



エージェント状態情報モジュール

Desktop.agentStateInfo モジュールは、エージェント関連の情報における最新のデータ更新をリッスンします。

例

```
import {
  Service
} from "@wxcc-desktop/sdk-types";

type LatestInfoData = {
  teamId ? : string;
  teamName ? : string;
  dn ? : string;
  status ? : string;
  subStatus ? : string;
  idleCodes ? : Service.Aqm.Configs.Entity[];
  wrapupCodes ? : Service.Aqm.Configs.Entity[];
  outDialRegex ? : string;
  isOutboundEnabledForTenant ? : boolean;
  isOutboundEnabledForAgent ? : boolean;
};
```

次の表に、ペイロードの詳細を示します。

名前	タイプ	説明	必須
teamId	文字列	チームの一意的識別子。	はい
teamName	文字列	チームの名前。	はい
dn	文字列	エントリポイントに対するダイヤル番号 (DN)。	はい
status	文字列	エージェントのログインステータス (例: LoggedIn)。	はい
subStatus	文字列	エージェントの対応可能ステータス (例: Available)。	はい

名前	タイプ	説明	必須
idleCodes	文字列	ルーティングされたリクエストをエージェントが受け付ける準備が整っていないことを示すアイドル理由コード。例：Idle、Coffee break、Meeting、Tea	はい
wrapupCodes	文字列	エージェントが顧客とのインタラクションを終了したことを示すラップアップ理由コード。	はい
outDialRegex	文字列	アウトダイヤルの正規表現の有効性。	はい
isOutboundEnabledForTenant	ブール値	テナントに対してアウトバウンド機能を有効にするかどうかを決定します。 <ul style="list-style-type: none"> • True：テナントに対してアウトバウンド機能を有効にします。 • False：テナントに対してアウトバウンド機能を無効にします。 	はい
isOutboundEnabledForAgent	ブール値	エージェントに対してアウトバウンド機能を有効にするかどうかを決定します。 <ul style="list-style-type: none"> • True：エージェントに対してアウトバウンド機能を有効にします。 • False：エージェントに対してアウトバウンド機能を無効にします。 	はい

- [メソッド \(2 ページ\)](#)
- [イベント \(4 ページ\)](#)

メソッド

latestData

最新のユーザー関連情報を取得します。

例

```
const latestData: LatestInfoData = Desktop.agentStateInfo.latestData;
```

戻り値

{Object} 最新のユーザー関連情報をリストします。

応答の例

```
{
  "agentName": "John Doe",
  "teamName": "Sales",
  "teamId": "964",
  "dn": "8895579172",
  "status": "LoggedIn",
  "subStatus": "Idle",
  "idleCodes": [{
    "id": "1646",
    "isDefault": false,
    "isSystem": true,
    "name": "Aux on Login"
  }],
  "wrapupCodes": [{
    "id": "2065",
    "isDefault": false,
    "isSystem": false,
    "name": "Wrapping up as customer disconnected"
  }],
  "outDialRegex": "([0-9a-zA-Z]+[-._])*[0-9a-zA-Z]+",
  "isOutboundEnabledForTenant": true,
  "isOutboundEnabledForAgent": true
}
```

stateChange(state, auxCodeIdArray)

ユーザーの状態の変更を取得します。

例

```
const state = await Desktop.agentStateInfo.stateChange({ state: s, auxCodeIdArray: "0"
});
```

パラメータ

名前	タイプ	説明	必須
state	文字列	現在のエージェントの状態。	はい
auxCodeIdArray	文字列	エージェントの状態を表す一意の識別子。	はい

戻り値

{Object} エージェントの状態の変更に関する情報をリストします。

応答の例

```
{
  "data": {
    "agentId": "7c867aa9-ec768-341a-b767-e5hd6ae7g701",
    "agentSessionId": "3d017488-527a-4e89-9313-5d4eb353c789",
    "auxCodeId": "0",
    "connectedChannels": [
      "38136791-cf79-4663-a5d0-39b0d094abf8"
    ],
    "eventType": "AgentDesktopMessage",
    "lastIdleCodeChangeTimestamp": null,
    "lastStateChangeReason": ""
  }
}
```

```

        "lastStateChangeTimestamp": 1612245463183,
        "orgId": "f111e3af-1a45-42ef-9erf-4562354b8a25",
        "status": "LoggedIn",
        "subStatus": "Available",
        "trackingId": "9008a4a0-651b-11eb-a97d-0d76c582d799",
        "type": "AgentStateChangeSuccess"
    },
    "orgId": "f111e3af-1a45-42ef-9erf-4562354b8a25",
    "trackingId": "notifs_aa848b62-4fcf-4d4c-95ad-1233bb7f899c",
    "type": "AgentStateChange"
}

```

fetchAddressBooks()

アドレス帳の詳細を取得します。

例

```
const books = await Desktop.agentStateInfo.fetchAddressBooks()
```

戻り値

{Array} オブジェクトの配列。

応答の例

```

type AddressBooks = {
  speedDials: [{
    desc: 'Jane Doe',
    dn: '9997770094'
  },
  {
    desc: 'John Doe',
    dn: '9997770095'
  },
]
};
type Address = {
  desc: string;
  dn: string;
  phoneBookName ? : string;
};

```

イベント

addEventListener

dn、status、subStatus のいずれかのフィールドが変更された場合にログに記録される、updated という名前のイベントをリッスンします。

詳細については、[addEventListener\(event, handler\)](#) を参照してください。

例

```
Desktop.agentStateInfo.addEventListener("updated", updatedList =>
  console.log(updatedList) [{
    "name": "dn",

```

```
        "value": "+12580258011",
        "oldValue": ""
    }, {
        "name": "status",
        "value": "LoggedIn",
        "oldValue": "DefaultState"
    }, {
        "name": "subStatus",
        "value": "Available",
        "oldValue": ""
    }
    ] *
    /
)
```

