日本語ドキュメント

Model Vista MCP Module

Model Vista MCP Module のユーザードキュメント。

2026年10月30日

目次

01 MCP Module

02 AI クライアントの接続

03 モデルの読み込みとステートの選択

04 ビューと XR の操作

05 アプリケーションツールリファレンス

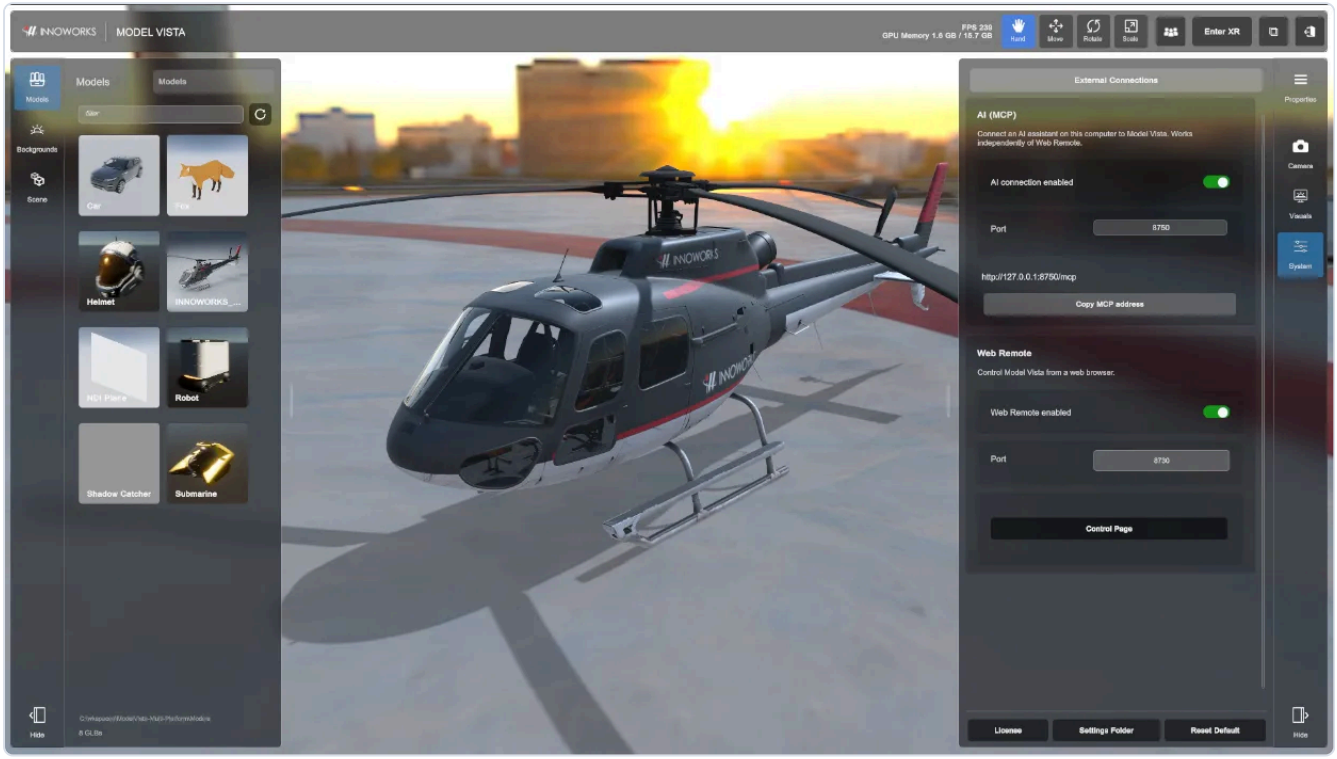
06 追加の実行時操作の検出

07 MCP 接続のトラブルシューティング

MCP Module

Model Vista MCP Module は、Model Context Protocol（MCP）を使って AI アシスタントを実行中の Model Vista アプリケーションに接続する、別途ライセンスが必要な Module です。モデルの確認、Model Vista ステートの選択、シーンの調整、現在の表示のキャプチャをアシスタントに依頼できます。

アプリケーションでは、接続は **System > External Connections** の **AI (MCP)** として表示されます。**Web Remote** とは独立して動作します。



要件

対応する Model Vista バージョン

アプリケーション	対応バージョン
Model Vista for Windows、HDRP デスクトップおよび XR	MCP Module を含む Model Vista 2.0.6

追加要件

- **System > License** で有効にした MCP Module の利用権限。この画面では **AI (MCP)** と表示されます。
- Model Vista と同じコンピューター上で動作する、Streamable HTTP MCP 対応の AI クライアント。

- レビュー対象のモデルおよび背景にアクセスできる、実行中の Model Vista セッション。

MCP Module の利用権限には、他の Module の利用権限は含まれません。それぞれの操作には、該当する利用権限と実行環境での対応が必要です。

クイックスタート

1. 接続を有効にして AI クライアントを設定します。
2. シーンを変更せずに、読み込み済みモデルを一覧表示するよう依頼します。
3. 検出された名前を使ってモデルを読み込み、ステートを選択します。
4. ビューと XR を操作するか、現在のアプリケーション表示をキャプチャします。
5. 変更を依頼するたびに、返された状態を確認します。

一般的な操作にはアプリケーションツールリファレンス、追加の実行時操作にはVariable Exposer リファレンス、接続や操作の問題にはトラブルシューティングを参照してください。

AI クライアントの接続

AI (MCP) の有効化

1. Model Vista を起動します。
2. **System > License** を開き、**AI (MCP)** の利用権限があり、有効になっていることを確認します。
3. **System > External Connections** を開きます。
4. **AI (MCP)** で **AI connection enabled** を有効にします。
5. アドレスが表示されるまで待ち、**Copy MCP address** を選択します。

既定のアドレスは次のとおりです。

```
http://127.0.0.1:8750/mcp
```

接続の初期値は無効です。有効化設定とポートは Web Remote とは別に保存されます。利用権限がない場合、AI (MCP) パネルは表示されません。接続を有効にしてもライセンス要件を回避することはできません。

クライアントへのサーバー追加

1. AI クライアントの MCP サーバー設定を開きます。
2. **Streamable HTTP** を使用する **Model Vista** という名前のサーバーを追加します。
3. コピーしたアドレスをサーバー URL として貼り付けます。
4. 接続し、会話で Model Vista のツールを利用できるようにします。
5. 「Model Vista MCP を使い、読み込み済みモデルを一覧表示してください。何も変更しないでください」と依頼します。

接続したクライアントでは、`list_loaded_models` や `variable_exposer_list_fields` などのツールを検出できます。読み込み済みモデルの一覧が空の場合は、シーンにモデルが読み込まれていないことを示します。接続の失敗を意味するものではありません。

リスナーはローカルコンピューターからの接続を受け付けます。クライアント、またはそのローカル接続コンポーネントを同じコンピューター上で実行してください。MCP アドレスは Web Remote の **Control Page** アドレスとは別です。

クラウドを使用する AI クライアントは、ツールの結果やスクリーンショットをモデルサービスへ送信する場合があります。モデルの内容に適したクライアント設定を選択してください。

ポートの変更と切断

使用されていない `1` から `65535` の **Port** を入力します。ポートを変更するとリスナーが再起動するため、新しいアドレスをコピーし、クライアントを再接続してください。同時に起動する各アプリケーションインスタンスには、異なるポートを割り当てます。

MCP 呼び出しの受け付けを停止するには、**AI connection enabled** を無効にします。**License** で AI (MCP) を無効にした場合もリスナーは停止します。接続を停止しても、すでに始まったアプリケーション操作はキャンセルされません。

一時的な設定で起動

起動引数は、その起動中に限り、保存済みの接続設定より優先されます。

Terminal window

```
& '.\Model Vista.exe' --modelvista-mcp --modelvista-mcp-port=8750
```

MCP を無効にして起動するには `--modelvista-no-mcp` を使用します。これらの引数は保存済み設定を変更せず、MCP Module の利用権限も引き続き必要です。

モデルの読み込みとステートの選択

モデルの検索と読み込み

1. アシスタントに `list_library_models` を呼び出し、利用できるモデル名を表示するよう依頼します。
2. 結果から一つ選び、`load_model` で読み込むよう依頼します。
3. 読み込みの完了を待ちます。処理中に同じ読み込みを再送しないでください。
4. `list_loaded_models` を呼び出し、新しいシーンインスタンスを確認するよう依頼します。

ライブラリアセット ID は読み込めるファイルを識別し、読み込み済みモデル ID はシーン内のインスタンスを識別します。各操作に適した ID を使用してください。同じ名前のモデルが複数ある場合は ID を使用します。

`list_libraries` と `select_library` は閲覧するライブラリを変更します。ライブラリの選択だけでは、シーンモデルの読み込みや削除は行われません。

ステートの確認と適用

1. `list_loaded_models` でモデルを選びます。
2. 読み込み済みモデル ID を指定して `list_model_states` を呼び出します。
3. 返されたグループとオプションを選び、`set_model_state` で適用するよう依頼します。
4. `applied` の値と、選択済みオプションの読み戻し結果を確認し、見た目も確認します。

例えば「このモデルのステートグループとオプションを一覧表示してください。変更する前に選択肢を示してください」と依頼します。オプションを選んだ後は「そのオプションを適用し、適用後の選択ステートを報告してください」と依頼します。

`set_model_state` の引数には文字列を使用します。

```
{
  "model": "<loaded-model ID from list_loaded_models>",
  "group": "<group ID or exact name from list_model_states>",
  "option": "<option ID or exact name from list_model_states>"
}
```

ステートグループは、表示・マテリアル・アニメーション・名前付きビューなどに関する、モデル内のオプションのグループです。標準の GLB には Model Vista ステートが含まれない場合があります。すべてのモデルにドア、ローター、カメラビューがあると仮定せず、利用できるオプションを検出してください。

シーンの調整

表示やトランスフォームを変更する前に `get_model` を読み取ります。`set_model_transform` の位置はワールド座標のメートル、回転はワールド座標の度、スケールはローカル値です。相対スケールは各軸に乗算されます。`unlock: "true"` を明示的に指定しない限り、トランスフォームのロックは維持されます。

`set_background` の前に `list_backgrounds` を使用します。環境やグラフィックス設定を変更する前に `get_environment` または `get_graphics_options` を読み取り、返された利用可否と適用状態を確認します。

`delete_model` はシーンインスタンスを削除し、ライブラリの元ファイルは保持します。削除を依頼する前に対象インスタンスを明示し、操作後に読み込み済みモデルの一覧を更新してください。

ビューと XR の操作

デスクトップビューのフレーミング

1. アシスタントに `get_xr_mode` を読み取るよう依頼します。
2. デスクトップモードで、`list_loaded_models` からモデルを選びます。
3. `front`、`back`、`left`、`right`、`top`、`bottom` のいずれかを指定して `set_camera_view` を依頼します。
4. `get_camera` を読み取り、結果を確認します。

これらのビューはモデルのローカル軸に従います。+Z が正面、+X が右、+Y が上です。モデルの形状から意味上の正面を推測するものではありません。

現在の視線方向を維持してモデルを画面に収めるには `frame_model` を使用します。カメラのアニメーション移動が始まるため、リクエストが受け付けられたことは、カメラが最終位置に到達したことを意味しません。

`move_camera` は現在のカメラ方向を基準にメートル単位で移動します。`rotate_camera` はカメラの向きを変更するか、現在の注視点の周囲を周回します。`set_camera_fov` は `15` から `90` 度の垂直視野角を受け付けます。

XR レビューでの MCP 使用

1. `get_xr_mode` を読み取り、XR を利用できることを確認します。
2. XR に入る場合は `enabled: "true"` を指定して `set_xr_mode` を依頼します。
3. 読み戻しで要求したモードになっていることを確認し、切り替え処理が完了するまで待ちます。
4. レビュー中に利用できるモデル、ステート、環境、実行時 UI のツールを使用します。

デスクトップカメラの位置操作は XR では利用できず、ヘッドセット利用者の位置を変更するものではありません。作成済みの名前付きビューが必要な場合はモデルのステートを指定し、XR モードやモデルのトランスフォームを維持したい場合はその条件を明示してください。

XR を終了するには `enabled: "false"` を指定して `set_xr_mode` を依頼し、デスクトップカメラ操作の前にモードを再確認します。

操作パネルの表示とキャプチャ

`list_windows` で利用できる実行時パネルを検出します。`show_window` でパネルを開きます。例えば `window: "system"` と `section: "external-connections"` で接続設定を開けます。パネルを開いた後に `list_ui_scrolls` を使用し、返された領域 ID を指定して `scroll_ui` を呼び出します。

見た目の結果を確認させたい場合は、アシスタントに `capture_screenshot` を呼び出すよう依頼します。アプリケーションの UI が含まれ、XR ではデスクトップミラーを使用します。

- `save: "true"` が既定値です。`{PersistentDataPath}/Screenshots` にフル解像度の PNG を保存し、保存先パスと画像プレビューを返します。
- `save: "false"` はフル解像度ファイルを保存せず、プレビューのみを返します。

プレビューは圧縮され、保存画像より小さい場合があります。アニメーションは、時間を空けてキャプチャした画面を比較してください。`get_model_animation` は再生クロックを読み取りますが、見た目の動きが正しいことを証明するものではありません。

アプリケーションツールリファレンス

使用前に接続先サーバーのツールを検出してください。利用可否と操作の成否は、実行中のアプリケーション、読み込んだコンテンツ、動作モード、ライセンス機能に依存します。

アプリケーションツールの引数は文字列です。数値や真偽値も `"true"`、`"10"`、`"0.5"` のように指定します。必須・任意の引数は各ツールの検出されたスキーマに従ってください。Variable Exposer ツールは、それぞれのスキーマを使用します。

ライブラリ、モデル、ステート

ツール	用途
`list_libraries` / `select_library`	モデルまたは背景のライブラリを検出して選択します。選択は閲覧対象を変更します。
`list_library_models` / `load_model`	元アセットを検出し、ライブラリアセット ID または一意の名前で読み込みます。
`list_loaded_models` / `select_model`	シーンインスタンスを検出し、実行時の操作対象を選択します。
`get_model`	表示、トランスフォーム、選択、ロックを読み取ります。
`set_model_visibility`	通常の表示ルールに従って、読み込み済みモデルを表示・非表示にします。
`set_model_transform`	`position`、`rotation`、`scale` を `absolute` または `relative` モードで変更します。
`delete_model`	シーンインスタンスを削除します。ライブラリの元ファイルは保持されます。
`list_model_states` / `set_model_state`	ステートのグループとオプションを検出して適用し、選択結果を読み戻します。
`get_model_animation`	再生を変更せずに再生クロックをサンプリングします。

カメラと XR

ツール	用途
`get_camera`	デスクトップカメラの位置・回転とレンズを読み取ります。
`set_camera_view`	モデル軸に基づく六つのビューからモデルをフレーミングします。
`frame_model`	現在の視線方向でモデルを画面に収める移動を開始します。
`move_camera` / `rotate_camera`	デスクトップカメラの移動、見回し、周回を行います。
`set_camera_fov`	透視カメラの垂直視野角を設定します。
`get_xr_mode` / `set_xr_mode`	XR の状態を読み取り、開始・終了し、切り替え状態を確認します。

カメラ操作にはデスクトップモードが必要です。位置操作とキャプチャの動作はビューと XRを参照してください。

実行時 UI とスクリーンショット

ツール	用途
`list_windows` / `show_window`	実行時パネルを検出して開き、必要に応じて設定セクションを選択します。
`hide_window`	パネルまたは `all` を非表示にします。動作は現在のインターフェース構成に従います。
`show_library`	`models` または `backgrounds` のサムネイルを表示します。
`list_ui_scrolls` / `scroll_ui`	表示中のスクロール領域を検出し、現在の領域 ID でスクロールします。
`capture_screenshot`	アプリケーションの画像プレビューを返し、任意で PNG を保存します。

背景、グラフィックス、Turntable

ツール	用途
`list_backgrounds` / `set_background`	背景を検出し、返された ID または一意の名前で読み込みます。
`get_environment` / `set_environment`	対応する設定を取得し、返された範囲内で1項目を変更します。
`get_graphics_options` / `set_graphics_option`	対応するグラフィックス操作を検出し、変更後の実際の状態を読み戻します。
`get_turntable`	割り当て、再生、角度、速度、利用可否を読み取ります。
`assign_model_to_turntable`	読み込み済みモデルを割り当て、`accepted` と `confirmed` を確認します。
`release_turntable_model`	モデルを読み込んだまま割り当てを解除します。
`set_turntable`	`play`、`reverse`、`pause`、`angle`、`speed` を設定します。MCP の速度は `0` から `30` の度/秒です。

ネットワーク経由で行う割り当て・解除は、リクエストが受理された後も処理中の場合があります。実際の割り当てが要求と一致するまで `get_turntable` を読み取ってください。安定した読み戻しが必要な角度設定では、先に再生を一時停止します。

追加の実行時操作の検出

アプリケーションツール以外の実行時操作が必要な場合は Variable Exposer ツールを使用します。実行中のセッションに登録されたフィールド、関数、インスタンスのバインディングを検出します。

安定した検出用ツール

ツール	引数と用途
`variable_exposer_list_fields`	任意の `path` フィルターで、フィールドとメタデータを一覧表示します。
`variable_exposer_get_field`	`identifier` と、インスタンスの場合は `targetId` を指定し、値と選択肢を読み取ります。
`variable_exposer_set_field`	`identifier`、`value`、必要に応じてインスタンスの `targetId` を指定し、変更可能なフィールドに書き込みます。
`variable_exposer_list_functions`	呼び出せる関数と記述情報を一覧表示します。
`variable_exposer_get_function`	`identifier` と、必要に応じてインスタンスの `targetId` を指定し、関数のメタデータを確認します。
`variable_exposer_list_instances`	`identifier` を指定し、そのパスの対象 ID とラベルを検出します。
`variable_exposer_invoke_function`	`identifier`、任意のインスタンス `targetId`、検出されたペイロード型に合う `arguments` を指定します。
`variable_exposer_get_job`	`jobId` を指定し、バックグラウンド呼び出しの状態と結果を読み取ります。

値を変更する前の検出

1. フィールドと関数を一覧表示します。関連する操作が異なるカテゴリにある場合もあります。
2. フィールドの選択肢や関数の記述情報で、必要な識別子とペイロードを確認します。
3. インスタンスに紐づくパスでは `variable_exposer_list_instances` を呼び出し、返された `instanceId` を `targetId` として使用します。
4. 現在の値を読み取り、意図した変更を依頼します。
5. 変更後の値やアプリケーション状態を読み取り、結果を確認します。

識別子やインスタンス ID を推測しないでください。モデルの読み込みや削除で操作やステートのインスタンスが変わるため、検出をやり直します。書き込み可能なフィールドに、独立した名前付き関数があるとは限りません。

アプリケーションツールの文字列引数とは異なり、`value` と関数の `arguments` には JSON のスカラー、配列、オブジェクトを使用できます。検出されたスキーマに合わせ、同じパスに複数のインスタンスがある場合は `targetId` を指定します。

長時間の関数を追跡

`variable_exposer_invoke_function` で `background: true` を指定すると、すぐに `jobId` を受け取れます。ID を保持し、状態が `succeeded` または `failed` になるまで `variable_exposer_get_job` を呼び出します。ジョブの実行中に関数を再度呼び出さないでください。

実行状態と返された結果の両方を確認します。完了した関数が `false` を返す場合や、要求を受け付けた後もアプリケーションで処理が続く場合があります。通信の成功だけで判断せず、読み戻しで意図した変更を確認してください。

ジョブ ID はリスナーの再起動で失効し、完了済みジョブは容量に上限のあるキャッシュから削除される場合があります。MCP を停止しても開始済みの処理は取り消されません。ID が失効した場合は、再試行する前に実際のアプリケーション状態を確認します。

MCP 接続のトラブルシューティング

症状	確認事項
AI (MCP) パネルがない	実行中のアプリケーションに MCP Module が含まれ、**System > License** で **AI (MCP)** のライセンスが有効になっていることを確認します。
「Unable to start MCP」と表示される	ポートが使用中の場合があります。未使用のポートを選び、新しいアドレスをコピーして再接続します。
クライアントが接続できない	Model Vista を起動したまま **AI connection enabled** を有効にし、同じコンピューター上で表示された `/mcp` URL と Streamable HTTP を使用します。
Web Remote は動くが MCP は動かない	独立した AI (MCP) の有効化設定とポートを確認します。Control Page の URL は MCP の URL ではありません。
読み込み済みモデルの一覧が空である	`list_library_models` でアセットを検出し、選んだ結果を読み込みます。
モデルやステート名が曖昧	再検出し、重複した名前ではなく返された ID を使用します。
ツールやフィールドがない	再接続してツールを更新し、アプリケーションツール、フィールド、関数、インスタンスを確認します。読み込み済みモデルや別の利用権限が必要な操作もあります。
XR でデスクトップカメラ操作が失敗する	XR を維持したまま、利用可能なモデル・ステート操作を使うか、意図的に XR を終了し、読み戻し結果を待ってからカメラ操作を行います。
変更が拒否される、または `applied` が false	処理中の状態、トランスフォームのロック、表示ルール、グラフィックスの利用可否、ライセンスを確認します。再試行前に実際の状態を読み取ります。
Turntable の割り当てが受け付けられたが確認されない	`get_turntable` を読み取ります。ネットワークでの受け付けは実際の割り当てより先になる場合があります。
ステートが選択されたがアニメーションが見えない	`get_model_animation` と描画された表示を確認します。選択結果と進む再生クロックだけでは、見た目の動きは確認できません。
スクリーンショットがタイムアウトする	アプリケーションまたは Game view を表示し、フレームが描画されてから再試行します。
ジョブ ID が見つからない	リスナーの再起動や完了済みジョブの失効が考えられます。操作を繰り返す前にアプリケーション状態を確認します。

再起動やライセンス変更後

ポート変更、Model Vista の再起動、MCP の利用権限の復帰後は AI クライアントを再接続します。以前のインスタンス ID を使用する前に検出を更新してください。

長時間の操作がクライアントでタイムアウトした場合は、Model Vista でまだ処理中か、完了したかを確認します。クライアントのタイムアウトは、操作がキャンセルされたことを意味しません。

問題報告

Model Vista のバージョン、AI クライアントとバージョン、選択した MCP ポート、デスクトップ/XR モード、正確なツール名と引数、返されたエラーや状態、期待した結果を含めます。モデルの読み込み、リスナーの再起動、ライセンス変更、XR 切り替え後の問題かどうかも説明してください。共有するログやスクリーンショットから認証情報と非公開のモデル情報を除いてください。