



## Cisco Finesse

- 新機能, on page 1
- 更新済み機能, on page 9
- 特記事項, on page 11
- 廃止された機能, on page 12
- 削除され、サポートされない機能, on page 13
- サードパーティ ソフトウェアへの影響, on page 13

## 新機能

### Finesse フェールオーバーの改善

このリリースでは、Cisco Finesse CTI フェールオーバーおよびデスクトップ フェールオーバーのパフォーマンスが最適化されています。

- CTI フェールオーバー: エージェント PG 12.5(1) を使用して展開した場合、CTI サーバ/エージェント PG のフェールオーバーの時間が、約 35 秒 ~ 75 秒の間で変化します。エージェント PG 12.0 (1) または 11.6 (1) を使用して展開した場合、CTI サーバ/エージェント PG のフェールオーバーの時間が、約 75 秒 ~ 120 秒の間で変化します。
- デスクトップ フェールオーバー: エージェント PG 12.5(1) を使用して展開した場合、デフォルトのデスクトップ レイアウトでデスクトップがフェールオーバーする時間は、およそ 50 秒 ~ 110 秒の間で変化します。Agent PG 12.0 (1) または 11.6 (1) を使用して展開した場合、デスクトップ フェールオーバーの時間は、エージェント PG (1) の場合と比較して約 40 秒長くなります。

Cisco Finesse に対して 8 個の vCPU を使用した OVA が設定されている場合、CTI サーバ/エージェント PG のフェールオーバーとデスクトップ フェールオーバーにかかる時間が 20% 向上します。この設定は、24000 エージェント展開タイプを含むすべての展開タイプでサポートされています。8 個の vCPU を使用した OVA の詳細については、[https://www.cisco.com/c/dam/en/us/td/docs/voice\\_ip\\_comm/uc\\_system/virtualization/virtualization-cisco-finesse.html](https://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-cisco-finesse.html) の「Cisco Finesse の仮想化」を参照してください。

フェールオーバーの合計時間は、WAN 帯域幅、サインインしているユーザの数、ネットワーク遅延、設定された vCPU の数、エージェント PG のバージョン、および Finesse デスクトップに設定されているガジェットの数によって異なります。デスクトップフェールオーバーのほとんどの改善点は、（CTI フェールオーバーとは異なり）過去のバージョンのエージェント PG でも改善されています。

最適なフェールオーバー パフォーマンスを確保するための展開方法とガイドラインの詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「デスクトップフェールオーバー最適化のガイドライン」および「フェールオーバー計画」の項を参照してください。

カスタムガジェットによるフェールオーバー パフォーマンスの改善方法の詳細については、<https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide>にある『Cisco Finesse Web サービス デベロッパーガイド』の「ガジェット開発のベストプラクティス」の項を参照してください。

### デスクトップパフォーマンスの改善

このリリースでは、次の分野で Finesse デスクトップ パフォーマンスが最適化されています。

- デフォルトで、リソースのロードに HTTP/2 を使用します。これにより、デスクトップを起動したときに、古い HTTP 1.1 と比べて大幅な改善が得られます。
- Finesse デスクトップをキャッシュから（ガジェットを使用せずに）再ロードする場合に、デスクトップの帯域幅の消費が大幅に削減されます。
- リソースのロードに使用する要求が少なくなります。
- クライアント側のリソース キャッシュを使用することで、静的リソースをはるかに高速に提供します。
- 設定データに対する REST 応答をチームレベルでキャッシュし、デスクトップのロードを改善します。
- ガジェット定義をキャッシュすることで、ガジェットのロードを改善します。

### Finesse サーバのパフォーマンスの改善

このリリースでは、次の分野で Finesse サーバのパフォーマンスが最適化されています。

- サーバがロードされている間の CPU の平均消費量を大幅に削減します。
- 動的サーバページを回避し、SSL ターミネーション、より高速な CTI メッセージ解析、キャッシュされた静的リソースを使用することによって、サーバのパフォーマンスを最適化しています。
- Cisco Finesse サーバが 64 ビット Java 8 を使用する際に、より多くのメモリにアクセスし、GC 遅延を減らしています。
- CTI 要求の処理を最適化することで、サーバに要求を送信する際の遅延を減らしています。
- CTI 通信の全体的な遅延が減少しています。

## キーボードのショートカット

このリリースでは、キーボードショートカットを使用し、Finesse デスクトップ機能に簡単にアクセスすることができます。キーボードショートカットにより、Finesse エージェントおよびスーパーバイザのデスクトップで特定のアクションを実行するための別の方法が定義されます。管理者は、`utils finesse set_property desktop enableShortCutKeys` を `true` に設定して、この機能を有効にすることができます。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html> にある『Cisco Finesse エージェントおよびスーパーバイザデスクトップ ユーザガイド』の「キーボードショートカットへのアクセス」の項を参照してください。

## デスクトップチャット検索

このリリースでは、デスクトップチャット検索を絞り込むことにより、Cisco Unified Communications Manager IM and Presence Service で定義された組織単位 (OU) に基づいて、ユーザの詳細を表示することができます。次の CLI を使用して、管理者は OU ベースのユーザ検索を Cisco Finesse で設定できます。

- `utils finesse set_property desktop desktopChatOUSearchFieldKey <value>`
- `utils finesse set_property desktop desktopChatOUSearchFieldValue <value>`

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html> にある『Cisco Finesse アドミニストレーションガイド』の「デスクトップチャットサーバの設定」の項を参照してください。

## コール変数の変更

このリリースでは、エージェントとスーパーバイザは、コール変数の値を Finesse デスクトップから編集できます。管理者は、ECC 変数などの `callVariable` 値を、編集可能であれば何でも設定できます。エージェントとスーパーバイザは、アクティブ コールまたは後処理状態のときに、コール変数の値を編集できます。

**Note**

コール変数を編集する操作を行うと、特定のコール内で変数の値が更新されます。ダイアログ イベントをリッスンしているすべてのエンティティは、Cisco Finesse の通知を介して更新されたコール変数を受信します。同じエージェント PG に接続されている CTI クライアントがある場合は、それらも CTI コールイベントを通じてコールデータの変更通知を受信します。ただし、コール変数の入力に使用されるアプリケーションのスクリプトやデータベースが、この編集によって直接影響を受けることはありません。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html> にある『Cisco Finesse アドミニストレーションガイド』の「コール変数の変更」のセクション、および <https://www.cisco.com/c/en/us/support/>

[customer-collaboration/finesse/products-user-guide-list.html](https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html)にある『Cisco Finesse エージェントおよびスーパーバイザ デスクトップ ユーザガイド』を参照してください。

## ガジェットやコンポーネントのドラッグアンドドロップおよびサイズ変更

このリリースでは、エージェントとスーパーバイザは、ガジェットまたはコンポーネントのドラッグアンドドロップやサイズ変更を、Finesse デスクトップで行うことができます。管理者は、これらの機能のデスクトッププロパティ値を、次のデスクトップレイアウトを使用してカスタマイズできます。

- デフォルト レイアウト ([デスクトップ レイアウト (Desktop Layout)])
- チーム固有のレイアウト ([チームリソースの管理 (Manage Team Resources)] > [デスクトップ レイアウト (Desktop Layout)])

あるいは、管理者は、`utils finesse set_property desktop enableDragDropAndResizeGadget` を `true` に設定して、これらの機能を有効にすることもできます。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「ガジェットやコンポーネントのドラッグアンドドロップおよびサイズ変更」の項を参照してください。

## ガジェットの展開と折りたたみ

このリリースでは、ガジェットの展開や折りたたみを Finesse デスクトップで動的に行うことにより、使用可能な画面スペースを最適化できます。

詳細については、<https://developer.cisco.com/docs/finesse/#!javascript-library>にある JavaScript ライブラリの「コンテナサービス」の項を参照してください。

## デスクトップ レイアウト エディタ

このリリースでは、Cisco Finesse 管理コンソールのデスクトップ レイアウトおよびチームリソースに、2つのタイプのエディタが用意されています。

- **テキスト エディタ**: プレーンテキストエディタ。デフォルトのエディタです。[すべて展開 (Expand All)] オプションを使用すると、すべてのコードの詳細が表示され、[検索 (Search)] ボックスで結果を絞り込むことができます。
- **XML エディタ**: XML エディタです。このエディタで、管理者がコメント (`&lt;!-- -->`) を追加または編集することはできません。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「デフォルトのレイアウト XML」セクションを参照してください。

## デスクトップ プロパティのカスタマイズ

このリリースでは、次のレイアウトにより、管理者がデスクトップレイアウトを使用して、個々のチームのデスクトッププロパティをカスタマイズすることができます。

- デフォルトレイアウト（[デスクトップレイアウト（Desktop Layout）]）
- チーム固有のレイアウト（[チームリソースの管理（Manage Team Resources）]>[デスクトップレイアウト（Desktop Layout）]）

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「デスクトッププロパティのカスタマイズ」および「チームレベルでのデスクトッププロパティのカスタマイズ」の項を参照してください。

## クラウド接続の設定

このリリースでは、管理者が Finesse 管理コンソールで Cloud Connect サーバの設定を構成して、Cisco Webex のエクスペリエンス管理のような Cisco Cloud Services に接続できます。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「クラウド接続サーバーの設定」のセクションを参照してください。

## WebProxy サービス

このリリースでは、WebProxy サービスを Finesse サーバに導入し、SSL ターミネーションとキャッシュ サービスを Finesse サービスに追加して、遅延の減少とパフォーマンスの向上を実現しています。詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「WebProxy サービス」の項を参照してください。

ガジェット開発者は、テスト目的で更新されたガジェット応答にアクセスするために、プロキシをバイパスする必要があります。詳細については、<https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide>にある『REST API 開発者ガイド』の「ガジェット開発のベストプラクティス」セクションを参照してください。

## セキュリティ バナー メッセージ

このリリースでは、管理者ページとデスクトップのサインインページでカスタムバナーメッセージを提供しています。次の CLI を使用して、管理者がカスタムセキュリティバナーメッセージを定義できます。

- `utils finesse set_property desktop desktopSecurityBannerMessage <value>`
- `utils finesse set_property admin adminSecurityBannerMessage <value>`

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「デスクトッププロパティ」セクションと「サービスプロパティ」セクションを参照してください。

## デスクトップログインの自動再試行

このリリースでは、デバイス関連のエラーによってデスクトップのログインが失敗したときに、デスクトップのログインを自動的に再試行できます。次の CLI を使用して、管理者はログインの再試行を有効にしたり、再試行の回数と間隔を秒単位で定義したりできます。デフォルトでは、このプロパティの値は true に設定されています。

- `utils finesse set_property desktop enableRetryLoginFeature {true/false}`
- `utils finesse set_property desktop loginFailureRetryAttempts <value>`
- `utils finesse set_property desktop loginFailureRetryInterval <value>`

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「デスクトッププロパティ」の項を参照してください。

## Finesse IP フォン エージェント証明書の管理

このリリースでは、ソリューション全体のセキュリティ上の脆弱性に対処するために、Finesse IP フォン エージェント (IPPA) に HTTPS を使用する必要があります。管理者は、以下の証明書と設定の変更を必ずインポートして、FIPPA 機能を使用する必要があります。

- Cisco Unified Communications Manager (CUCM) の証明書を、信頼ストアに **tomcat-trust** としてインポートします。
- Cisco Finesse 証明書を、**Phone-trust** として CUCM トラストストアにインポートします。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「Finesse IP 電話のエージェント証明書の管理」の項を参照してください。

## HTTP Secure サポート

このリリースは、HTTP Secure (HTTPS) のみをサポートしています。HTTP のサポートは、管理コンソール、デスクトップ (エージェントおよびスーパーバイザ)、Web サービス、デスクトップモジュール (ガジェット)、および Finesse IPPA で無効になっています。すべての HTTP 要求は、自動的に HTTPS にリダイレクトされます。

HTTPS アクセスの場合は、`https://FQDN of Finesse server:8445/` と入力します。

無効化されたポートの詳細については、Cisco Finesse の「[特記事項, on page 11](#)」を参照してください。

## HTTP/2 のサポート

このリリースは、デフォルトで HTTP/2 プロトコルをサポートしています。

## ログ収集機能の拡張

### 3rdpartygadget ログ ディレクトリ

このリリースでは、Finesse 3rdpartygadget サーバの情報、エラー、起動、およびシャットダウンに関するログを、3rdpartygadget のログ ディレクトリで提供しています。

### WebProxy サービスのログ

管理者は、**file get activelog webproxy recurs compress** CLI を使用して、Webproxy サービスのログを取得することができます。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html> にある『Cisco Finesse アドミニストレーションガイド』の「ログ収集」の項を参照してください。

### デスクトップユーザの自動ログ収集

次の CLI を使用して、管理者はエージェントとスーパーバイザのデスクトップ ログの自動収集スケジュールを、作成、リスト、および削除することができます。

```
utils finesse desktop_auto_log_collection {create|list|delete}
```

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html> にある『Cisco Finesse アドミニストレーションガイド』の「ログ収集のスケジュール」の項を参照してください。

## コマンドの設定

このリリースでは、次の CLI が導入されています。

### コール変数のロギング

次の CLI を使用して、管理者はコール変数のロギングを有効または無効にできます。

- **utils finesse set\_property webservices logCallVariables** {true|false}
- **utils finesse set\_property fippa logCallVariables** {true|false}

### X.509 証明書の信頼検証の実施

管理者は、**utils finesse set\_property webservices trustAllCertificates** を *false* に設定して、X.509 CA または自己署名証明書の検証を有効にすることができます。

### セカンダリ リソースのプリロード

管理者は、`utils finesse set_property desktop preLoadSecondaryResources` を `true` に設定することで、セカンダリ サーバからの静的リソースのプリロードを有効化し、フェールオーバーの速度を上げることができます。

### XMPP ソケットおよび BOSH/WebSocket (HTTP)

管理者は、`utils finesse set_property webservice enableInsecureOpenfirePort` を `true` に設定して、Cisco Finesse Notification Service のセキュアでない XMPP ポート (5222) および HTTP-BOSH/WebSocket ポート (7071) を有効にすることができます。

### WebProxy サービス

次の CLI を使用して、管理者は WebProxy サービスによって生成されたログに対し、`access-log-level` および `log-severity` の消去や設定を行うことができます。

- `utils webproxy cache clear`
- `set webproxy access-log-level`
- `set webproxy log-severity`
- `show webproxy access-log-level`
- `show webproxy log-severity`

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html> にある『Cisco Finesse アドミニストレーションガイド』の「Cisco Finesse CLI」の項を参照してください。

## REST API

このリリースでは、次の API が導入されています。

- CompressedClientLog: Post Compressed Log to Finesse
- クラウド コネクトの設定: Get
- クラウド コネクトの設定: Set
- クラウド コネクトの統合: Delete
- TeamResource: Get Reason Codes
- TeamResource: Get Wrap-Up Reasons
- TeamResource: Get Media Properties Layouts
- TeamResource: Get Phone Books
- TeamResource: Get Workflows



## REST API 応答のキャッシュ

ログインパフォーマンスを向上させるために、Finesse webproxy は、次の REST API 応答をキャッシュします。

- ChatConfig
- TeamResource API には、理由コード、ラップアップの理由、メディア プロパティのレイアウト、電話帳、およびワークフローが含まれています。TeamResource API の応答は、チームレベルでキャッシュされます。

詳細については、<https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide> にある『REST API 開発者ガイド』を参照してください。

## JavaScript API

このリリースでは、次の API が導入されています。

- `finesse.shortcutkey.ShortcutKeyService`
- `finesse.utilities.DesktopCache`

詳細については、<https://developer.cisco.com/docs/finesse/#!javascript-library> にある JavaScript ライブラリを参照してください。

## 更新済み機能

### セキュリティ機能の拡張

このリリースでは、セキュリティについて次の変更が実装されています。

- デフォルトでは、Cisco Finesse Notification Service の安全でない XMPP ポート 5222 および BOSH/WebSocket (HTTP) ポート 7071 は無効になっています。  
ポートの有効化の詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html> にある『Cisco Finesse アドミニストレーションガイド』の「サービスプロパティ」の項を参照してください。
- X.509 証明書を必ず検証する必要があります。次の期限切れでない有効な X.509 CA または自己署名証明書を、必ず Cisco Finesse のトラストストアにインポートする必要があります。
  - Cisco Finesse のノード証明書は、デフォルトで利用可能です。期限切れの証明書は無効であるため、管理者は証明書の有効性を確認する必要があります。
    - 期限切れでない有効な Cisco Finesse プライマリ証明書が、セカンダリ Cisco Finesse に存在する必要があります。
    - 期限切れでない有効な Cisco Finesse セカンダリ証明書が、プライマリ Cisco Finesse に存在する必要があります。

- CUCM 証明書を、プライマリとセカンダリの両方の Finesse ノードにインポートしてください。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「セキュリティ機能の拡張」のセクションを参照してください。

## ログインのエラーメッセージ

このリリースでは、Finesse デスクトップでのログイン失敗に対して、詳細なエラーメッセージが表示されます。これにより、管理者はログを参照しなくても、クライアントのログイン失敗に対応できます。

エラーメッセージは、UI と API の両方で更新されます。

- Finesse デスクトップ UI は、サインインシナリオの CTI 操作によって提供される第 2 レベルのテキストエラーメッセージを含むように更新されます。

周辺機器のエラーコード 12004 と 12005 は、デスクトップのサインイン再試行に置き換えられます。サインインの再試行が失敗すると、CTI 操作によって提供される第 2 レベルのテキストエラーメッセージが表示されます。

- Cisco Finesse API ペイロードは、CTI 操作によって提供される周辺機器エラーコードと第 2 レベルテキストエラーメッセージを含めるように更新されています。新しく追加されたパラメータは次のとおりです。

- peripheralErrorCode
- peripheralErrorMsg
- peripheralErrorText

詳細については、<https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide>にある『REST API 開発者ガイド』の「Cisco Finesse API エラー」のセクションを参照してください。

例

```
13036 PERERR_GW_E_JTAPIOBJ_PERFORMANSWERCALL_NO_TERMINAL_CONNECTION
```

```
1st Level Text = 'JTAPI Gateway - Error on ANSWER operation'
2nd Level Text = 'The routine performAnswerCall in class JTapiObj got a null connection from a call to 'findTerminalConnection'
```

## [チームパフォーマンス (Team Performance)] ガジェット

Team Performance ガジェットでは、スーパーバイザは [検索 (Search)] ボックスを使用し、エージェントの詳細をエージェント名、状態、内線番号などの検索条件によって絞り込むことができます。

## キュー統計のサポート

キュー統計でサポートされているエージェントとスーパーバイザの最大数が、1500 から 2000 に増加しました。

## 電話帳の連絡先の制限事項

すべての電話帳におけるエージェントあたりの連絡先の最大数が、1500 から 6000 に増加しました。

## REST API の変更

次のペイロードが更新されました。

- **MediaPropertiesLayout API:** `uiEditable` ペイロードは、コール変数の値が Finesse デスクトップ（エージェントおよびスーパーバイザ）で編集可能かどうかを示します。

### 電話帳の連絡先の制限事項

エージェントあたりの連絡先の最大数が、電話帳と連絡先の API で、1500 から 6000 に増加されました。

詳細については、<https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide> にある『REST API 開発者ガイド』を参照してください。

## JavaScript API の変更点

次の関数が更新されました。

- **ContainerServices:** `collapseMyGadget` 関数と `expandMyGadget` 関数は、それぞれガジェットのコンテンツを非表示および表示します。
- **DialogBase:** `updateCallVariables` 関数は、ダイアログのコール変数を更新します。

詳細については、<https://developer.cisco.com/docs/finesse/#!javascript-library> にある JavaScript ライブラリを参照してください。

## 特記事項

### ポートの詳細

このリリースでは、次のポートに対するサポートが無効になっています。

- **tomcat (HTTP) :** 80、8082 に表示されません: HTTP を使用して Finesse にアクセスすると、301 HTTP リダイレクト ステータス応答がセキュア ポート 8445 に発行されます。

- BOSH/WebSocket (HTTP) : 7071 および XMPP: 5222。これらのポートは、デフォルトで無効になっています。これらのポートを有効にするには、**utils finesse set\_property webservices enableInsecureOpenfirePort** を *true* に設定します。

詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「サービスプロパティ」の項を参照してください。

### アップグレード時の証明書の削除

アップグレードが正常に行われると、シスコによって未承認となった CA はプラットフォームのトラストストアから削除されます。ただし、必要に応じて削除した CA を再度追加することができます。

- Cisco がサポートする CA のリストについては、<https://www.cisco.com/security/pki>にある「シスコが信頼する外部ルートバンドル」を参照してください。
- 証明書の追加方法の詳細については、<https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-tech-notes-list.html>にある『CUCM 証明書管理と変更通知』の「新しい Tomcat 信頼証明書の挿入」の項を参照してください。

### ネットワーク帯域幅

セカンダリ側のリソースは、迅速なフェールオーバーのためにプリロードされます。管理者は、ネットワーク帯域幅の使用量を増やさないために、ブラウザ固有の設定ガイドラインに従う必要があります。

最適なフェールオーバー パフォーマンスを確保するための展開方法とガイドラインの詳細については、<https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>にある『Cisco Finesse アドミニストレーションガイド』の「デスクトップ フェールオーバー最適化のガイドライン」の項を参照してください。

## 廃止された機能

次の GET User API は廃止されています。これらの API は下位互換性のために用意されており、TeamResource API と比較するとパフォーマンスが劣ります。

- ReasonCode
- WrapUpReason
- MediaPropertiesLayout
- PhoneBook
- WorkFlow

## 削除され、サポートされない機能

### コンテキストサービスの管理

コンテキスト サービス管理の機能は、Cisco Finesse 管理コンソールから削除されています。

### HTTP のサポート

Cisco Finesse の HTTP サポートは削除されています。ユーザは、HTTPS を介して Cisco Finesse と安全に通信できるようになりました。

次の CLI が削除されています。

- `utils finesse application_https_redirect status`
- `utils finesse application_https_redirect enable`
- `utils finesse application_https_redirect disable`

## サードパーティ ソフトウェアへの影響

なし。

