

# トリバム API ドキュメント

# trivum API

trivum HTTP インターフェイス .....	1
1. コマンド .....	1
1.1. ZoneCommand .....	1
1.2. ゾーンソースの設定 .....	3
1.3. ゾーン属性を設定 .....	4
1.4. トリバム お気に入り .....	4
1.5. トリバムプレイリスト .....	5
1.6. TuneIn のお気に入り .....	5
1.7. FM プリセット .....	5
1.8. NAS のステータスと制御 .....	6
1.9. グループ経営 .....	6
1.10. ページング .....	8
2. インタラクティブミュージックセレクション .....	9
3. ゾーンステータスの取得 .....	9
3.1. 同期 .....	9
3.2. 非同期 .....	10
3.3. 付録: Visu クライアント アプリケーションの回路図の例 .....	11
3.3.1. シングルスレッドアプリケーション .....	11
3.3.2. 2 スレッドのアプリケーション例 .....	11

# trivum HTTP インターフェイス

2023 年 7 月 25 日

trivum HTTP インターフェイスは、Web ブラウザで簡単にテストできるリクエストを受け取り、XML 形式で応答を返します。

## 1. コマンド

### 1.1. ZoneCommand

ゾーンをオフにしたり、音量を変更したりするなどの基本的な操作を実行できます。

```
/xml/zone/runCommand.xml?zone=@zoneId&command=commandNumber
```

#### ゾーン ID

ゾーンの ID。可能な ID のリストについては、

[Automation /trivum API](#) の下の Web 構成を調べるか、以下の `getAll.xml` の例を参照してください。



一部のアクチュエータは、内部の未使用の構成ファイルが原因で、@0 ではなく @1 によって最初のゾーンにアドレス指定する場合があります。これを修正するには、次の方法で構成全体をリセットできます:

システム/バックアップ/復元/すべての構成データをリセット

@0 の代わりにゾーン名を指定することもできます。特殊文字が含まれている場合は、% を使用して書き換えます。

```
/xml/zone/runCommand.xml?zone=living%20room&command=...
```

#### コマンド番号

これは、次の可能な値を持つ数値コマンドです。

ZONECMD_POWER_OFF	001	
ZONECMD_MUTE	002	toggle mute
ZONECMD_MUTE_ON	680	since v9.29
ZONECMD_MUTE_OFF	681	since v9.29
ZONECMD_VOLUME_INC	003	
ZONECMD_VOLUME_DEC	004	
ZONECMD_VOLUME_INC2	009	
ZONECMD_VOLUME_DEC2	010	
ZONECMD_VOLUME_INC5	011	
ZONECMD_VOLUME_DEC5	012	

ZONECMD_ALLOFF	015	
ZONECMD_SNOOZE	017	
ZONECMD_LOCAL_SOURCE	019	if present (LineIn)
ZONECMD_USE_PREV_SOURCE	029	see Zones / zone / KNX/HTTP sources
ZONECMD_JOIN	030	
ZONECMD_UNJOIN	031	
ZONECMD_USE_NEXT_SOURCE	041	see Zones / zone / KNX/HTTP sources
ZONECMD_USE_NEXT_ZONE	042	see Zones / zone / KNX/HTTP sources
ZONECMD_DEFAULT_STREAMING	050	if present
ZONECMD_DEFAULT_TUNER	051	if present
ZONECMD_VOLUME_DEC_1	080	
ZONECMD_VOLUME_DEC_10	089	
ZONECMD_VOLUME_INC_1	090	
ZONECMD_VOLUME_INC_10	099	
MULTIKEY_BASIC_FORWARD	400	skip to next track, preset
MULTIKEY_BASIC_BACKWARD	401	skip to prev. track, preset
MULTIKEY_BASIC_FASTFORWARD	402	
MULTIKEY_BASIC_FASTBACKWARD	403	
MULTIKEY_BASIC_PLAYPAUSE	406	
MULTIKEY_PLAY	431	
MULTIKEY_PAUSE	432	
MULTIKEY_STOP	433	
MULTIKEY_STATION_DOWN	490	
MULTIKEY_STATION_UP	491	
MULTIKEY_NEXT_ALBUM	493	
MULTIKEY_PREVIOUS_ALBUM	494	
MULTIKEY_NEXT_PLAYLIST	495	
MULTIKEY_PREVIOUS_PLAYLIST	496	
ZONECMD_START_PAGING_1	500	
ZONECMD_START_PAGING_32	531	
ZONECMD_STOP_PAGING_1	550	
ZONECMD_STOP_PAGING_32	581	
ZONECMD_STOP_PAGING_ALL	599	
ZONECMD_PRESET_1	600	
ZONECMD_PRESET_7	606	
ZONECMD_GROUP_START_1	621	
ZONECMD_GROUP_START_8	628	
ZONECMD_GROUP_STOP	630	
ZONECMD_GROUP_STOP_1	631	
ZONECMD_GROUP_STOP_8	638	
ZONECMD_GROUP_STOP_ALL	639	
ZONECMD_STREAMING_NOPLAY	641	
ZONECMD_VOLUME_00	900	
ZONECMD_VOLUME_99	999	
ZONECMD_ROOM_VOLUME_00	1000	
ZONECMD_ROOM_VOLUME_99	1099	

例

電話	function
/xml/zone/getAll.xml	考えられるゾーン ID をすべてリストします。
/xml/zone/get.xml?zone=@0	単一ゾーンのステータスを取得します。オプションのパラメーターは次のとおりです: &addSourceBasicData &addSourceStatusData
/xml/zone/getSelection.xml?grouped	完全なグループ情報を含むゾーンをリストします。
/xml/zone/runCommand.xml?zone=@0&command=50	最初のゾーンをデフォルトのストリーミングに切り替えます。
/xml/zone/runCommand.xml?zone=@0&command=1	最初のゾーンをオフに切り替えます。
/xml/zone/runCommand.xml?zone=@0&command=15	すべてのゾーンをオフにします。
/xml/zone/runCommand.xml?zone=@0&command=680	ミュート on
/xml/zone/runCommand.xml?zone=@0&command=681	ミュートオフ

## 1.2. ゾーンソースの設定

短い名前でもゾーンソースを選択してください

/xml/zone/set.xml?zone=@0&source=@shortSourceName

### shortSourceName

テキスト	アクション	備考
a a1 a3	最初のアナログ入力 最初のアナログ入力 3 番目のアナログ入力	デバイス モデルに応じて、0 ~ 8 アナログ
p p5	最初の FM チューナー プリセット 5 番目の FM チューナー プリセット	デフォルトの FM チューナーがゾーンに設定されて いる必要があります。
f ` `f2	最初の trivum お気に入り 2 番目の trivum お気に入り	
y y2	最初の trivum プレイリスト 2 番目の trivum プレイリスト	
i i2	最初のチューンイン プリセット 2 番目のチューンイン プリセット	

テキスト	アクション	備考
s	ゾーンのデフォルトのストリームソース	最近の選択を再生中
t	ゾーンのデフォルトの FM チューナー	最近の再生周波数

例

API 呼び出し	備考
<code>/xml/zone/set.xml?zone=@0&amp;source=@a1</code>	最初のアナログ入力に切り替え
<code>/xml/zone/set.xml?zone=@0&amp;source=@t</code>	ゾーンのデフォルトの FM チューナーに切り替えて、最近の周波数を再生します
<code>/xml/zone/set.xml?zone=@0&amp;source=@p3</code>	ゾーンのデフォルトの FM チューナーとプレイステーションのプリセットに切り替えます 3
<code>/xml/zone/set.xml?zone=@0&amp;source=@f2</code>	ゾーンのデフォルトのストリーミングに切り替えて、trivum お気に入り 2
を再生します。 <code>/xml/zone/set.xml?zone=@0&amp;source=@i5</code>	ゾーンのデフォルトのストリーミングに切り替えて、TuneIn Webradio プリセット 5
C4 のみ: カード スロット n によるソースを使用します。(n >= 0)	<code>/xml/zone/runCommand.xml?zone=@0&amp;command=15</code>

### 1.3. ゾーン属性を設定

ボリューム、ミュート、バランス、低音など、ゾーンの基本的な値を変更します。

API 呼び出し	備考
<code>/xml/zone/set.xml?zone=@0&amp;volume=10</code>	ボリュームを設定 (0 ... 100)
<code>/xml/zone/set.xml?zone=@0&amp;action=1</code>	数値コマンドを実行する <code>/xml/zone/runCommand.xml</code> と同じです。この場合は <code>ZONECMD_POWER_OFF (1)</code>
<code>/xml/zone/set.xml?zone=@0&amp;balance=0</code>	バランスを -15 (左いっぱい) から 15 (右いっぱい) まで設定します
<code>/xml/zone/set.xml?zone=@0&amp;bass=-5</code>	低音の低減または強化を -15 から 15  まで設定します。 <code>/xml/zone/set.xml?zone=@0&amp;treble=5</code>

### 1.4. トリバム お気に入り

trivum 個のお気に入りを作成するには:

- NAS アルバムなどの音楽コンテンツを再生する
- 次に、右上の「...」を選択します
- 次に、"trivum お気に入り追加" を選択します。

trivum 個のお気に入りのリストを取得します。

`/api/v1/trivum/favorite.xml`

trivum のお気に入りを再生:

`/xml/zone/set.xml?source=@f1&zone=@0`

オプションを追加することもできます:

オプション	備考
<code>&amp;sequence=random-sequential</code>	ランダムな開始トラックを選択
<code>&amp;sequence=ランダム-ランダム</code>	永続的にランダムな順序で再生

## 1.5. トリバムプレイリスト

trivum 個のプレイリストのリストを取得します。

`/api/v1/trivum/playlist.xml`

trivum プレイリストを再生します。

`/xml/zone/set.xml?source=@y1&zone=@0`

オプションを追加することもできます:

オプション	備考
<code>&amp;sequence=random-sequential</code>	ランダムなトラックから開始するには
<code>&amp;sequence=ランダム-ランダム</code>	ランダムなトラックのみを再生するには

## 1.6. TuneIn のお気に入り

これらは、TuneIn ステーションの再生中に右上の「...」によって作成することもできます。

TuneIn のお気に入りのリストを取得します。

`/api/v1/tunein/favorite.xml`

TuneIn のお気に入りを再生します。

`/xml/zone/set.xml?source=@i1&zone=@0`

## 1.7. FM プリセット

FM プリセットを一覧表示します。

`/xml/system/getTunerStationList.xml`

C4 では、システム全体の FM プリセットのリストが表示されますが、FM チューナーカードごとにローカルプリセットは保存されません。

## 1.8. NAS のステータスと制御

API 呼び出し	備考
<code>/xml/system/getMusicCenterStatus.xml</code>	NAS ライブラリのステータスを取得
<code>/xml/system/scanMusicCenterShares.xml</code>	フル NAS スキャンを再実行

## 1.9. グループ経営

グループは、1 回の呼び出しで作成、変更、または削除できます。

```
/xml/zone/createGroup.xml?zone=zVisu&oldgroup=zMaster&members=++-----
```

パラメーター：

	名前	remark
zVisu	視覚化クライアントの現在のゾーンのインデックス	マスター
(両方のゾーンが現在異なるソースを再生している場合)	音楽を使用するグループマスターのインデックス。 +/-	どのゾーンがグループに参加すべきかをグラフィカルに伝える文字。たとえば、4 ゾーンシステムの場合は、4 文字以内で入力します (自動的に「-」で埋められます)。

例: 2 番目のゾーンが最初のゾーンの再生に加わります

- 最初のゾーンはストリームを再生しており、2 番目のゾーンは FM チューナーを再生しており、他のすべてのゾーンはオフになっています。
- 2 番目のゾーンは、最初のゾーンを持つグループに追加する必要があり、最初のゾーン (ストリーム) から音楽を引き継ぐ必要があります。

```
/xml/zone/createGroup.xml?zone=1&oldgroup=0&members=++--
```

結果: 2 番目のゾーンは、最初のゾーンと同じストリームの再生を開始します。

例: 最初のゾーンが 2 番目のゾーンの再生に参加します

- 最初のゾーンはストリームを再生しており、2 番目のゾーンは FM チューナーを再生しており、他のすべてのゾーンはオフになっています。
- 最初のゾーンを 2 番目のゾーンを持つグループに追加し、2 番目のゾーン (チューナー) から音楽を引き継ぐ必要があります。

```
/xml/zone/createGroup.xml?zone=0&oldgroup=1&members=++--
```

結果: 最初のゾーンは、2 番目のゾーンと同じ FM チューナーの再生を開始します。



これは、両方のゾーンが異なるソースを再生している場合、  
"oldgroup" がグループ参加後に再生する音楽を決定することを意味します。

例: 2 番目のゾーンはグループを脱退する必要があります

```
/xml/zone/createGroup.xml?zone=0&oldgroup=0&members=+---
```

ここで関連するのは、メンバー リストの + から - への変更です。

グループ内の音量レベルを変更する

グループ内では、ゾーンは通常、分離されたボリューム レベル  
を使用しませんが、ボリュームの変更はすべてのグループ メンバーに影響します。  
この相互依存関係は次の呼び出しによって処理されます。

```
/xml/zone/setVolume.xml
```

デフォルトでは、この呼び出しは絶対音量レベルを単に「設定」するのではなく、指定されたターゲット  
音量の方向に「少しずつ」進みます。これは、ビジュアライゼーションの + または -  
ボタンと併用するのが最適です。

API 呼び出し	備考
<pre>/xml/zone/setVolume.xml?id=@0&amp;volume=0</pre>	グループ全体のグループの音量を下げます。 ID はグループの任意のゾーン ID です。 すべてのゾーン メンバーの音量が数段階下がります。
<pre>/xml/zone/setVolume.xml?id=@0&amp;volume=99</pre>	グループ全体のグループ音量を上げます。 すべてのゾーン メンバーの音量が数段階上がります。
<pre>/xml/zone/setVolume.xml?id=@0&amp;groupMemberVolume=99</pre>	単一ゾーンのボリュームを段階的に増やします。 他のグループ メンバーには影響しません。
<pre>/xml/zone/setVolume.xml?id=@0&amp;groupMemberVolume=0</pre>	単一ゾーンのボリュームを段階的に下げます。 他のグループ メンバーには影響しません。
<pre>/xml/zone/setVolume.xml?id=@0&amp;stop</pre>	ボリュームステップを直ちに停止します。
<pre>/xml/zone/setVolume.xml?id=@0&amp;groupMemberVolume=50&amp;absolute</pre>	他のグループ メンバーから分離された単一ゾーンの 絶対 ボリュームを設定します。 (慎重に使用してください。)

グループ内の新しいボリューム レベル情報を取得するには、getChanges 呼び出しを行い、ボリューム  
ステータス リストを調べます。

```
/xml/zone/getChanges.xml?zone=@0&visuid=90&apiLevel=2&now
```

出力例、グループ化されている場合、ゾーン/ステータス: の下

```

<zone>
  ...
  <status>
    <volume>17</volume> - volume of zone making the getChanges call
    ...
    <group>
      <zone>0</zone>
      <volume>17</volume> - volume for zone id 0
    </group>
    <group>
      <zone>1</zone>
      <volume>26</volume> - volume for zone id 1
    </group>
    <group>
      ...
    </group>
    <groupMembers>2</groupMembers>
  </status>
</zone>

```

getChanges の詳細な説明については、[ゾーン ステータスの取得](#) を参照してください。

## 1.10. ページング

ページングは Web 構成で構成する必要があります。次に、次の呼び出しを使用できます。

ページング開始

[/xml/paging/start.xml](#)

パラメーター

名前	説明
<a href="#">id</a>	ページング ID、0 ~ 31
<a href="#">ボリューム</a>	オプション、5 ~ 100。 が指定されていない場合は、設定されたページング ボリューム レベルが使用されます。
<a href="#">autostoptime</a>	オプション、5 ~ 100 秒。 指定しない場合は、構成された停止設定が使用され ます。

例

[/xml/paging/start.xml?id=0&volume=10&autostoptime=10](#)

ページングは定義された時間が経過すると自動的に停止しますが、次のコマンドを呼び出すことでより早く停止することもできます。

[/xml/paging/stop.xml?id=0](#)

## 2. インタラクティブミュージックセレクション

次で始まります:

```
/xml/system/getWebTouchMenu.xml?this=music&zone=@0&visuid=90
```

これにより、次のようなレコードが生成されます。

```
<row>
  <type>action</type>
  <mode>menu</mode>
  <action>/xml/system/getWebTouchMenu.xml?which=trivumFavorites&keypad=4</action>
  <icon>/imgs/visuIconServiceFavorites_128px.png</icon>
  <text>trivum_20favorites</text>
</row>
```

次に、レコードごとに:

- ビジュアライゼーションでテキスト フィールドをデコードして表示します。  
\_20 は、ASCII コード 0x20 (スペース) を持つ文字を意味します。
- タッチされた場合、アクション URL を呼び出し、次のメニュー レベルを表示します。



特定のメニュー レベルの永続的な可用性に依存しないでください。  
特に音楽サービスで提供されるメニューは、時間の経過とともに変化する可能性があります。

## 3. ゾーン ステータスの取得

### 3.1. 同期

1 つの短い API 呼び出しでゾーンのステータスをポーリングします。

```
/xml/zone/getChanges.xml?zone=@0&visuid=90&apiLevel=2&now
```

パラメーター

名前	関数
<code>visuid</code>	外部視覚化インスタンスを識別する 1 ~ 99 の数字。 この API ドキュメント内の、 <code>visuid=90</code> はテストリクエストに使用されます。
<code>apiLevel</code>	常に 2 である必要があります。これにより、「keypad/Basic」の下に「button」xml オブジェクトが生成されます。 <code>now</code>

新しいゾーンのステータスを直ちに返し、接続を閉じるようにサーバーに指示します。 &now がない場合、呼び出しはタイムアウトになるまで、またはゾーンステータス情報が変更されるまでブロックされます。	reload=1
--	----------

コントロールユニットについて (ビジュアライゼーション)

visuid=90 でリクエストを送信すると、ID 90 の *Control Unit* オブジェクトがサーバーに作成されます。

*Control Units* の下の Web 構成で、現在のコントロールユニットのリストを取得できます。

最初のアクセスの後、ユニットは "Notconfigured" としてリストされます。たとえば、power" を短く押してオプション "Off" を設定するなどして構成を変更すると、その構成は *Configured* と呼ばれ、後でコントロールユニットリストをクリーンアップしてもこれは削除されません。

このユニットへのリクエストがない場合、しばらくすると、"currently inactive control units" の下にリストされます。

## 3.2. 非同期

これは、HTTP 呼び出しがすぐには返されず、何かに変更されるまでブロックされることを意味します。

例：

`/xml/zone/getChanges.xml?zone=@0&visuid=90&apiLevel=2`

「&now」がないことに注意してください。次のことが起こります。

最初の API 呼び出し時:

ID 90 のコントロールユニットが作成され、最初のゾーンにリンクされます。

API 呼び出しは、ゾーンの完全なステータスデータとともに直ちに返されます。

以降のすべての API 呼び出し:

既存のコントロールユニット 90 を再利用する。

API

呼び出しは、次の場合までブロックされる可能性があります。

- タイムアウトに達します (約 10 秒)。この場合、次のような応答が返されます:

```
<rows><system><timeout>1</timeout>
```

- または、ゾーン内のボリュームなど、何かに変更されるまで。

2 つの `getChanges` 呼び出しの間にサーバーで (多くの) ステータスデータが変更された場合、呼び出しはまったくブロックされず、すぐに新しいステータスが返されます。

タイムアウトを受け取ったら、すぐに `getChanges` を再実行してください。つまり、別の I/O スレッドなどで、ループ内で `getChanges` を際限なく実行できます。リクエストは変更された場合にのみ返されるため、サーバーで負荷の問題が発生することはありません。

タイムアウトを受信しない場合、つまり呼び出しがすぐに返されるか、数秒後（何かが変更された直後）に返された場合は、ステータス データを処理してから、getChanges 要求を再実行します。

## 3.3. 付録: Visu クライアント アプリケーションの回路図の例

### 3.3.1. シングルスレッドアプリケーション

これには、ソケットの応答データが存在するかどうかを (select() 呼び出し経由で) プログラミング言語でテストできる必要があります。

メインスレッド

- 開始: `/xml/zone/getChanges.xml?visuid=90&now` を送信します
- ループの開始: GUI を更新します。
  - ユーザーからの入力イベントを処理します。
  - 次のような同期コマンドを送信します:  
`/xml/zone/runCommand.xml?...`  
応答を受信し、rc を確認し、XML ステータス データを処理します (getChanges 応答と同じ)
  - 進行中の getChanges 呼び出しに対する応答データが存在するかどうかを確認します (C コード: ソケットでの select() 呼び出し)  
trivum サーバーからのデータが存在する場合:
    - `<userdata name="rc">0</userdata>` を探します。  
存在しない場合は、
      - \_ エラーを処理し、数秒待ちます。
      - タイムアウトではない場合
      - \_ プロセス XML 応答 (ステータス データ)Endif  
非同期呼び出し (送信のみ)  
`/xml/zone/getChanges.xml&visuid=90&onlyChanges`  
Endif
  - 1分以内にサーバーからデータが到着しない場合
    - 非同期呼び出し (送信のみ)  
`/xml/zone/getChanges.xml&visuid=90&onlyChanges`  
endif
  - ループを再実行します

### 3.3.2. 2 スレッドのアプリケーション例

別の I/O スレッドのソケットでブロッキング受信を実行したい場合に使用できます。

メインスレッド

- GUIを更新します。
- ユーザーからの入力イベントを処理します。
- 次のような同期コマンドを送信します:

`/xml/zone/runCommand.xml?...`

応答を受信し、rcを確認し、XMLステータスデータを処理します  
(getChanges 応答と同じ)

- ステータスデータとエラーをステータススレッドから受信します。
- このループを再実行します。

#### ステータススレッド

- 最初のループのIF:
  - `/xml/zone/getChanges.xml?visuid=90&now` を送信します  
ELSE
  - `/xml/zone/getChanges.xml?visuid=90&onlyChanges` を送信します
- 応答を受信します (これは最大 10 秒間ブロックされます)
- `<userdata name="rc">0</userdata>` を探します。  
これが存在しない場合は、エラーが発生します。  
エラーが発生したときに単にループを再実行するだけでなく、少なくとも数秒待ってからメインスレッドに通知してください。
- `<rows><system><timeout>1</timeout>` を探します。  
これが存在する場合
  - すぐにループを再実行します。  
他
  - 返信ステータスデータを処理し、新しいステータスデータをメインスレッドにコピーします。
- このループを再実行します。