除するには、以下の手順を実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。

3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 削除するセッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 削除するセッションを選択します。
8. 「アクションの選択」メニューから、「削除」を選択します。

メトロ・ミラー・プロパティの表示:

メトロ・ミラーリング・セッションについての情報を表示して、関連するコピー記述を変更します。

IBM Systems Director Navigator for i5/OS でメトロ・ミラーリング・プロパティを変更するには、以下の手順を実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. セッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. セッションを選択します。
8. 「アクションの選択」メニューから、「プロパティ」を選択します。関連したコピー記述を変更するには、コピー記述を選択して、「編集」を選択します。

グローバル・ミラーの管理

IBM System Storage のグローバル・ミラー・テクノロジーを使用する i5/OS 高可用性環境では、グローバル・ミラーが構成された外部ディスク装置と i5/OS システムとの間に、グローバル・ミラーリング・セッションを構成する必要があります。これらのセッションをシステムで管理することができます。

グローバル・ミラー・セッションの中断:

システムのメンテナンスを行うために、グローバル・ミラー・セッションの中断が必要な場合があります。

グローバル・ミラー・セッションを中断するには、以下のステップを行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 中断するディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 中断するセッションを選択します。
8. 「アクションの選択」メニューから、「中断 (Suspend)」を選択します。

グローバル・ミラー・セッションの再開:

システムの保守の実行などのルーチン操作を完了した後で、中断されたグローバル・ミラー・セッションを再開して、高可用性を再び使用できるようにする必要があります。

中断されたグローバル・ミラーリング・セッションを再開するには、以下のステップを実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 中断しているディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 中断しているセッションを選択します。
8. 「アクションの選択」メニューから、「再開」を選択します。

グローバル・ミラー・セッションの削除:

グローバル・ミラー・セッションを削除して、高可用性用および災害時回復用のセッションを使用しないようにすることができます。

グローバル・ミラー・セッションを削除するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 削除するセッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 削除するセッションを選択します。
8. 「アクションの選択」メニューから、「削除」を選択します。

グローバル・ミラー・セッション・プロパティーの変更:

グローバル・ミラー・セッションに関する情報を表示し、関連したコピー記述を変更します。

IBM Systems Director Navigator for i5/OS を使用してグローバル・ミラー・プロパティーを変更するには、以下の手順を実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. セッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. セッションを選択します。
8. 「アクションの選択」メニューから、「プロパティー」を選択します。関連したコピー記述を変更するには、コピー記述を選択して、「編集」を選択します。

FlashCopy の管理

FlashCopy は、IBM System Storage テクノロジーの一種で、外部ディスク装置のポイント・イン・タイム・コピーの作成を可能にします。メトロ・ミラーまたはグローバル・ミラーを使用する i5/OS 高可用性

ソリューションにおいて FlashCopy を使用すると、データのコピーをとってメディアにバックアップすることができるため、バックアップ・ウィンドウを短縮することができます。FlashCopy を使用するには、システムと外部ストレージ・ユニットとの間にセッションを作成する必要があります。

FlashCopy セッションの構成

IBM System Storage テクノロジーを使用する i5/OS 高可用性環境では、データのポイント・イン・タイム・コピーを作成するように FlashCopy セッションを構成することができます。

IBM System Storage DS6000 での FlashCopy の使用について詳しくは、IBM System Storage DS6000 Information Center を参照してください。IBM System Storage DS8000 での FlashCopy の使用について詳しくは、IBM System Storage DS8000 Information Center を参照してください。

FlashCopy セッションを構成するには、以下の手順を実行してください。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. ソース・コピーとして使用するディスク・プールを選択します。
6. 「アクションの選択」メニューから、「新規セッション」を選択します。
7. ウィザードの指示に従って、作業を完了します。

FlashCopy の更新

IBM System Storage 外部ストレージ・ユニットで FlashCopy ボリュームの再同期化を実行する際に、FlashCopy セッションを更新することができます。再同期を行うことにより、ボリューム全体を再度コピーすることなく、コピーを作成することができます。この処理は、永続的な関係が存在する場合にのみ可能となります。この関係によって、ストレージ・ユニットは、ソース・ボリュームとターゲット・ボリュームに対する更新を継続的に追跡します。永続的な関係があると、バックグラウンド・コピーが完了した後に、ソース・ボリュームとターゲット・ボリュームの関係が維持されます。i5/OS で作成された FlashCopy セッションでは、IBM System Storage 装置上にある FlashCopy に関係したアクティビティの管理およびモニターを行うことができます。

FlashCopy セッションを更新するには、以下の手順で行います。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 更新するセッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 更新するセッションを選択します。
8. 「アクションの選択」メニューから、「FlashCopy の更新 (Update FlashCopy)」を選択します。

FlashCopy の再接続

FlashCopy セッションを再接続します。

FlashCopy セッションを再接続するには、以下のステップを実行します。

1. Web ブラウザーで、<http://mysystem:2001> と入力します。mysystem はシステムのホスト名です。

2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 再接続するセッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 再接続するセッションを選択します。
8. 「アクションの選択」メニューから、「FlashCopy の再接続 (Reattach FlashCopy)」を選択します。

FlashCopy の切り離し

選択した FlashCopy セッションのソースから、ターゲット・ボリュームを切り離すことができます。

選択した FlashCopy セッションのソースからターゲット・ボリュームを切り離すには、以下の手順を実行します。

1. Web ブラウザーで、http://mysystem:2001 と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 切り離すセッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. ターゲットおよびソース・ボリュームを切り離すセッションを選択します。
8. 「アクションの選択」メニューから、「FlashCopy の切り離し (Detach FlashCopy)」を選択します。

FlashCopy の削除

FlashCopy セッションを削除します。

FlashCopy セッションを削除するには、以下の手順を実行します。

1. Web ブラウザーで、http://mysystem:2001 と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. 削除するセッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. 削除するセッションを選択します。
8. 「アクションの選択」メニューから、「削除」を選択します。

FlashCopy からのデータの復元

FlashCopy が IBM System Storage 装置上で完了した後で、データのソース・コピーの障害に備えて、そのデータをターゲット・ボリュームからソース・ボリュームに復元することができます。これを実行するには、i5/OS で作成した FlashCopy セッションを逆にする必要があります。ただし、セッションを反転すると、ターゲットからソースにまたデータをコピーし直し、ソースを前のバージョンに戻すこととなります。

重要: FlashCopy セッションを反転すると、ソース・コピーに対して行われた変更を、ターゲットのデータをソースに戻すことにより、取り消します。これによって、ソースは前の時点に戻ります。

FlashCopy セッションを反転するには、以下のステップを実行します。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. ソース・コピーのディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッションを開く (Open Sessions)」を選択します。
7. セッションを選択します。
8. 「アクションの選択」メニューから、「FlashCopy の反転 (Reverse FlashCopy)」を選択します。

FlashCopy プロパティーの変更

FlashCopy セッションに関する情報を表示し、関連したコピー記述を変更します。

FlashCopy セッションに関する情報を変更するには、以下の手順で行います。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから、「構成およびサービス」を選択します。
4. 「ディスク・プール」を選択します。
5. セッションに関連付けられたディスク・プールを選択します。
6. 「アクションの選択」メニューから、「セッション」を選択します。
7. セッションを選択します。
8. 「アクションの選択」メニューから、「プロパティー」を選択します。関連したコピー記述を変更するには、コピー記述を選択して、「編集」を選択します。

高可用性ソリューションのトラブルシューティング

i5/OS 高可用性ソリューションの構成後、クラスターやサイト間ミラーリングなど、さまざまなテクノロジーに関する問題に遭遇することがあります。

クラスターのトラブルシューティング

クラスターに特有の問題のエラー回復ソリューションがあります。

クラスターが正常に機能していないと思える状況が生じることがあります。このトピックでは、クラスターに関して生じる可能性のある問題についての情報を示します。

クラスターの問題が存在するかどうかの判別

クラスターに関する問題を診断するには、ここから始めてください。

クラスターが正常に機能していないと思える状況が生じることがあります。問題が存在すると思われる場合、以下のステップを使用して問題が存在するかどうか、および問題の本質を判別することができます。

- システム上でクラスタリングがアクティブかどうかの判別

- 1 クラスター・リソース・サービスがアクティブかどうかを判別するには、システム・ジョブのリスト内で 2 つのジョブ - QCSTCTL および QCSTCRGM - を探してください。これらのジョブがアクティブであれば、クラスター・リソース・サービスもアクティブです。 IBM Director Navigator for i5/OS また

は System i ナビゲーターの実行管理機能を使用してサブシステム内のジョブを表示できますが、WRKACTJOB (活動ジョブ処理) コマンドを使用して同じことを行うこともできます。また、DSPCLUINF (クラスター情報の表示) コマンドを使用して、クラスターの状況情報を表示することもできます。

- 追加のジョブ・クラスター・リソース・サービスもアクティブであることがあります。クラスター・ジョブを使うと、クラスター・リソース・サービス・ジョブがどのようにフォーマットされるかに関する情報が提供されます。

• CPFBB26 メッセージの原因の判別

```
Message . . . . : Cluster Resource Services not active or not responding.  
Cause . . . . . : Cluster Resource Services is either not active or cannot  
respond to this request because a resource is unavailable or damaged.
```

このエラーは、CRG ジョブが非アクティブか、クラスターが非アクティブかのいずれかであることを意味します。DSPCLUINF (クラスター情報の表示) コマンドを使用して、ノードがアクティブかどうかを識別することができます。ノードが非アクティブの場合は、クラスター・ノードを開始します。アクティブの場合は、CRG を確認して CRG に問題がないかどうか判別します。

システム・ジョブ内で CRG ジョブを検索します。IBM Director Navigator for i5/OS または System i ナビゲーターの実行管理機能を使用してサブシステム内のジョブを表示できますが、WRKACTJOB (活動ジョブ処理) コマンドを使用して同じことを行うこともできます。また、DSPCRGINF (CRG情報の表示) コマンドを CRG 名を指定して使用することにより、特定の CRG に関する状況情報を表示することもできます。CRG ジョブが非アクティブの場合は、CRG ジョブ・ログを検索して終了の原因を判別します。問題を修正したら、CHGCLURCY (クラスター・リカバリーの変更) コマンド、またはノード上のクラスターを終了してから再始動して、CRG ジョブを再開します。

• 問題を示しているメッセージを探します。

- F10 を選択して「詳細なメッセージの組み込み」と「詳細なメッセージの除外」を切り替えて、クラスター・コマンドに関連するすべてのメッセージを検討します。すべての詳細メッセージを含めるよう選択し、その他のアクションが必要かどうかを判別します。
- QSYSOPR 内で応答を待っている照会メッセージを探します。
- QSYSOPR 内でクラスター問題を示しているエラー・メッセージを探します。通常、これらは CPFBB00 から CPFBBFF までの範囲にあります。
- 活動記録ログを表示して (DSPLOG CL コマンド)、クラスター問題を示すメッセージを探します。通常、これらは CPFBB00 から CPFBBFF までの範囲にあります。

• クラスター・ジョブのジョブ・ログを調べて、重大エラーを探します。

これらのジョブは最初にロギング・レベルが (4 0 *SECLVL) に設定されているので、必要なエラー・メッセージを見つけることができます。これらのジョブおよび出口プログラム・ジョブのロギング・レベルが適切に設定されていることを確認してください。クラスタリングがアクティブではない場合でも、クラスター・ジョブおよび出口プログラム・ジョブのスパール・ファイルを探することができます。

• ある種のハング状態が疑われる場合、クラスター・ジョブの呼び出しスタックを調べます。

何かの種類 DEQW (デキュー待機) に問題がないかどうかを判別してください。問題がある場合、各スレッドの呼び出しスタックを調べて、いずれかのスレッドの呼び出しスタックに getSpecialMsg が含まれていないかを確認します。

• クラスター垂直ライセンス内部コード (VLIC) のログ項目を調べます。

これらのログ項目には 4800 メジャー・コードがあります。

• NETSTAT コマンドを使用して、通信環境に異常がないかどうかを判別します。

NETSTAT は TCP/IP ネットワーク経路、インターフェース、TCP 接続、およびシステム上の UDP ポートの状態に関する情報を戻します。

- NETSTAT オプション 1 (TCP/IP インターフェース状況の処理) を使用して、クラスタリングに使用するために選択した IP アドレスが「活動中」の状況であることを確認します。さらに、LOOPBACK アドレス (127.0.0.1) も活動中であることを確認します。
- NETSTAT オプション 3 (TCP/IP 接続状況の処理) を使用して、ポート番号を表示します (F14)。ローカル・ポート 5550 は「接続待機」状況となっているはずですが、このポートは、STRTCPSVR *INETD コマンドを介して開く必要があります。このことは、QTOGINTD (ユーザー QTCP) ジョブがアクティブのジョブ・リスト内に存在することによって証明されます。クラスタリングがノード上で開始した場合、ローカル・ポート 5551 は開いていて、「*UDP」状況でなければなりません。クラスタリングが開始していない場合、ポート 5551 は閉じている必要があります。これが開いていると、対象ノードでクラスタリングが正常に開始できなくなります。
- ping を使用します。クラスター・ノードの開始を試みたときにそのノードを ping できない場合、内部クラスタリング・エラー (CPFBB46) が表示されます。

クラスタに関する回復情報の収集

クラスタの処理 (WRKCLU) コマンドを使って、クラスタの全体像に関する情報を収集することができます。その情報は、エラーの解決で活用することができます。

クラスタの処理 (WRKCLU) コマンドは、クラスタのノードとオブジェクトの表示と処理に使用します。このコマンドを実行すると、「クラスタの処理」画面が表示されます。クラスタ内のノードとクラスタ情報の表示以外に、クラスタ情報を表示してクラスタに関するデータを収集するのにも、このコマンドを使用することができます。

エラー回復情報を収集するには、次のようなステップを行います。

1. 文字ベース・インターフェースで、WRKCLU OPTION(OPTION) と入力します。次のようなオプションを指定して、どのクラスタ状況情報を処理したいかを指示することができます。

*SELECT

「クラスタの処理」メニューを表示します。

*CLUINF

クラスタ情報を表示します。

*CFG クラスタの性能および構成パラメータを表示します。

*NODE

クラスタ内のノードのリストを示した「クラスタ・ノードの処理」パネルを表示します。

*DEVDMN

クラスタ内の装置ドメインのリストを示した「装置ドメインの処理」パネルを表示します。

*CRG クラスタ内のクラスタ・リソース・グループのリストを示した「クラスタ・リソース・グループの処理」パネルを表示します。

*ADMDMN

クラスタ内の管理可能ドメインのリストを示した「管理可能ドメインの処理」パネルを表示します。

*SERVICE

クラスタ内のすべてのクラスタ・リソース・サービス・ジョブに関連したトレースおよびデバッグの情報を収集します。この情報は、各クラスタ・リソース・サービス・ジョブごとにメンバーと一緒にファイルに書き込まれます。このオプションは、サービス・プロバイダー

から指定された場合のみ使用してください。これは、クラスター・トレースのダンプ (DMPCLUTRC) のプロンプト・パネルを表示します。

クラスタの一般的な問題

ここでは、クラスターで生じる可能性のある最も一般的な問題のいくつかをリストして、その回避方法および回復方法を示します。

以下の共通問題は容易に回避可能であるか、または容易に修正可能です。

クラスター・ノードを始動または再始動できない。

この状況は通常、通信環境で問題が生じていることに原因があります。この状況を回避するには、ループバック・アドレス、INETD 設定値、ALWADDCLU 属性、およびクラスター通信の IP アドレスを含むネットワーク属性が正しく設定されていることを確認してください。

- リモート・ノードを始動しようとする場合、ALWADDCLU ネットワーク属性がターゲット・ノードに正しく設定されている必要があります。この属性は環境に応じて、*ANY または *RQSAUT のいずれかに設定する必要があります。
- ローカル上およびターゲット・ノード上でクラスタリングに使用するために選択した IP アドレスは、「活動中」の状況を示していなければなりません。
- ローカル上およびターゲット・ノード上の LOOPBACK アドレス (127.0.0.1) も、活動中でなければなりません。
- ローカル・ノードおよびすべてのリモート・ノードは、ネットワーク・ルーティングがアクティブであることを確認するために、クラスタリングに使用する IP アドレスを使用しての PING が可能でなければなりません。
- ターゲット・ノード上で INETD がアクティブである必要があります。INETD がアクティブなら、ターゲット・ノード上のポート 5550 は「接続待機」状況になっているはずですが、INETD サーバーの開始方法については、『INETD サーバー』の項を参照してください。
- ノードを始動しようとする前は、始動するノード上のポート 5551 は閉じている必要があります。これが開いていると、対象ノードでクラスタリングが正常に開始できなくなります。

結合されていない複数の単一ノード・クラスターが生じた。

これは始動するノードが他のクラスター・ノードと通信できない場合に生じることがあります。通信パスを調べてください。

出口プログラムからの応答が遅い。

この状況が生じる一般的な原因は、出口プログラムで使用されるジョブ記述の設定が正しくないことです。MAXACT パラメーターの設定値が低すぎるために、たとえば任意の時点でアクティブの出口プログラムのインスタンスが 1 つだけに制限されることがあります。この値は *NOMAX に設定することをお勧めします。

一般的なパフォーマンスが遅い。

この症状については、いくつかの一般的な原因があります。

- 最も可能性の高い原因は、共用通信回線でのトラフィックが大きいことです。
- 可能性のある別の原因は、通信環境とクラスター・メッセージ調整パラメーターとの間に不整合があることです。クラスター・リソース・サービス情報検索 (QcstRetrieveCRSInfo) API を使用して調整パラメーターの現行設定値を表示し、クラスター・リソース・サービス変更

(QcstChgClusterResourceServices) API を使用して設定値を変更することができます。旧式のアダプター・ハードウェアを使用している場合、デフォルトのクラスター調整パラメーター設定値では、クラスターのパフォーマンスが低下することがあります。「旧式」の定義に含まれるアダプター・ハードウェアのタイプには、2617、2618、2619、2626、および 2665 があります。この場合、「パフォーマンス・クラス」調整パラメーターの設定値を「Normal」にすることをお勧めします。

- この状況を生じる別の一般的な原因は、IP マルチキャスト・グループに問題があることです。複数のノードのプライマリー・クラスター・アドレス (クラスターの作成時またはノードの追加時に指定のノードに入力された最初のアドレス) が共通 LAN 上に存在する場合、クラスターは IP マルチキャスト機能を利用します。NETSTAT コマンドを使用して、プライマリー・クラスター・アドレスがマルチキャスト・ホスト・グループの 226.5.5.5 を示していることを確認してください。これは、対象アドレスに対してオプション 14 「マルチキャスト・グループの表示」を使用することによって表示できます。マルチキャスト・グループが存在しない場合、クラスター・リソース・サービス情報検索 (QcstRetrieveCRSInfo) API を使用して、*Enable multicast* クラスター調整パラメーターがデフォルトの設定値「TRUE」に設定されたままであることを確認してください。
- クラスターのノードすべてがローカル LAN 上にあるか、またはネットワーク経路全体をとおして 1,464 バイトよりも大きい最大転送単位 (MTU) のパケット・サイズを処理できるルーティング機能を持っている場合、*Message fragment size* のクラスター調整パラメーター値を、経路 MTU にもっと適した値に増加すれば、大きなクラスター・メッセージ転送 (1,536 KB よりも大きい) を大幅に高速化することができます。

新しいリリースの機能をいずれも使用できない。

新リリースの機能を使用しようとしたときにエラー・メッセージ CPFBB70 が表示された場合、クラスター・バージョンは以前のバージョン・レベルに設定されたままであるということです。すべてのクラスター・ノードを新規のリリース・レベルにアップグレードしてから、クラスター・バージョンの調整インターフェースを使用して現行のクラスター・バージョンを新規のレベルに設定する必要があります。詳細は、『クラスターのクラスター・バージョンの調整』の項を参照してください。

ノードを装置ドメインに追加することも、System i ナビゲーター・クラスター管理インターフェースにアクセスすることもできない。

System i ナビゲーター・クラスター管理インターフェースへのアクセスや切り替え可能な装置の使用を可能にするには、システムに i5/OS オプション 41 (HA 切り替え可能リソース) をインストールする必要があります。さらに、そのオプションのための有効なライセンス・キーも必要です。

クラスター PTF を適用したが機能しているように見えない。

PTF を適用した後、以下のタスクを実行したことを確認してください。

1. クラスターを終了する
2. サインオフしてからサインオンする

以前のプログラムは活動化グループが破棄されるまでアクティブのままです。すべてのクラスター・コード (クラスター API を含む) は、デフォルトの活動化グループで実行します。

3. クラスターを開始する

ほとんどのクラスター PTF の場合、PTF を活動化するには、ノード上でクラスタリングをいったん終了してから再始動する必要があります。

出口プログラムのジョブ・ログに CEE0200 がある。

このエラー・メッセージで、呼び出し元モジュールは QLEPM、呼び出し元プロシージャは Q_LE_leBdyPeilog です。出口プログラムが呼び出すプログラムは、*CALLER または名前の指定された活動化グループ内で実行しなければなりません。この状態を修正するには、出口プログラムまたはエラーの生じたプログラムを変更しなければなりません。

クラスター・リソース・サービスのジョブ・ログに CPD000D とその後続く CPF0001 とがある。

このエラー・メッセージが表示されたら、QMLTTHDACN のシステム値が 1 または 2 に設定されていることを確認してください。

クラスターがハングしたと思われる。

クラスター・リソース・グループの出口プログラムが未解決であることを確認してください。出口プログラムを検査するには、WRKACTJOB ((活動ジョブ処理) コマンドを使用して、「機能」列に PGM-QCSTCRGEXT があるかどうかを確かめます。

区画エラー

クラスター状態によっては、簡単に修正することができます。クラスター区画が生じた場合の回復方法を確認してください。このトピックでは、クラスター区画の回避方法を示し、区画をマージして元に戻す方法を例示します。

クラスター内でクラスター区画が発生するのは、クラスター内にある 1 つ以上のノードどうしの連絡が途絶えたときに、その不通のノードの障害を確認できない場合です。これを、論理区画 (LPAR) 環境の区画と混同しないでください。

活動記録ログ (QHST) または QCSTCTL ジョブ・ログにエラー・メッセージ CPFBB20 を受け取った場合、クラスターが区画に分割されているので、その回復方法を知る必要があります。以下の例は、4 つのノード (A、B、C、および D) から成るクラスターを含むクラスター区画を示しています。この例では、クラスター・ノード B および C の間で通信が停止したので、そのクラスターが 2 つのクラスター区画に分割された場合を示しています。クラスターが区画に分割される前に、4 つのクラスター・リソース・グループが存在していました。それらは任意のタイプであることが可能で、CRG A、CRG B、CRG C、および CRG D と呼ばれます。この例は、クラスター・リソース・グループごとのリカバリー・ドメインを示しています。

表 35. クラスター区画におけるリカバリー・ドメインの例

ノード A	ノード B	x	ノード C	ノード D
CRG A (バックアップ 1)	CRG A (プライマリー)			
	CRG B (プライマリー)		CRG B (バックアップ 1)	
	CRG C (プライマリー)		CRG C (バックアップ 1)	CRG C (バックアップ 2)
CRG D (バックアップ 2)	CRG D (プライマリー)		CRG D (バックアップ 1)	
区画 1			区画 2	

通信パス内のいずれかの場所で最大転送単位 (MTU) がクラスター通信の調整可能パラメーター、メッセージ・フラグメント・サイズよりも小さくなると、クラスターは区画に分割されることがあります。クラスター IP アドレスの MTU を検証するには、対象ノードで TCP/IP ネットワーク状況の処理 (WRKTCPSTS) コマンドを使用します。通信パス全体の各ステップでも、MTU を検査しなければなりません。

ん。MTU がメッセージ・フラグメント・サイズよりも小さい場合、パスの MTU を増加させるか、またはメッセージ・フラグメント・サイズを小さくします。クラスター・リソース・サービス情報検索 (QcstRetrieveCRSInfo) API を使用して調整パラメーターの現行設定値を表示し、クラスター・リソース・サービス変更 (QcstChgClusterResourceServices) API を使用して設定値を変更することができます。

クラスターが区画に分割された状態にある原因が修正された後、クラスターは再確立された通信リンクを検出して、メッセージ CPFBB21 を活動記録ログ (QHST) または QCSTCTL ジョブ・ログに出します。これにより、オペレーターはクラスターがクラスター区画の状態から回復したことを知ります。クラスターが区画に分割された状態が修正されると、数分後にクラスターが互いにマージして元に戻る可能性があることに注意してください。

1 次クラスター区画と 2 次クラスター区画の判別:

クラスター区画内で実行可能なクラスター・リソース・グループ・アクションのタイプを判別するには、その区画が 1 次または 2 次のどちらのクラスター区画かが分からなければなりません。区画が検出された場合には、クラスターで定義されている各クラスター・リソース・グループごとに、各区画は 1 次または 2 次のどちらかの区画に指定されます。

プライマリー・バックアップ・モデルの場合、現在のノード役割がプライマリーであるノードが 1 次区画に収容されます。他の区画はすべて 2 次になります。1 次区画は、すべてのクラスター・リソース・グループで同じとは限りません。

対等モデルには、次のような区画規則があります。

- リカバリー・ドメイン・ノードがすべて 1 つの区画に含まれる場合、その区画が 1 次区画になります。
- リカバリー・ドメイン・ノードが区画を超える場合、1 次区画なしになります。どちらの区画も 2 次区画になります。
- クラスター・リソース・グループがアクティブになっていて、特定の区画内に対等ノードがない場合、そのクラスター・リソース・グループは、その区画内で終了します。
- 2 次区画内で操作上の変更が許可されるのは、そのような操作上の変更に対する制約事項が満足された場合だけです。
- 2 次区画内では、構成上の変更は許可されません。

クラスター・リソース・グループ API ごとに、以下の制限があります。

表 36. クラスター・リソース・グループ API 区画の制約事項

クラスター・リソース・グループ API	1 次区画で許可される	2 次区画で許可される
リカバリー・ドメインへのノードの追加	X	
CRG 装置項目の追加		
クラスター・リソース・グループの変更	X	
CRG 装置項目の変更	X	X
クラスター・リソース・グループの作成		
クラスター・リソース・グループの削除	X	X
情報の配布	X	X
クラスター・リソース・グループの終了 ¹	X	
切り替えの開始	X	
クラスター・リソース・グループのリスト	X	X

表 36. クラスター・リソース・グループ API 区画の制約事項 (続き)

クラスター・リソース・グループ API	1 次区画で許可される	2 次区画で許可される
クラスター・リソース・グループ情報の一覧表示	X	X
リカバリー・ドメインからのノードの除去	X	
CRG 装置項目の除去	X	
クラスター・リソース・グループの開始 ¹	X	
注:		
1. 対等クラスター・リソース・グループの場合はすべての区画で許可されていますが、その影響を受けるのは API を実行中の区画だけです。		

これらの制限の適用によって、クラスターの区画化が解消されたときに、クラスター・リソース・グループを同期することができます。区画化された状態からノードがクラスターに再結合するとき、1 次区画内のクラスター・リソース・グループのバージョンが 2 次区画のノードにコピーされます。

対等モデルの 2 つの 2 次区画をマージすると、アクティブの状況のクラスター・リソース・グループを収容している区画が優先されることが宣言されます。クラスター・リソース・グループの両方の区画が同じ状況である場合、クラスター・リソース・グループのリカバリー・ドメイン内に最初にリストされているノードを収容している区画が優先されることが宣言されます。優先区画内のクラスター・リソース・グループのバージョンが、もう一方の区画内のノードにコピーされます。

区画が検出されると、クラスター・ノード項目追加、クラスター・バージョンの調整、およびクラスター作成 API はどの区画内でも実行できません。装置ドメイン項目追加 API を実行できるのは、装置ドメイン内のどのノードも区画に分割されていない場合だけです。他のすべてのクラスター制御 API は、区画内で実行できます。しかし、API が実行するアクションの影響はその API を実行している区画内に限定されません。

区画ノードの失敗への変更:

時々、「区画」状況が報告されているときに、実際にはノード障害が生じていることがあります。このことは、クラスター・リソース・サービスが 1 つ以上のノードとの通信を失ったものの、ノードが引き続き作動可能かどうかを検出できない場合に生じます。この状況が生じた場合に、ノードに障害が起きたことを知らせるための簡単なメカニズムがあります。

重要: ノードに障害が起きたことをクラスター・リソース・サービスに知らせると、区画状況からの回復を行うのがより簡単になります。ただし、実際にはそのノードが引き続きアクティブであり、本当の区画が発生している場合には、ノード状況を障害に変更してはなりません。そうに変更すると、複数の区画に属するノードが、クラスター・リソース・グループのためのプライマリーの役割をするものと見なされてしまいます。2 つのノードがいずれも自分をプライマリー・ノードと認識してしまうと、複数のノードがそれぞれ独自にファイルのコピーに変更を加えた場合、ファイルやデータベースなどのデータがばらばらになったり破壊されたりすることがあります。さらに、各区画内のそれぞれ 1 つのノードがプライマリーの役割を割り当てられていた場合、この 2 つの区画をマージすることはできません。

ノードの状況が「失敗」に変更されると、区画内のそれぞれのクラスター・リソース・グループのリカバリー・ドメイン内のノードの役割が、配列し直されることがあります。「失敗」に設定されるノードは、最後のバックアップとして割り当てられます。複数のノードに障害が起き、それらの状況を変更する必要がある場合に、ノードを変更する順序が、リカバリー・ドメインのバックアップ・ノードの最終的な順序に影響を与えます。障害が起きたノードが CRG のプライマリー・ノードであった場合には、最初のアクティブのバックアップが、新規プライマリー・ノードとして再割り当てされます。

クラスター・リソース・サービスが、ノードとの通信が途切れたときに、そのノードがまだ作動可能かどうかを検出できない場合には、クラスター・ノードの状況は「**通信中でない**」になります。ノードの状況を「**通信中でない**」から「**失敗**」に変更する必要があるかもしれません。その後、ノードを再始動することができます。

ノードの状況を「**通信中でない**」から「**失敗**」に変更するには、以下のようにします。

1. Web ブラウザーで、`http://mysystem:2001` と入力します。mysystem はシステムのホスト名です。
2. ご使用のユーザー・プロファイルとパスワードでシステムにログオンします。
3. IBM Systems Director Navigator for i5/OS のウィンドウから「**クラスター・リソース・サービス**」を選択します。
4. 「**クラスター・リソース・サービス**」ページで、「**クラスター・ノードの処理**」タスクを選択して、クラスター内のノードのリストを表示します。
5. 「**アクションの選択**」メニューをクリックして、「**状況の変更**」をクリックします。ノードの状況を失敗に変更します。

関連情報

クラスター・ノードの変更 (CHGCLUNODE) コマンド

クラスター・ノード項目の変更 (QcstChangeClusterNodeEntry) API (英語)

区画化されたクラスター管理可能ドメイン:

区画化されたクラスター管理可能ドメインを処理する場合には、次の情報を考慮してください。

クラスター管理可能ドメインで区画が発生している場合、各区画にあるアクティブ・ノード間で引き続き変更点が同期化されます。ノードが再度マージされると、クラスター管理可能ドメインにより区画ごとのすべての変更が伝搬されます。こうすることにより、リソースとアクティブ・ドメインとの整合性が保たれます。クラスター管理可能ドメインのマージ処理に関しては、いくつかの考慮事項があります。

- すべての区画がアクティブであり、別の区画にある同じリソースに変更が加えられた場合は、マージ中にすべてのノードに対して最も新しい変更が適用されます。最も新しい変更は、変更が開始された各ノードの協定世界時 (UTC) により判別されます。
- すべての区画が非アクティブであった場合は、各リソースのグローバル値は、最後に区画がアクティブだった際の変更に基づいて解決されます。モニター対象リソースに対するこれらの変更の実際の適用は、クラスター管理可能ドメインを表す対等 CRG が開始されるまで行われません。
- マージ前に一部の区画がアクティブで、一部の区画は非アクティブであった場合は、アクティブ区画で行われた変更を表すグローバル値が非アクティブ区画に伝搬されます。次に非アクティブ区画が開始され、非アクティブ区画のノードに対して行われた処理中のすべての変更が、マージ対象ドメインに伝搬されることとなります。

ヒント: クラスター区画:

クラスター区画に対しては、以下のヒントを参考にしてください。

1. 区画のマージを可能にするために、区画内の操作を制限するための規則が設けられています。このような制限がなければ、クラスターの再構成には、かなりの量の作業が必要となります。
2. 1 次区画内のノードが破棄された場合、2 次区画内で特殊な処理が必要となることがあります。そのような事態が発生する原因となる最も一般的なケースは、1 次区画を構成するサイトの消失です。『区画エラーからの回復』の中の例を使用する一方で、区画 1 が破棄されたと仮定します。この場合、クラスター・リソース・グループ B、C、および D のプライマリー・ノードは、区画 2 になければなりません。最も簡単な回復は、「クラスター・ノード項目の変更」を使用してノード A とノード B の両方を

障害に設定することです。その方法の詳細は、『区画化ノードを障害ノードに変更する』を参照してください。回復は、手動で行うこともできます。そのためには、以下の操作を実行します。

- a. ノード A および B を、区画 2 のクラスターから除去します。区画 2 が現在のクラスターです。
- b. 新しいクラスターに必要な任意の論理複製環境を確立します。つまり、クラスター・リソース・グループ開始 API/CL コマンドなどです。

区画 2 内のクラスター定義からノードが除去されるので、区画 1 および区画 2 のマージ試行が失敗します。クラスター定義内のミスマッチを訂正するために、区画 1 内のそれぞれのノードに対してクラスター削除 (QcstDeleteCluster) API を実行します。その後、区画 1 からクラスターにノードを追加し、すべてのクラスター・リソース・グループ定義、リカバリー・ドメイン、および論理複製を再設定します。このことは、大量の作業を必要とし、エラーも発生しがちです。それで、この手順は、サイトがなくなった状況でのみ行うべきです。

3. ノード開始操作の処理は、開始するノードの状況によって異なります。

ノードに障害が起きたか、またはノード終了操作がノードを終了しました。

- a. クラスター・リソース・サービスが、追加されるノードで開始されます。
- b. クラスター定義が、クラスター内のアクティブ・ノードから、開始されるノードにコピーされます。
- c. リカバリー・ドメイン内で開始されるノードを持つクラスター・リソース・グループが、クラスター内のアクティブ・ノードから、開始されるノードにコピーされます。クラスター・リソース・グループは、開始されるノードから、クラスター内のアクティブ・ノードにコピーされません。

ノードが区画化ノードである場合:

- a. アクティブ・ノードのクラスター定義が、開始されるノードのクラスター定義と比較されます。定義が同じであれば、マージ操作として開始が継続されます。定義が一致しない場合には、マージが停止し、ユーザーの介入が必要となります。
- b. マージが継続する場合、開始されるノードがアクティブ状況に設定されます。
- c. リカバリー・ドメイン内で開始されるノードを持つクラスター・リソース・グループが、クラスター・リソース・グループの 1 次区画から、クラスター・リソース・グループの 2 次区画にコピーされます。クラスター・リソース・グループは、開始されるノードから、クラスター内ですでにアクティブとなっているノードにコピーされることがあります。

クラスターの回復

生じる可能性のある他のクラスター障害から回復する方法について示します。

クラスター・ジョブ障害からの回復:

クラスター・リソース・サービス・ジョブの障害は、通常、何か他の問題があることを示しています。

障害が起きたジョブに関連したジョブ・ログを調べて、なぜ障害が起きたのかを記述するメッセージを探します。エラー状態を訂正します。

クラスター回復変更 (CHGCLURCY) コマンド を使って、ノード上でクラスタリングの終了と再始動を行う必要なしに終了したクラスター・リソース・グループ・ジョブを再始動することができます。

1. CHGCLURCY CLUSTER(EXAMPLE)CRG(CRG1)NODE(NODE1)ACTION(*STRCRGJOB) このコマンドは、ノード NODE1 上のクラスター・リソース・グループのジョブ CRG1 が投入される原因になります。NODE1 上でクラスター・リソース・グループのジョブを開始するには、NODE1 上でクラスタリングがアクティブになっている必要があります。
2. ノードでクラスタリングを再始動します。

IBM ビジネス・パートナー・クラスター管理プロダクトを使用している場合は、プロダクトに付属している資料を参照してください。

関連情報

クラスター回復の変更 (CHGCLURCY) コマンド

損傷を受けたクラスター・オブジェクトの回復:

オブジェクトが損傷を受けることはあまりありませんが、クラスター・リソース・サービス・オブジェクトが損傷を受ける可能性はあります。

それがアクティブ・ノードである場合、システムは、クラスター内の別の回復を試行します。システムは次の回復ステップを実行します。

内部オブジェクトが損傷を受けた場合

1. 損傷を受けたノードが終了します。
2. クラスター内に別のアクティブ・ノードが最低 1 つある場合には、損傷を受けたノードが自動的に自身を再始動し、クラスターに再結合します。再結合のプロセスで、損傷を受けた状態が修正されます。

クラスター・リソース・グループが損傷を受けた場合

1. 損傷を受けた CRG を持つノードが、その CRG に関連した、現在処理中の操作に失敗します。システムは、別のアクティブ・ノードからの CRG の自動回復を試行します。
2. リカバリー・ドメイン内にアクティブなメンバーが最低 1 つある場合には、CRG 回復が作動します。ない場合には、CRG ジョブが終了します。

システムが他のアクティブ・ノードを識別したり、それに到達したりできない場合には、以下の回復ステップを実行する必要があります。

内部オブジェクトが損傷を受けた場合

内部クラスタリング・エラーを受け取ります (CPFBB46、CPFBB47、または CPFBB48)。

1. 損傷を含むノードのクラスタリングを終了します。
2. 損傷を含むノードのクラスタリングを再始動します。このことを、クラスター内の別のアクティブ・ノードから実行します。
3. ステップ 1 および 2 で問題が解決しない場合には、損傷を受けたノードをクラスターから除去します。
4. クラスターおよび適切なクラスター・リソース・グループのリカバリー・ドメインに、システムを戻します。

クラスター・リソース・グループが損傷を受けた場合

オブジェクトが損傷を受けたことを知らせるエラーを受け取ります (CPF9804)。

1. 損傷を受けたクラスター・リソース・グループを含むノードで、クラスタリングを終了します。
2. DLTCRG コマンドを使用して CRG を削除します。
3. CRG オブジェクトを含むクラスター内にアクティブなノードが他にない場合には、メディアから復元します。
4. 損傷を受けたクラスター・リソース・グループを含むノードで、クラスタリングを開始します。これは、どのアクティブ・ノードからでも実行できます。

5. クラスタリングを開始すると、システムはすべてのクラスター・リソース・グループを再同期します。クラスター内の他のノードに CRG が含まれていない場合には、CRG を再作成する必要があるかもしれません。

完全なシステム消失後のクラスターの回復:

システムでの想定外の電力喪失が原因で完全なシステム消失が起きた場合、ここに記載されている情報と「システムの回復」トピック内の該当するチェックリストを一緒に使用して、システム全体を回復してください。

シナリオ 1: 同じシステムへの復元

1. ライセンス内部コードと i5/OS との間で、装置ドメイン情報に不整合が生じるのを防ぐため、オプション 3 (ライセンス内部コードの導入および構成の回復) を使用して、ライセンス内部コードをインストールするようお勧めします。

注: 「ライセンス内部コードの導入および構成の回復」操作が成功するためには、同じディスク装置を持っている必要があります。ただし、ロード・ソース・ディスク装置に障害が起きた場合は例外です。また、同じリリースを回復しなければなりません。

2. ライセンス内部コードをインストールした後、「システムの回復」トピック中の『ディスク構成の復元』の手順に従ってください。このステップによって、ディスク・プールを再構成しなくて済むようになります。
3. システム情報を回復し、回復したばかりのノードでクラスタリングを開始する用意ができたなら、アクティブ・ノードからクラスタリングを開始する必要があります。このことにより、最新の構成情報が、回復したノードに伝搬されます。

シナリオ 2: 別のシステムへの復元

システム情報を回復し、ジョブ・ログをチェックして、すべてのオブジェクトが復元されたことを確認した後、次のステップを実行して、正確なクラスター装置のドメイン構成を取得する必要があります。

1. 復元したばかりのノードから、クラスターを削除します。
2. アクティブ・ノードから、以下の手順を実行します。
 - a. 回復されたノードをクラスターから除去します。
 - b. 回復されたノードをクラスターに戻します。
 - c. 回復されたノードを装置ドメインに戻します。
 - d. クラスター・リソース・グループを作成するか、またはリカバリー・ドメインにノードを追加します。

災害後のクラスターの回復:

災害が起きてすべてのノードが失われた場合、クラスターを再構成する必要があります。

そのようなシナリオに備えるために、クラスター構成情報を保管して、その情報のハードコピー印刷出力を取っておくことをお勧めします。

バックアップ・テープからのクラスターの復元:

通常の操作時には、バックアップ・テープから復元する必要が生じることはありません。

これが必要なのは、災害が起きてクラスター内のすべてのノードが消失した場合のみです。災害が発生した場合には、バックアップ戦略および回復戦略の作成後に定めた一般回復手順に従って回復を図ります。

サイト間ミラーリングのトラブルシューティング

最初の段落と要約で使用する簡略説明をここに記載します。

ユーザーのテキストをここに入力します。

- 興味のある点
- その他の興味のある点

副見出し

コンセプト用の小さなセクションです。

例

コンセプト用の例の小さなセクションです。

リモート・ミラーリング・メッセージ

リモート・ミラーリング・メッセージの説明およびリカバリーを検討して、リモート・ミラーリングの問題を解決します。

0x00010259

説明: システムがミラー・コピーを検索できなかったため、操作が失敗しました。

リカバリー: デバイス・ドメイン内のノードで応答しなかったものがあります。クラスタリングがアクティブであることを確認してください。必要に応じて、ノードでクラスタを開始します。詳しくは、103 ページの『ノードの開始』を参照してください。要求を再試行してください。問題が解決しない場合には、技術サポートの提供者に連絡してください。

0x0001025A

説明: ディスク・プール・グループ内のディスク・プールで、リモート・ミラーリングが行われていないものがあります。

リカバリー: ディスク・プール・グループ内のあるディスク・プールがリモート・ミラーリングされていれば、ディスク・プール・グループ内のすべてのディスク・プールがリモート・ミラーリングされていなければなりません。次のアクションのいずれかを実行してください。

1. リモート・ミラーリングされていないディスク・プールに対して、リモート・ミラーリングを構成する。
2. リモート・ミラーリングされたディスク・プールに対して、リモート・ミラーリングを構成解除する。

0x00010265

説明: 切り離されたミラーリング済みコピーが使用可能になっています。

リカバリー: 切り離されたミラーリング済みコピーを使用不可にして、再接続操作をもう一度行ってください。

0x00010380

説明: ミラー・コピーの構成から、ディスク装置が欠落しています。

リカバリー: 欠落しているディスクを検索するか、またはミラー・コピー内で修正します。宛先ノードで製品アクティビティ・ログを確認してください。IOP キャッシュ・ストレージを再利用してください。

0x00011210

説明: ディスク・プール・グループの提案された 2 次ディスク・プールは、リモート・ミラーリングされていません。

リカバリー: ディスク・プール・グループ内のあるディスク・プールがリモート・ミラーリングされていれば、ディスク・プール・グループ内のすべてのディスク・プールがリモート・ミラーリングされていなければなりません。この時点で、またはこの操作が完了した後もリモート・ミラーリングされていない、提案された 2 次ディスク・プールに対して、リモート・ミラーリングを構成する必要があります。

0x00011211

説明: 重複するミラー・コピーが存在します。

リカバリー: ローカルでミラーリングされたディスク装置、Enterprise Storage Server® FlashCopy、またはバックレベルの独立ディスク・プール・コピーのうち、2 つのシステム上に存在する可能性があるものを確認します。詳しくは、ミラー・コピー・ノードのプロダクト活動ログを参照してください。重複を除去して、要求を再試行してください。問題が解決しない場合には、技術サポートの提供者に連絡してください。IBM のサポートおよびサービスが必要な場合は、i5/OS テクニカル・サポートを参照してください。

コードに関するライセンス情報および特記事項

IBM は、お客様に、すべてのプログラム・コードのサンプルを使用することができる非独占的な著作使用权を許諾します。お客様は、このサンプル・コードから、お客様独自の特別のニーズに合わせた類似のプログラムを作成することができます。

強行法規で除外を禁止されている場合を除き、IBM、そのプログラム開発者、および供給者は「プログラム」および「プログラム」に対する技術的サポートがある場合にはその技術的サポートについて、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。

いかなる場合においても、IBM および IBM のサプライヤーならびに IBM ビジネス・パートナーは、その予見の有無を問わず発生した以下のものについて賠償責任を負いません。

1. データの喪失、または損傷。
2. 直接損害、特別損害、付随的損害、間接損害、または経済上の結果的損害
3. 逸失した利益、ビジネス上の収益、あるいは節約すべかりし費用

国または地域によっては、法律の強行規定により、上記の責任の制限が適用されない場合があります。

IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラムのインストール

i5/OS 高可用性ソリューションをインプリメントする前に、IBM System i High Availability Solutions Manager (iHASM) ライセンス・プログラム (5761-HAS) を、高可用性に属する各システム上にインストールする必要があります。

iHASM ライセンス・プログラムをインストールするには、以下のインストール要件を満たしておく必要があります。

1. V6R1 i5/OS のインストールまたはアップグレードを行います。
2. i5/OS オプション 41 (HA 切り替え可能リソース) をインストールします。

- | iHASM ライセンス・プログラムをインストールするには、以下の手順を実行します。
- | 1. コマンド行から GO LICPGM と入力します。
 - | 2. 「ライセンス・プログラムの処理」画面で、オプション 11 (ライセンス・プログラムのインストール) を選択します。
 - | 3. 製品 5761-HAS、オプション *BASE を選択し、System i 用の iHASM 製品をインストールします。 Enter を押します。
 - | 4. 「インストール・オプション」画面で、指示に従ってインストール装置の名前を入力します。 Enter を押して、インストールを開始します。

付録. 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-8711
東京都港区六本木 3-2-12
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation
Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

- 1 本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム
- 1 契約の契約条項、IBM プログラムのご使用条件、IBM 機械コードのご使用条件、またはそれと同等の条項
- 1 に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

表示されている IBM の価格は IBM が小売り価格として提示しているもので、現行価格であり、通知なしに変更されるものです。卸価格は、異なる場合があります。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. _年を入れる_. All rights reserved.

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

プログラミング・インターフェース情報

本書には、プログラムを作成するユーザーが IBM i5/OS のサービスを使用するためのプログラミング・インターフェースが記述されています。

商標

以下は、International Business Machines Corporation の米国およびその他の国における商標です。

i5/OS

IBM

IBM (ロゴ)

System i

System i5

IBM Systems Storage

TotalStorage

FlashCopy

- | Adobe、Adobe ロゴ、PostScript、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。
- | Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。
- | Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

使用条件

これらの資料は、以下の条件に同意していただける場合に限りご使用いただけます。

個人使用: これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布 (頒布、送信を含む) または表示 (上映を含む) することはできません。

商業的使用: これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任なしで提供されます。



Printed in Japan