



# コンテンツ レプリケーション の設定

この章では、オンデマンド コンテンツ レプリケーション、およびコンテンツのステージングとレプリケーションの設定方法について説明します。



(注)

オンデマンド コンテンツ レプリケーションと、コンテンツのステージングおよびレプリケーションの機能には、CSS 拡張機能セットのライセンスが必要です。

この章の内容は、特に指定のない限り、すべての CSS モデルに共通です。

この章の主な内容は次のとおりです。

- [「オンデマンド コンテンツ レプリケーションの設定」](#)
- [「コンテンツのステージングとレプリケーションの設定」](#)

## オンデマンド コンテンツ レプリケーションの設定

Web サイトにとって最も難しい問題の 1 つに、ホット コンテンツが突然発生することによって生ずる予測不可能なトラフィックの混雑をいかに管理するか、という問題があります。オンデマンド コンテンツ レプリケーションを使用すると、コンテンツ要求を追跡し、Web サーバまたはキャッシュをオーバーフローさせるホット コンテンツを動的に特定し複製することが可能になります。



(注) オンデマンド コンテンツ レプリケーション機能を使用するには、CSS 拡張機能セットのライセンスが必要です。

オンデマンド コンテンツ レプリケーションはトラフィックベースで行われます。トラフィック フローが増加すると、コンテンツのレプリケーション サービスが自動的に有効になります。オンデマンド コンテンツ レプリケーションを設定すると、CSS によって次の処理が自動的に行われます。

1. URL のヒット数がホット リストしきい値の設定を超えたときに、ホット リストを使用してホット コンテンツを検出します。
2. コンテンツ ルールを動的に変更して、ホット コンテンツに対するサービスが可能なサービスを追加します。

次の項で、サービス レプリケーションの設定方法について説明します。

- [「オンデマンド コンテンツ レプリケーションの設定のクイック スタート」](#)
- [「ホット リストの設定」](#)
- [「レプリケーションのサービス タイプの指定」](#)
- [「最大経過時間の設定」](#)
- [「最大コンテンツ数の設定」](#)
- [「最大ディスク使用量の設定」](#)
- [「コンテンツ レプリケーションの FTP アクセスの設定」](#)
- [「FTP レコードの作成」](#)

## オンデマンド コンテンツ レプリケーションの設定のクイック スタート

表 14-1 に、オンデマンド コンテンツ レプリケーションの設定に必要な手順の概要を示します。それぞれの手順には、作業に必要な CLI コマンドも示しています。これらの CLI コマンドの機能と、関連する各オプションの詳細については、表 14-1 以降の項を参照してください。

表 14-1 オンデマンド コンテンツ レプリケーションの設定のクイックスタート

---

### 作業とコマンドの例

---

1. グローバル設定モードに移行します。

```
# config  
(config)#
```

2. 必要に応じて FTP レコードを作成します。「[FTP レコードの作成](#)」を参照してください。

```
(config)# ftp-record myftprecord 172.16.6.58 bobo "secret" /
```

3. 所有者モードに入ります。

```
(config)# owner arrowpoint  
(config-owner[arrowpoint])#
```

4. 所有者コンテンツ設定モードに移行します。

```
(config-owner[arrowpoint])# content rule1  
(config-owner-content[arrowpoint-rule1])#
```

5. **hotlist** コマンドを使用して、要求回数が最も多いコンテンツ (ホットコンテンツ) の名前を取り込むリストを設定します。詳細については、「[ホットリストの設定](#)」を参照してください。**hotlist** コマンドには、次の 4 つのオプションを指定できます。

- interval
- size
- threshold
- type

```
(config-owner-content[arrowpoint-rule1])# hotlist interval 10
```

---

表 14-1 オンデマンド コンテンツ レプリケーションの設定のクイックスタート (続き)

## 作業とコマンドの例

6. ローカル タイプ (デフォルト) とレプリケーション タイプのサービスを、それぞれ 1 つ以上設定します。サービス設定の詳細については、第 3 章「サービスの設定」を参照してください。

```
(config)# service local_serv
(config-service[local_serv])# exit
(config)# service rep_serv
(config-service[rep_serv])# type rep-cache-redir
```

7. FTP へのアクセス メカニズムを、オンデマンド レプリケーション用サービスと関連付けます。このコマンドは、パブリッシング サービスを提供するサービスごとに使用する必要があります。「コンテンツ レプリケーションの FTP アクセスの設定」を参照してください。

```
(config-service[rep_serv])# access ftp myftprecord
```

8. (オプション) **hotlist** コマンドを明示的に実行する代わりに、コマンドスケジューラを使用して有効化します。コマンドスケジューラを使用すれば、**hotlist** コマンドを特定の時点で挿入し、それ以降の時点で無効化する (**no hotlist**) ことができます。**hotlist** コマンドを有効にすると、指定されたホットリストに含まれるコンテンツは、ヒット回数が **hotlist threshold** の値を超えるたびにレプリケーションが可能になります。コマンドスケジューラの使用方法の詳細については、『Cisco Content Services Switch Administration Guide』を参照してください。

```
# (config)# cmd-sched record content_replication 30 21 3 6 1
```

表 14-1 に示した各コマンドを実行すると、次のような実行設定が得られます。

```
!***** SERVICE *****
service local_serv

service rep_serv
  type rep-cache-redir
  access ftp myftprecord

!***** OWNER *****
owner arrowpoint
  address "200 Beaver Brook Road, Boxborough, MA 01719"

content rule1
  vip address 192.1.1.100
  protocol tcp
  port 80
  hotlist interval 10
```

## ホット リストの設定

コンテンツ ルールにホット リスト アトリビュートを定義すると、最も頻繁にアクセスされるコンテンツがどれかを判断できます。この情報に基づいて、複製する必要のあるコンテンツを正確に判断できます。CSS では、コンテンツ ルールにホット リスト アトリビュートを設定することができます。**hotlist** コマンドを使用して、要求回数が最も多いコンテンツ (ホット コンテンツ) の名前を取り込むリストを設定します。



(注) replication-store タイプや replication-cache タイプのサービスを機能させるには、ホット リストを設定して有効化する必要があります。

設定所有者コンテンツ モードで、特定のコンテンツのホット リストについて、次のアトリビュートを設定することができます。

- **hotlist** : ホット リストを有効にする。特定のコンテンツ ルールについてホット リストを有効にするには、対応する所有者コンテンツ モードで **hotlist** コマンドを入力します。たとえば、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# hotlist
```

## ■ オンデマンド コンテンツ レプリケーションの設定

ホットリストを無効にするには、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# no hotlist
```

- **hotlist interval** : ホット リストの更新間隔を設定する。間隔には 1 ~ 60 (分) を入力します。デフォルトは 1 です。たとえば、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# hotlist interval 10
```

ホット リストの更新間隔をデフォルトの 1 に戻すには、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# no hotlist interval
```

- **hotlist size** : ホット リストのサイズを設定する。このルールについて保持するエントリの合計数を 1 ~ 100 の値で入力します。デフォルトは 10 です。たとえば、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# hotlist size 20
```

ホット リストのサイズをデフォルトの 10 に戻すには、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# no hotlist size
```

- **hotlist threshold** : ホット リストのしきい値を設定する。0 ~ 65535 の整数を入力してしきい値を指定します。このしきい値を超えるコンテンツがホットコンテンツとみなされます。デフォルトは 0 です。たとえば、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# hotlist threshold 9
```

ホット リストのしきい値をデフォルトの 0 に戻すには、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# no hotlist threshold
```

- **hotlist type hitCount** : ホット リストのタイプをヒット カウントに設定する。ヒット カウントは、コンテンツへのアクセス回数です。たとえば、次のように設定します。

```
(config-owner-content [arrowpoint-rule1])# hotlist type hitCount
```

ホット リストのタイプをデフォルト設定 (**hitCount**) に戻すには、次のように入力します。

```
(config-owner-content [arrowpoint-rule1])# no hotlist type
```

## レプリケーションのサービス タイプの指定

レプリケーション設定では、少なくとも 2 つのサーバを設定する必要があります。1 つはローカル タイプ、もう 1 つはレプリケーション タイプです。CSS では、次のようなレプリケーション専用のサービス タイプを使用できます。

- **type rep-cache-redir** : サービスを、リダイレクトを伴うレプリケーション キャッシュに指定する。
- **type rep-store** : サービスをレプリケーション ストア（コンテンツ要求の負荷分散に使用されるローカルのオーバーフロー サービス）に指定する。
- **type rep-cache-redir** : サービスを、リダイレクトを伴うレプリケーション ストアに指定する。

**type rep-cache-redir** に指定したサービスは、CSS がキャッシュ サーバとして使用します（このサービスにホット コンテンツがキャッシングされ、要求が送信されます）。レプリケーション サーバにコンテンツがキャッシングされると、CSS はホット コンテンツおよびダイナミック サービスのためのダイナミック コンテンツ ルールを作成します。

最大経過時間が過ぎると、CSS はホット コンテンツを削除します。この章で後述する「[最大経過時間の設定](#)」を参照してください。

たとえば、次のように設定します。

```
(config)# service serv1
(config-service[serv1])# type rep-cache-redir
```

サービスを **type rep-store** として指定すると、CSS はそのサービスでホット コンテンツを複製します。レプリケーション サーバにコンテンツが複製されると、そのホット コンテンツについてダイナミック コンテンツ ルールが自動的に作成されます。ダイナミック コンテンツ ルールは、既存ルールのすべてのアトリビュートを、次の点だけ変更して継承します。

- コンテンツ ルール コマンド **uri** でホット コンテンツを一意に識別する。
- サーバ タイプを **replication-store** から **local** に変更する。

## ■ オンデマンド コンテンツ レプリケーションの設定

ダイナミック コンテンツ ルールは、最大経過時間が過ぎると削除されます。詳細については、後述する「[最大経過時間の設定](#)」を参照してください。ダイナミック コンテンツ ルールのリストは、**show rule** 画面に表示されます。ダイナミック コンテンツ ルールは、**running-config** (実行設定) または **startup-config** (起動設定) ファイルには表示されません。



(注) レプリケーション タイプのサービスは、コンテンツがそのサービスで複製されるまでロード バランシング アルゴリズムには含まれません。

次に設定例を示します。

```
(config)# service serv1
(config-service[serv1])# type rep-store
```

## 最大経過時間の設定

**max age** コマンドを使用して、**rep-cache-redis**、**rep-store**、または **rep-store-redis** として定義されたサービスで複製されるオブジェクトの最大経過時間を定義します。最大経過時間 (分) を 1 ~ 1440 の値で入力します。デフォルトは 120 です。

たとえば、次のように入力します。

```
(config-service[serv1])# max age 10
```

複製されたオブジェクトの最大経過時間をデフォルトの 120 分に戻すには、次のコマンドを入力します。

```
(config-service[serv1])# no max age
```



## 最大コンテンツ数の設定

**max content** コマンドを使用して、**rep-cache-redis**、**rep-store**、または **rep-store-redis** として定義されたサービスで複製するコンテンツの最大数を定義します。コンテンツの最大数を 1 ~ 65535 の値で入力します。デフォルトは 100 です。

たとえば、次のように入力します。

```
(config-service[serve1])# max content 50
```

最大コンテンツ数をデフォルト値の 100 に戻すには、次のコマンドを入力します。

```
(config-service[serve1])# no max content
```

## 最大ディスク使用量の設定

**max usage** コマンドを使用して、**rep-cache-redis**、**rep-store**、または **rep-store-redis** として定義されたサービスで複製に使用できる最大ディスク スペースを定義します。サービスのディスク スペースを 1 ~ 1000MB の範囲内で入力します。デフォルトは 1 です。

次に設定例を示します。

```
(config-service[serve1])# max usage 100
```

最大ディスク スペースをデフォルト値の 1MB に戻すには、次のコマンドを入力します。

```
(config-service[serve1])# no max usage
```

## コンテンツ レプリケーションの FTP アクセスの設定

パブリッシング サービスを提供する各サービスには、個別に FTP アクセス メカニズムを関連付ける必要があります。 **access ftp** コマンドを使用して FTP アクセス メカニズムとオンデマンド レプリケーション アクティビティのサービスを関連付けます。

このコマンドを使用して、FTP アクセス メカニズムをサービスに関連付ける場合は、既存の FTP レコードのベース ディレクトリがツリーのルートになります。WWW デーモンと FTP デーモン間のマッピングの一貫性を保つために、FTP アクセスのベース ディレクトリを、クライアントから見えるとおり、WWW デーモンのルート ディレクトリと同等にします。

**access ftp\_record** に既存の FTP レコードの名前を入力します。FTP レコードの名前は、スペースを含まないテキスト文字列を引用符で囲まずに入力します。



(注) FTP レコードを作成するには、**(config) ftp-record** コマンドを使用します。FTP レコードの作成については、この章の「[FTP レコードの作成](#)」を参照してください。

たとえば、次のように設定します。

```
(config-service[serv1])# access ftp myftprecord
```

サービスのアクセス メカニズムを削除するには、次のように入力します。

```
(config-service[serv1])# no access myftprecord
```



(注) コンテンツ レプリケーションは、WSFTP FTP アプリケーションをサポートしません。

## FTP レコードの作成

CSS から FTP サーバにアクセスする際に使用する File Transfer Protocol (FTP; ファイル転送プロトコル) レコード ファイルを作成するには、**ftp-record** コマンドを使用します。このグローバル設定モードのコマンドのシンタックスは次のとおりです。

```
ftp-record ftp_record ip_address (または hostname username)  
["password"][des-password des_password] base_directory
```

変数の内容は次のとおりです。

- *ftp\_record* : FTP レコード ファイルの名前。スペースを含まない 16 文字以内のテキスト文字列を引用符で囲まずに入力します。
- *ip\_address* または *hostname* : アクセスする FTP サーバの IP アドレスまたはホスト名。
- *username* : FTP サーバの有効なログイン ユーザ名。大文字小文字を区別して、スペースを含まない 32 文字以内のテキスト文字列を引用符で囲まずに入力します。
- *"password"* : FTP サーバのログイン ユーザ名用のパスワード。スペースを含まない文字列を、大文字小文字を区別して引用符で囲んで入力します。長さは 16 文字までです。
- *des\_password* : FTP サーバの有効なログイン ユーザ名の、DES 暗号化済みパスワード。大文字小文字を区別して、スペースを含まない 64 文字以内のテキスト文字列を引用符で囲まずに入力します。
- *base\_directory* : このレコードを使用する際のオプションのベース ディレクトリ。

たとえば、次のように設定します。

```
(config)# ftp-record myftprecord 172.16.6.58 bobo "secret" /
```

CSS から FTP レコード ファイルを削除するには、**no ftp-record** コマンドを使用して FTP レコード名を指定します。たとえば、次のように設定します。

```
(config)# no ftp-record myftprecord
```

## コンテンツのステージングとレプリケーションの設定

CSS は、パブリッシャ サービスとサブスクリバ サービスを使用して、コンテンツのステージングおよびレプリケーションをサポートします。これらの機能では、ステージング パブリッシャ サーバに投稿したコンテンツ（1 つのファイル、複数のファイル、ディレクトリ全体など）が、次のいずれかの手段によってトリガーされて CSS によって取り込まれ、複数のサブスクリバ サーバに動的に複製されます。

- CLI コマンド
- ステージング サーバ上の特定コンテンツへの変更。変更が検出されると、CSS はコンテンツをサブスクリバ サーバまたはキャッシュに動的に複製します。

CSS は、ファイル名、サイズ、およびファイルの日付を FTP ベースで検査し、特定のコンテンツへの変更を検出します。この検査は、`publisher interval` で設定された間隔または `replicate` コマンドによって行われます。サブスクリバは、パブリッシャとして指定されたサービスに関連する `access ftp` によって、パブリッシャとのインターフェイス方法を識別します。

- (オプション) コマンドスケジューラ機能を使用した、月、日、時間の指定。



(注)

---

コンテンツのステージングおよびレプリケーション機能を使用するには、CSS 拡張機能セットのライセンスが必要です。

---

ソフトウェアのアップグレード後や、新しいサブスクリバを追加した後は、パブリッシャとサブスクリバの情報を確実に同期させるためにも、`replicate` コマンドを `force` オプション付きで実行することが推奨されます。このコマンドを実行すると、CSS は `publisher interval` で設定された時間間隔に従って、パブリッシャに生じたファイルやディレクトリのあらゆる変更に関連しているサブスクリバを自動的に更新します。

設定済みのコンテンツ レプリケーションと同じソフトウェアイメージで CSS をリブートしたときに、その CSS がパブリッシャに変更があることを検出すると、パブリッシャから関連するサブスクリバに変更が適用され、両者の同期が維持されます。この動作は、これらのパブリッシャとサブスクリバが、以前に同期されていることが前提になります。



(注) サブスクライバのファイルやディレクトリを手動で操作すると、そのサブスクライバとパブリッシャ間の同期が失われます。サブスクライバとパブリッシャのコンテンツを再同期するには、**replicate** コマンドを **force** オプション付きで実行する必要があります。

コンテンツのステージングおよびレプリケーションの機能で複製されるファイルのサイズには、特に制限はありません。ただし、ファイルのサイズが大きくなれば、複製に必要な時間も長くなります。この機能で制限されるのは、複製ファイルの数（最大 100 万）だけです。CSS は、複製されたファイルを自身のディスクには格納しないことに注意してください。そのかわり CSS は、パブリッシャサービスとサブスクライバサービス間でデータを転送するための仮想パスを作成します。また、使用されるバッファリングは、TCP 関連のデフォルトのバッファリングだけです。

CSS は、複製されたコンテンツを継続的にアップデートするように設定できます。たとえば、CSS を使用すると、ニュース速報に関連するコンテンツを複製できます。アップデートしたコンテンツをステージング サーバへ投稿すると、そのアップデートはすべての分散ロケーションへ自動的に複製されます。

パブリッシャおよびサブスクライバのサービス タイプは、通常 **local** に定義します。サービス タイプを変更する必要はありません。

次の項では、パブリッシャ サービスおよびサブスクライバ サービスの設定方法について説明します。

- 「コンテンツのステージングおよびレプリケーションのクイック スタート」
- 「パブリッシングとサブスクライビング用の FTP アクセスの設定」
- 「パブリッシング サービスの設定」
- 「サブスクライバ サービスの設定」
- 「コンテンツ レプリケーションとステージングのためのコンテンツ ルールの設定」
- 「パブリッシャ コンテンツ レプリケーションの設定」
- 「コンテンツの表示」

## コンテンツのステージングおよびレプリケーションのクイック スタート

表 14-2 に、オンデマンド コンテンツ レプリケーションの設定に必要な手順の概要を示します。それぞれの手順には、作業に必要な CLI コマンドも示しています。これらの CLI コマンドの機能と、関連する各オプションの詳細については、表 14-2 以降の項を参照してください。

**表 14-2** コンテンツのステージングおよびレプリケーションのクイック スタート

---

### 作業とコマンドの例

---

1. 設定モードに入ります。

```
# config
(config)#
```

2. 必要に応じて FTP レコードを作成します。「[FTP レコードの作成](#)」を参照してください。

```
(config)# ftp-record myftprecord 172.16.6.58 bobo "secret" /
```

3. パブリッシング サービスとして使用されるサービスを作成します。

```
(config)# service pubserver
(config-service[pubserver])#
```

4. サービスをパブリッシング サービスとして設定します。「[パブリッシング サービスの設定](#)」を参照してください。

```
(config-service[pubserver])# publisher
```

5. 各サブスクリバとの間でコンテンツを同期する時間間隔（分単位）を設定します。「[パブリッシング サービスの設定](#)」を参照してください。

```
(config-service[pubserver])# publisher interval 120
```

6. FTP へのアクセス メカニズムを、オンデマンド レプリケーション用サービスと関連付けます。このコマンドは、パブリッシング サービスを提供するサービスごとに使用する必要があります。「[コンテンツ レプリケーションの FTP アクセスの設定](#)」を参照してください。

```
(config-service[pubserver])# access ftp myftprecord
```

7. サブスクリバ サービスとして使用されるサービスを作成します。

```
(config)# service subserver
(config-service[subserver])#
```

---

表 14-2 コンテンツのステージングおよびレプリケーションのクイック スタート (続き)

## 作業とコマンドの例

8. **subscriber** コマンドを使用して、パブリッシング サービスに対するサブスクリバとしてサービスを設定します。「[サブスクリバ サービスの設定](#)」を参照してください。

```
(config-service[subserver])# subscriber pubserver
```

9. 複製するファイルを指定するためにコンテンツ ルール内に URL を設定します。

```
(config-owner-content[arrowpoint-products.html])# url  
"/announcements/*.html"
```

URL の設定方法の詳細は、[第 10 章「コンテンツ ルールの設定」](#)を参照してください。

10. **add service** コマンドを使用して、コンテンツ ルールにサブスクリバ サービスを追加します。

```
(config-owner-content[arrowpoint-products.html])# add service  
subserver
```

11. パブリッシャとすべての関連サブスクリバ間のレプリケーションをただちに開始するには、**replicate** コマンドを使用します。

```
# replicate pubserver
```

12. (オプション) コンテンツ レプリケーションをただちに開始する代わりに、コマンド スケジューラを使用して、コンテンツ レプリケーションを開始する月、日、時刻を指定します。コマンド スケジューラの使用方法の詳細については、『*Cisco Content Services Switch Administration Guide*』を参照してください。

```
# (config)# cmd-sched record content_replication 30 21 3 6 1
```

13. (推奨) パブリッシング サービスの運用状況とコンテンツ情報を表示するには、**show publisher** コマンドを使用します。

```
# show publisher
```

14. (推奨) **show content** コマンドを使用して、CSS の Content Service Database (CSD; コンテンツ サービス データベース) のコンテンツ エントリを表示します。

表 14-2 に示した各コマンドを実行すると、次のような実行設定が得られます。

```
!***** SERVICE *****
service pubserver
  publisher interval 120
  access ftp myftprecord

service subserver
  subscriber pubserver

!***** OWNER *****
owner arrowpoint
  address "200 Beaver Brook Road, Boxborough, MA 01719"

content products.html
  protocol tcp
  port 80
  url "/announcements.html"
  add service subserver
```

## パブリッシングとサブスクライビング用の FTP アクセスの設定

パブリッシング サービスを提供する各サービス、およびサブスクライバとして設定した各サービスには、アクセス メカニズムを個別に関連付ける必要があります。**access ftp** コマンドを使用して、FTP アクセス メカニズムをサービスに関連付けて、パブリッシングおよびサブスクライビングのアクティビティ中に使用できるようにします。

FTP レコードには、既存の FTP レコードの名前を入力します。FTP レコードの名前は、スペースを含まないテキスト文字列を引用符で囲まずに入力します。



(注)

コンテンツのステージングおよびレプリケーションを設定するには、コンテンツのステージングおよびレプリケーションのコマンドを設定する前に、FTP レコードを作成する必要があります。FTP レコードを作成しないと、機能は適切に動作しません。FTP レコードを作成するには、**(config) ftp-record** コマンドを使用します。詳細については、この章で前述した「[FTP レコードの作成](#)」を参照してください。



このサービス モードのコマンドのシンタックスは次のとおりです。

```
(config-service [pubserver])# access ftp myftprecord
```

サービスのアクセス メカニズムを削除するには、次のように入力します。

```
(config-service [pubserver])# no access ftp
```

## パブリッシング サービスの設定

パブリッシング サービスは、相互に関連するサブスライバ サービス間でコンテンツを同期させるサービスです。パブリッシング動作中にコンテンツを移動するには、パブリッシャ サービスのアクセス メカニズムを設定する必要があります。この章で前述した **(config-service) access ftp** コマンドを使用して、パブリッシャ サービスのメカニズムを設定します。

サブスライバを同期化する間隔を定義すると、間隔はコマンド実行時からカウントされます。同期化に使用できないサブスライバはオフラインの状態に置かれ、処理が完了するまで繰り返し実行されます。

パブリッシャとサブスライバ間で CSS が複製できるファイルのサイズには制限がありません。パブリッシャとサブスライバ間でデータを転送するときに、CSS は仮想パイプを作成するので、複製されたファイルは CSS のディスクを使用しません。CSS は、TCP 通信スタックに関連するデフォルトのバッファリングを使用します。



(注)

パブリッシャ サービスは、少なくとも 1 つのサブスライバが設定されるまでアクティブになりません。サブスライバを設定する前にパブリッシャを設定しておく必要はありませんが、サブスライバがコンテンツ同期化のアップデートを受信するには、パブリッシャを設定しておく必要があります。

**publisher** コマンドを使用して、サービスをパブリッシング サービスとして設定します。パブリッシング サービスには、アプリケーションに適合するタイプのサービスであれば、どのサービス（たとえば、**local** または **proxy-cache**）でも使用できます。サービス タイプの詳細については、第 3 章「サービスの設定」を参照してください。

このサービス モードのコマンドのシンタックスおよびオプションは、次のとおりです。

- **publisher** : サービスをパブリッシング サービスとして設定する。
- **publisher interval minutes** : サブスクリバ間のコンテンツを同期する間隔（分）を定義する。このサービスをパブリッシング サービスとして設定していない場合、このコマンドは実行できません。同期化の間隔を分数で入力します。0 ~ 3600 の数値を入力します。デフォルトは 0（間隔は無効）です。
- **publisher interval minutes trigger\_filename** : 指定されたトリガー ファイルが変更された場合にだけ、サブスクリバ間のコンテンツを同期する間隔（分）を定義する。**trigger\_filename** は、1 ~ 64 文字で指定します。このサービスをパブリッシング サービスとして設定していない場合、このコマンドは実行できません。

サービスにパブリッシングを設定するには、次のコマンドを入力します。

```
(config-service [pubserver])# publisher
```

サービスのパブリッシングを削除するには、次のコマンドを入力します。

```
(config-service [pubserver])# no publisher
```

パブリッシャの同期化間隔を設定するには、次のコマンドを入力します。

```
(config-service [pubserver])# publisher interval 120
```

パブリッシャの同期化間隔をデフォルトの 0 に設定して無効にするには、次のコマンドを入力します。

```
(config-service [pubserver])# no publisher interval
```

## パブリッシャ設定の表示

パブリッシング サービスの運用状況とコンテンツ情報を表示するには、**show publisher** コマンドを使用します。オプションとシンタックスは次のとおりです。

- **show publisher** : 設定されたすべてのパブリッシング サービスの情報を表示する。
- **show publisher publisher\_name** : 指定したパブリッシング サービスの情報を表示する。
- **show publisher publisher\_name content {verbose}** : 指定したパブリッシング サービスのコンテンツ情報を表示する。verbose オプションを含めると、さらに詳しいコンテンツ情報を表示できます。

パブリッシング サービスについての情報を表示するには、次のコマンドを入力します。

```
(config-service)# show publisher
```

表 14-3 に、**show publisher** コマンドで表示されるフィールドについて説明します。

**表 14-3 show publisher コマンドのフィールド**

フィールド	説明
State	パブリッシャ サービスの状態
Access Type	パブリッシング アクティビティの際に使用する、サービスに関連付けられたアクセス メカニズム。現在は、FTP レコードが唯一のメカニズムです。
Access IP	FTP レコードの IP アドレス
Access Port	アクセス メカニズムに関連付けられている FTP レコードのポート番号
Access Username	FTP レコードで定義した FTP サーバのユーザ名
Access Base Dir	FTP レコードで定義したベース ディレクトリ
Published Files	パブリッシャからそのサブスクリバにパブリッシュしたファイル数
Published Bytes	パブリッシャからそのサブスクリバにパブリッシュしたバイト数

表 14-3 show publisher コマンドのフィールド (続き)

フィールド	説明
Subscribers	そのパブリッシャを使用するように設定されたサブスクライバの数
Trigger File	パブリッシャとサブスクライバ間の同期化が行われる原因となった変更が行われたファイル
Publish Interval	パブリッシャが、サブスクライバとの同期をチェックする間隔 (秒数)
Next Interval	パブリッシャが次に同期チェックを行う時間
Managed Files	パブリッシャが複製するファイルの数
Subscribers Synced	同期化されたサブスクライバの数
Managed Dirs	パブリッシャが複製するファイルの数
Managed Bytes	パブリッシャがトラッキングしているバイト数
Last Method	<p>前回パブリッシャがサブスクライバに対して試みた同期方式。同期方式には次のものがあります。</p> <ul style="list-style-type: none"> <li>• <b>cli</b> : ユーザによって開始される同期化</li> <li>• <b>interval</b> : 設定した間隔による同期化</li> <li>• <b>signal</b> : トリガー ファイルによる同期化</li> <li>• <b>retry</b> : パブリッシャが前回の同期化に失敗した場合の同期化の再試行</li> <li>• <b>reboot</b> : CSS のリブート</li> </ul>
Last Time	パブリッシャがサブスクライバとの同期を最後に試みた時間

## サブスクリバ サービスの設定

パブリッシング サービスに対するサブスクリバとしてのサービスを設定するには、**subscriber** コマンドを使用します。サブスクリバは、パブリッシャごとに最大 31 まで定義できます。

それぞれのサブスクリバでアクセス メカニズムを設定する必要があります。それぞれのサブスクリバのアクセス メカニズムを設定するには、この章で前述した (**config-service**) **access ftp** コマンドを使用します。

サービスをパブリッシャへのサブスクリバとして設定するには、次のコマンドを入力します。

```
(config-service[subserver])# subscriber pubserver
```

パブリッシング サービスからのサービスの受信を取り消すには、次のコマンドを入力します。

```
(config-service[subserver])# no subscriber
```



(注)

サブスクリバは、パブリッシャの状態がレディになってから、レディ状態またはアクセスできる状態になります。

## サブスクリバ設定の表示

サブスクリバ サービスの運用状態を表示するには、**show subscriber** コマンドを使用します。このコマンドのシンタックスは次のとおりです。

- **show subscriber** : 設定されたすべてのサブスクリバ サービスの情報を表示する。
- **show subscriber publisher\_name** : 指定したパブリッシング サービスのすべてのサブスクリバ サービスの情報を表示する。
- **show publisher publisher\_name subscriber\_name** : 指定したパブリッシング サービスの、指定したサブスクリバ サービスの情報を表示する。

サブスクリバ サービスの情報を表示するには、次のコマンドを入力します。

```
(config)# show subscriber
```

表 14-4 に、**show subscriber** コマンドで表示されるフィールドについて説明します。

表 14-4 **show subscriber** コマンドのフィールド

フィールド	説明
State	サブスクリバサービスの状態
Access Type	サブスクリビング アクティビティの際に使用する、サービスに関連付けられている FTP アクセス メカニズム
Access IP	アクセス メカニズムに関連付けられている FTP レコードの IP アドレス
Access Port	アクセス メカニズムに関連付けられている FTP レコードのポート番号
Access Username	アクセス メカニズムに関連付けられている FTP レコードのユーザ名
Access Base Dir	アクセス メカニズムに関連付けられている FTP レコードのベース ディレクトリ
Subscribed Files	サブスクリバで複製されたファイル数
Subscribed Bytes	サブスクリバで複製されたバイト数
Last Method	<p>前回パブリッシャがサブスクリバに対して試みた同期方式。同期方式には次のものがあります。</p> <ul style="list-style-type: none"> <li>• <b>cli</b> : ユーザによって開始される同期化</li> <li>• <b>interval</b> : 設定した間隔による同期化</li> <li>• <b>signal</b> : トリガー ファイルによる同期化</li> <li>• <b>retry</b> : パブリッシャが前回の同期化に失敗した場合の同期化の再試行</li> <li>• <b>reboot</b> : CSS のリブート</li> </ul>
Last Time	パブリッシャがサブスクリバとの同期を最後に試みた時間
Synchronized	サブスクリバがパブリッシャと現在同期しているかどうかを示す。

## コンテンツ レプリケーションとステージングのためのコンテンツ ルールの設定

コンテンツのステージングおよびレプリケーションを設定する際には、コンテンツ ルールに URL を設定して、複製するファイルを定義する必要があります。次に、サブスクリバ サービスをコンテンツ ルールに追加します。



(注) すべてのディレクトリにあるすべてのファイルを複製する場合は、コンテンツ ルールを作成する必要はありません。複製したいファイルだけを指定するコンテンツ ルールを作成すれば十分です。



(注) コンテンツ ルールには、サブスクリバの URQL は設定できません。

たとえば、ディレクトリ *announcements* に存在する、拡張子が *.html* のコンテンツに対する要求すべてに一致する URL を指定するには、次のように入力します。

```
(config-owner-content [arrowpoint-products.html])# url  
"/announcements/*.html"
```

URL の設定方法の詳細については、[第 10 章「コンテンツ ルールの設定」](#)を参照してください。

サブスクリバ サービスをコンテンツ ルールに追加するには、**add service** コマンドを使用します。たとえば、次のように入力します。

```
(config-owner-content [arrowpoint-products.html])# add service subserver
```

## パブリッシャ コンテンツ レプリケーションの設定

パブリッシャとすべての関連サブスクリバ間のレプリケーションをただちに開始するには、**replicate** コマンドを使用します。このコマンドは、パブリッシャのコンテンツの変更をサブスクリバに複製したり、すべてのコンテンツを新規のサブスクリバに強制的に再同期する手段として使用できます。

*publisher\_name* には、既存のパブリッシャの名前を入力します。*subscriber\_name* には、パブリッシャ サービスに関連付けられたサブスクリバの名前を入力します。

シンタックスおよびオプションは、次のとおりです。

- **replicate publisher\_name** : 指定したパブリッシャとそのサブスクリバ サービス間でコンテンツの変更を再同期化する。コンテンツに変更がない場合、再同期化は行われません。
- **replicate publisher\_name subscriber\_name** : 指定したパブリッシャと、指定したサブスクリバ サービス間でコンテンツの変更を再同期化する。コンテンツに変更がない場合、再同期化は行われません。
- **replicate publisher\_name subscriber\_name force** : 指定したパブリッシャと、指定したサブスクリバ サービス間で、変更の有無に関わらずコンテンツをすべて再同期化する。このオプションは次の場合にだけ使用します。
  - 新しい設定
  - ソフトウェアのアップグレード
  - 新規のサブスクリバ
  - サーバにディスク障害がある

ルーチン コンテンツの複製には **force** オプションは使用しないでください。**force** オプションでは、パブリッシャ ディレクトリ構造をスキャンしません。このため、前回のスキャン以降にパブリッシャ ディレクトリ構造に追加された、または削除されたファイルは示されません。上記で説明した以外で **force** オプションを使用する場合は、コンテンツを手動で複製することで CSS がパブリッシャ ツリーをスキャンできるようにしてから、**force** オプションを適用します。

次に使用例を示します。

```
# replicate pubserver
```



## コンテンツ レプリケーションのファイル エラー処理の設定

ごくまれな状況で、コンテンツ レプリケーションの処理中にファイル エラーが発生する場合があります。ファイル エラーは、アプリケーションまたはユーザが複製処理中にパブリッシュ ツリーからファイルを削除する場合に発生します。このようなファイル エラーが発生した場合は、削除されたファイルをスキャンで検出できません。複製中、CSS は別のスキャンを開始するか、削除されたファイルが利用可能になるまでスキャンを再試行します。

コンテンツ リプリケーションの処理中に発生したファイル エラーの処理方法は、**replication file-error** コマンドで CSS に指定します。このグローバル設定モードのコマンドのシンタックスは次のとおりです。

### **replication file-error retry|skip**

このコマンドには、次の各オプションがあります。

- **retry** : (デフォルト) 見つからないファイルの複製を CSS が試みている間は複製が一時的に中止される。
- **skip** : 見つからないファイルの処理をスキップして、複製処理を続行する。

次に使用例を示します。

```
(config)# replication file-error skip
```

## コンテンツの表示

**show content** コマンドを使用すれば、CSS のコンテンツ サービス データベース (CSD) のコンテンツ エントリを表示することができます。このコマンドは、すべてのモードで実行できます。

シンタックスを次に示します。

### **show content slot slot\_number {start-index index\_number}**

変数とオプションは次のとおりです。

- **slot slot\_number** : CSS シャーシの特定のスロットにあるモジュールからのコンテンツを表示する。CSS 11503 の場合、1 ~ 3 ままでが選択できます。CSS 11506 の場合、1 ~ 6 ままでが選択できます。スロット番号を指定しなかった場合は、CSS のスロット 1 にある SCM のコンテンツ エントリが表示されません。

## ■ コンテンツのステー징とレプリケーションの設定

- **start-index** *index\_number* : *index\_number* パラメータで指定した番号以降のコンテンツ エントリが表示される。この変数は、ブラウザを開始する CSS のコンテンツの場所を定義します。指定したインデックス番号から 64 KB までの情報を受信できます。さらに情報を表示するには、最後に表示されたインデックスを指定して **show content** コマンドを再び実行します。インデックス番号を指定するには、0 ~ 4095 の数値を入力します。開始するインデックスを指定しないと、0 のコンテンツ エントリから表示されます。

たとえば、シャーシのスロット 2 のモジュールのコンテンツをインデックス 150 から表示するには、次のように入力します。

```
(config)# show content slot 2 start-index 150
```

表 14-5 に、**show content** コマンドで表示されるフィールドについて説明します。

**表 14-5 show content コマンドのフィールド**

フィールド	説明
Pieces of Content for Slot	モジュールがあるシャーシのスロット番号
Subslot	セッションプロセッサがあるモジュールのスロット番号
Total Content	コンテンツ エントリの総数
Index	CSD 内で認識されているコンテンツを表す一意のインデックス
<address>	このコンテンツの IP アドレス
Protocol	このコンテンツの IP プロトコル
Port	このコンテンツのプロトコル ポート
Best Effort	このコンテンツの QoS クラス。このフィールドは、CSS では現在使用していません。
Streamed	コンテンツがストリーミング メディア (ビデオやオーディオ) かどうかを指定します。このフィールドは、CSS では現在使用していません。
URL	コンテンツの URL
Domain	このコンテンツのドメイン