



SDKWeb-Client-Entwicklerhandbuch

Amazon DCV



Amazon DCV: SDKWeb-Client-Entwicklerhandbuch

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Was ist Amazon DCV Web ClientSDK?	1
Voraussetzungen	2
Unterstützte Features	2
Browser-Unterstützung	2
Konvention zur Versionierung	3
Erste Schritte	4
Stellen Sie eine Connect zu einem DCV Amazon-Server her und holen Sie sich den ersten Frame	5
Schritt 1: Bereite deine Seite vor HTML	6
Schritt 2: Authentifizieren Sie sich, stellen Sie eine Verbindung her und rufen Sie den ersten Frame ab	6
Bonus: Automatisch ein HTML Anmeldeformular erstellen	9
Arbeiten Sie mit DCV Amazon-Funktionen	10
Die featuresUpdate Rückruffunktion verstehen	11
Umgang mit Feature-Updates	11
Amazon DCV Web UI verwenden SDK	12
Voraussetzungen	12
Schritt 1: Bereite deine Seite vor HTML	13
Schritt 2: Authentifizieren, verbinden und rendern Sie die DCVViewer React- Komponente.	14
Aktualisierung von AWS -UI auf Cloudscape Design System	18
SDKReferenz	19
DCVModul	19
Methoden	19
Mitglieder	22
Definitionen für Typ und Rückruf	27
Verbindungsklasse	69
Methoden	19
Authentifizierungsklasse	99
Methoden	19
Ressourcenklasse	100
Methoden	19
DCVAmazon-Webbenutzeroberfläche SDK	101
Komponenten	102

Versionshinweise und Dokumentverlauf	110
Versionshinweise	110
1.8.4 — 1. Oktober 2024	111
1.5.10 — 19. Dezember 2023	112
1.5.6 — 9. November 2023	112
1.4.4 — 29. Juni 2023	112
1.4.0 — 28. März 2023	113
1.3.1 — 9. Dezember 2022	115
1.3.0 — 11. November 2022	115
1.2.1 — 21. Juli 2022	116
1.2.0 — 29. Juni 2022	117
1.1.3 — 23. Mai 2022	117
1.1.2 — 19. Mai 2022	118
1.1.1 — 23. März 2022	118
1.1.0 — 23. Februar 2022	119
1.0.4 — 20. Dezember 2021	120
1.0.3 — 01. September 2021	120
1.0.2 — 30. Juli 2021	121
1.0.1 — 31. Mai 2021	121
1.0.0 — 24. März 2021	122
Dokumentverlauf	122
.....	cxxiv

Was ist der Amazon DCV Web ClientSDK?

Note

Amazon DCV war früher bekannt als NICE DCV.

Amazon DCV ist ein leistungsstarkes Remote-Display-Protokoll. Mit dieser Lösung können Sie Remote-Desktops und Anwendungs-Streamen aus jeder Cloud oder jedem Rechenzentrum sicher auf jedem Gerät in unterschiedlichen Netzwerkbedingungen bereitstellen. Wenn Sie Amazon DCV mit Amazon verwenden EC2, können Sie grafikintensive Anwendungen remote auf EC2 Amazon-Instances ausführen. Anschließend können Sie die Ergebnisse zu weniger leistungsfähigen Clientcomputern streamen. So wird die Notwendigkeit teurer dedizierter Workstations beseitigt.

Der Amazon DCV Web Client SDK ist eine JavaScript Bibliothek, mit der Sie Ihre eigenen DCV Amazon-Webbrowser-Client-Anwendungen entwickeln können. Ihre Endbenutzer können diese Anwendungen verwenden, um eine Verbindung zu einer laufenden DCV Amazon-Sitzung herzustellen und mit ihr zu interagieren.

Mit dem Amazon DCV Web Client SDK als Baustein können Sie maßgeschneiderte Webanwendungen erstellen, die Benutzern von überall aus sofortigen Zugriff auf ihren Desktop oder ihre Anwendungen bieten, und zwar mit einer reaktionsschnellen und flüssigen Leistung, die kaum von einer nativ installierten Anwendung zu unterscheiden ist.

In diesem Handbuch wird erklärt, wie Sie den Amazon DCV Web Client verwenden SDK, um Ihre benutzerdefinierten Webbrowser-Client-Anwendungen für die Interaktion mit DCV Amazon-Sitzungen innerhalb Ihrer Workflows zu erstellen.

Themen

- [Voraussetzungen](#)
- [Unterstützte Features](#)
- [Browser-Unterstützung](#)
- [Konvention zur Versionierung](#)

Voraussetzungen

Bevor Sie mit der Arbeit mit dem Amazon DCV Web Client beginnen SDK, stellen Sie sicher, dass Sie mit Amazon DCV - und DCV Amazon-Sitzungen vertraut sind. Weitere Informationen finden Sie im [DCV Amazon-Administratorhandbuch](#).

Der Amazon DCV Web Client SDK unterstützt Amazon DCV Server Version 2020 und höher.

Unterstützte Features

Sie können benutzerdefinierte Webbrowser-Client-Anwendungen erstellen, die die folgenden DCV Amazon-Funktionen unterstützen:

- Connect zu Windows DCV Amazon-Servern her
- Connect zu Linux DCV Amazon-Servern her
- Verwalten von Streaming-Modi
- Übertragen von Dateien
- Drucken aus Sitzungen
- Kopieren und Einfügen
- Stereo 2.0-Audiowiedergabe
- Stereo-2.0-Audioaufnahme (auf Windows-Servern)
- Touchscreen
- Stylus (auf Linux-, Windows 10- und Windows Server 2019-Servern)
- Unterstützung für mehrere Monitore

Weitere Informationen zu diesen Funktionen finden Sie unter [Unterstützte Funktionen](#) im DCV Amazon-Benutzerhandbuch.

Browser-Unterstützung

Der Amazon DCV Web Client SDK unterstützt JavaScript (ES6) und kann von unseren TypeScript Anwendungen aus JavaScript verwendet werden.

Der Amazon DCV Web Client SDK unterstützt die folgenden Webbrowser:

Browser	Version
Google Chrome	Die letzten drei Hauptversionen
Mozilla Firefox	Die letzten drei Hauptversionen
Microsoft Edge	Die letzten drei Hauptversionen
Apple Safari für macOS	Die letzten drei Hauptversionen

Konvention zur Versionierung

Die Amazon DCV Web SDK Client-Version ist im folgenden Format definiert: *major.minor.patch*. Die Versionierungskonvention entspricht im Allgemeinen dem [semantischen](#) Versionierungsmodell. Eine Änderung in der Hauptversion, z. B. von 1.x.x bis 2.x.x, weist darauf hin, dass grundlegende Änderungen eingeführt wurden, die möglicherweise Codeänderungen und eine geplante Bereitstellung erfordern. Eine Änderung in der Nebenversion, z. B. von 1.1.x bis 1.2.x, ist abwärtskompatibel, kann aber veraltete Elemente enthalten.

Erste Schritte mit dem Amazon DCV Web Client SDK

Der Amazon DCV Web Client SDK besteht aus einer `dcv.js` Hauptdatei und einigen Hilfskomponenten. Alle Dateien werden in einem komprimierten Archiv verteilt, das von der [DCVAmazon-Website](#) heruntergeladen werden kann.

Um mit dem Amazon DCV Web Client zu beginnen SDK

1. Das Amazon DCV Web SDK Client-Archiv ist mit einer sicheren GPG Signatur digital signiert. Um die Signatur des Archivs zu überprüfen, müssen Sie den NICE GPG Schlüssel importieren. Öffnen Sie dazu ein Terminalfenster und importieren Sie den NICE GPG Schlüssel.

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

```
$ gpg --import NICE-GPG-KEY
```

2. Laden Sie das Amazon DCV Web SDK Client-Archiv und die Amazon DCV Web SDK Client-Archivsignatur von der [DCVAmazon-Website](#) herunter.
3. Überprüfen Sie die Signatur des Amazon DCV Web SDK Client-Archivs anhand der Signatur.

```
$ gpg --verify  
signature_filename.zip.sign  
archive_filename.zip
```

Beispielsweise:

```
$ gpg --verify nice-dcv-web-client-sdk-1.8.4-840.zip.sign nice-dcv-web-client-  
sdk-1.8.4-840.zip
```

4. Wenn die Signatur erfolgreich verifiziert wurde, extrahieren Sie den Inhalt des Amazon DCV Web SDK Client-Archivs und platzieren Sie das entpackte Verzeichnis auf Ihrem Webserver. Beispielsweise:

```
$ unzip  
archive_filename.zip
```



```
-d /  
  path_to  
  /  
  server_directory  
  /
```

Important

- Sie müssen die Ordnerstruktur beibehalten, wenn Sie den Amazon DCV Web Client SDK auf Ihrem Webserver bereitstellen.
- Wenn Sie Amazon DCV Web UI verwenden SDK, beachten Sie bitte, dass die DCVViewer React-Komponente erwartet, dass die EULA third-party-licenses .txt- und .txt-Dateien aus diesem Paket im URL Pfad für den eingebetteten Webserver vorhanden sind. Die third-party-licenses .txt-Datei sollte so geändert werden, dass sie auch den Inhalt der entsprechenden Datei aus dem Amazon DCV Web SDK Client-Paket und möglicherweise alle anderen Lizenzinformationen aus den Bibliotheken enthält, die von der nutzenden Benutzeranwendung verwendet werden.

Stellen Sie eine Connect zu einem DCV Amazon-Server her und holen Sie sich den ersten Frame

Das folgende Tutorial zeigt Ihnen, wie Sie Ihre HTML Seite für Ihren benutzerdefinierten Webclient vorbereiten, wie Sie sich authentifizieren und eine Verbindung zu einem DCV Amazon-Server herstellen und wie Sie den ersten Frame mit gestreamten Inhalten aus der DCV Amazon-Sitzung empfangen.

Themen

- [Schritt 1: Bereite deine Seite vor HTML](#)
- [Schritt 2: Authentifizieren Sie sich, stellen Sie eine Verbindung her und rufen Sie den ersten Frame ab](#)
- [Bonus: Automatisch ein HTML Anmeldeformular erstellen](#)

Schritt 1: Bereite deine Seite vor HTML

Auf Ihrer Webseite müssen Sie die benötigten JavaScript Module laden und ein `<div>` HTML Element mit einem gültigen Wert hinzufügen, `id` an dem der Amazon DCV Web Client SDK den Inhaltsstream vom DCV Remote-Amazon-Server Amazon soll.

Beispielsweise:

```
<!DOCTYPE html>
<html lang="en" style="height: 100%;">
  <head>
    <title>DCV first connection</title>
  </head>
  <body style="height: 100%;">
    <div id="root" style="height: 100%;"></div>
    <div id="dcv-display"></div>
    <script type="module" src="index.js"></script>
  </body>
</html>
```

Schritt 2: Authentifizieren Sie sich, stellen Sie eine Verbindung her und rufen Sie den ersten Frame ab

In diesem Abschnitt wird gezeigt, wie Sie den Benutzerauthentifizierungsprozess abschließen, wie Sie eine Verbindung zum DCV Amazon-Server herstellen und wie Sie den ersten Inhaltsframe vom DCV Amazon-Server empfangen.

Importieren Sie zunächst aus der `index.js` Datei den Amazon DCV Web Client SDK. Es kann entweder als Universal Module Definition (UMD) -Modul importiert werden, wie folgt:

```
import "./dcvjs/dcv.js"
```

Andernfalls kann es `1.1.0`, ausgehend von der Version, auch als ECMAScript Module (ESM) aus dem entsprechenden Paket importiert werden, etwa so:

```
import dcv from "./dcvjs/dcv.js"
```

Definieren Sie die Variablen, die zum Speichern des Authentication-Objekts, des Connection-Objekts und des DCV Amazon-Servers verwendet URL werden sollen.

```
let auth,  
    connection,  
    serverUrl;
```

Beim Laden des Skripts protokollieren Sie die Amazon DCV Web SDK Client-Version und rufen Sie beim Laden der Seite die main Funktion auf.

```
console.log("Using Amazon DCV Web Client SDK version " + dcv.version.versionStr);  
document.addEventListener('DOMContentLoaded', main);
```

Die main Funktion legt die Protokollebene fest und startet den Authentifizierungsprozess.

```
function main () {  
    console.log("Setting log level to INFO");  
    dcv.setLogLevel(dcv.LogLevel.INFO);  
  
    serverUrl = "https://your-dcv-server-url:port/";  
  
    console.log("Starting authentication with", serverUrl);  
  
    auth = dcv.authenticate(  
        serverUrl,  
        {  
            promptCredentials: onPromptCredentials,  
            error: onError,  
            success: onSuccess  
        }  
    );  
}
```

Die success Funktionen `promptCredentials`, `error`, und sind obligatorische Callback-Funktionen, die im Authentifizierungsprozess definiert werden müssen.

Wenn der DCV Amazon-Server zur Eingabe von Anmeldeinformationen auffordert, empfängt die `promptCredentials` Callback-Funktion die angeforderte Aufforderung zur Eingabe der Anmeldeinformationen vom Amazon-Server. DCV Wenn der DCV Amazon-Server für die Verwendung der Systemauthentifizierung konfiguriert ist, müssen die Anmeldeinformationen angegeben werden. In den folgenden Codebeispielen wird davon ausgegangen, dass der Benutzername `my_dcv_user` und das Passwort lautet `my_password`.

Wenn die Authentifizierung fehlschlägt, empfängt die `error` Callback-Funktion ein Fehlerobjekt vom DCV Amazon-Server.

Wenn die Authentifizierung erfolgreich ist, empfängt die `success` Callback-Funktion ein Array von Paaren, das die Sitzungs-ID (`sessionId`) und die Autorisierungstoken (`authToken`) für jede Sitzung enthält, zu der der `my_dcv_user` Benutzer auf dem DCV Amazon-Server eine Verbindung herstellen darf. Das folgende Codebeispiel ruft die `Connect`-Funktion auf und stellt eine Verbindung zur ersten im Array zurückgegebenen Sitzung her.

Note

Ersetzen Sie es im folgenden Codebeispiel `MY_DCV_USER` durch Ihren eigenen Benutzernamen und `MY_PASSWORD` durch Ihr eigenes Passwort.

```
function onPromptCredentials(auth, challenge) {
  // Let's check if in challenge we have a username and password request
  if (challengeHasField(challenge, "username") && challengeHasField(challenge,
    "password")) {
    auth.sendCredentials({username: MY_DCV_USER, password: MY_PASSWORD})
  } else {
    // Challenge is requesting something else...
  }
}

function challengeHasField(challenge, field) {
  return challenge.requiredCredentials.some(credential => credential.name === field);
}

function onError(auth, error) {
  console.log("Error during the authentication: " + error.message);
}

// We connect to the first session returned
function onSuccess(auth, result) {
  let {sessionId, authToken} = {...result[0]};

  connect(sessionId, authToken);
}
```

Connect zum DCV Amazon-Server her. Die `firstFrame` Callback-Methode wird aufgerufen, wenn der erste Frame vom DCV Amazon-Server empfangen wird.

```
function connect (sessionId, authToken) {
  console.log(sessionId, authToken);

  dcv.connect({
    url: serverUrl,
    sessionId: sessionId,
    authToken: authToken,
    divId: "dcv-display",
    callbacks: {
      firstFrame: () => console.log("First frame received")
    }
  }).then(function (conn) {
    console.log("Connection established!");
    connection= conn;
  }).catch(function (error) {
    console.log("Connection failed with error " + error.message);
  });
}
```

Bonus: Automatisch ein HTML Anmeldeformular erstellen

Das `challenge` Objekt wird zurückgegeben, wenn die `promptCredentials` Callback-Funktion aufgerufen wird. Es enthält eine Eigenschaft mit dem Namen `requiredCredentials`, bei der es sich um ein Array von Objekten handelt — ein Objekt pro Anmeldeinformationen, die vom DCV Amazon-Server angefordert werden. Jedes Objekt enthält den Namen und den Typ der angeforderten Anmeldeinformationen. Sie können die `requiredCredentials` Objekte `challenge` und verwenden, um automatisch ein HTML Anmeldeformular zu erstellen.

Das folgende Codebeispiel zeigt Ihnen, wie das geht.

```
let form,
    fieldSet;

function submitCredentials (e) {
  var credentials = {};
  fieldSet.childNodes.forEach(input => credentials[input.id] = input.value);
  auth.sendCredentials(credentials);
  e.preventDefault();
}
```

```
function createLoginForm () {
  var submitButton = document.createElement("button");

  submitButton.type = "submit";
  submitButton.textContent = "Login";

  form = document.createElement("form");
  fieldSet = document.createElement("fieldset");

  form.onsubmit = submitCredentials;
  form.appendChild(fieldSet);
  form.appendChild(submitButton);

  document.body.appendChild(form);
}

function addInput (name) {
  var type = name === "password" ? "password" : "text";

  var inputField = document.createElement("input");
  inputField.name = name;
  inputField.id = name;
  inputField.placeholder = name;
  inputField.type = type;
  fieldSet.appendChild(inputField);
}

function onPromptCredentials (_, credentialsChallenge) {
  createLoginForm();
  credentialsChallenge.requiredCredentials.forEach(challenge =>
  addInput(challenge.name));
}
```

Arbeiten Sie mit DCV Amazon-Funktionen

Die Verfügbarkeit der DCV Amazon-Funktionen hängt von den für die DCV Amazon-Sitzung konfigurierten Berechtigungen und den Funktionen des Webbrowsers des Kunden ab.

Die Funktionen, die in einer DCV Amazon-Sitzung verfügbar sind, werden durch die Berechtigungen verwaltet, die für die Sitzung angegeben wurden. Das bedeutet, dass selbst wenn eine Funktion vom Amazon DCV Web Client unterstützt wird, der Zugriff auf diese Funktion aufgrund der

vom Sitzungsadministrator definierten Berechtigungen möglicherweise verhindert wird. Weitere Informationen finden Sie unter [Konfiguration der DCV Amazon-Autorisierung](#) im DCVAmazon-Administratorhandbuch.

Die featuresUpdate Rückruffunktion verstehen

Wenn sich die Verfügbarkeit einer Funktion in einer DCV Amazon-Sitzung ändert, SDK benachrichtigt Sie der Amazon DCV Web Client mithilfe der featuresUpdate Rückruffunktion, die Sie beim Verbindungsaufbau angeben. Beispielsweise:

```
featuresUpdate: function (connection, list) {  
    ...  
},
```

Die Rückruffunktion benachrichtigt Sie nur über die Funktionen, für die sich die Verfügbarkeit geändert hat. Der list Parameter ist ein Array von Zeichenfolgen und enthält nur die Namen der aktualisierten Funktionen. Wenn sich beispielsweise die Verfügbarkeit der Audioeingabefunktion für die Sitzung ändert, umfasst der Parameter nur ["audio-in"]. Wenn sich zu einem späteren Zeitpunkt die Verfügbarkeit der Funktionen zum Kopieren und Einfügen in der Zwischenablage für die Sitzung ändert, umfasst der Parameter nur ["clipboard-copy", "clipboard-paste"] Folgendes:

Umgang mit Feature-Updates

Die featuresUpdate Rückruffunktion benachrichtigt Sie nur darüber, dass sich die Verfügbarkeit einer oder mehrerer Funktionen geändert hat. Um zu erfahren, welche Funktionen aktualisiert wurden, müssen Sie das Feature mithilfe der connection.queryFeature Methode abfragen. Dies kann jederzeit nach Erhalt der Änderungsmitteilung erfolgen. Diese Methode gibt a zurückPromise, das den aktualisierten Status der angeforderten Funktion zurückgibt. Der status Wert ist immer verknüpft und hat eine boolesche (true|false) -Eigenschaft namens. enabled Einige Funktionen haben möglicherweise zusätzliche Eigenschaften im status Wert. Wenn die Verfügbarkeit der Funktion nicht aktualisiert wurde, wird sie abgelehnt.

Der folgende Beispielcode zeigt, wie das geht.

```
// Connection callback called  
function featuresUpdate (_, list) {  
    if (list.length > 0) {  
        list.forEach((feat) => {
```

```
    connection.queryFeature(feat).then(status => console.log(feat, "is",
status.enabled)));
  });
}
```

Amazon DCV Web UI verwenden SDK

Das folgende Tutorial zeigt Ihnen, wie Sie sich beim DCV Amazon-Server authentifizieren, eine Verbindung zu ihm herstellen und die `DCVViewer` React-Komponente über die Amazon DCV Web UI SDK rendern.

Themen

- [Voraussetzungen](#)
- [Schritt 1: Bereite deine Seite vor HTML](#)
- [Schritt 2: Authentifizieren, verbinden und rendern Sie die DCVViewer React-Komponente.](#)
- [Aktualisierung von AWS -UI auf Cloudscape Design System](#)

Voraussetzungen

Sie müssen `React`, `ReactDOM`, `Cloudscape Design Components`, `React`, `Cloudscape Design Global Styles` und `Cloudscape Design Design Tokens` installieren.

```
$ npm i react react-dom @cloudscape-design/components @cloudscape-design/global-styles
@cloudscape-design/design-tokens
```

Sie müssten auch herunterladen `Amazon DCV Web Client SDK`. Lesen [Erste Schritte mit dem Amazon DCV Web Client SDK](#) Sie die step-by-step Anleitung dazu.

Sie müssen einen Alias für den Import des `dcv` Moduls erstellen, da es sich um eine externe Abhängigkeit für `Amazon DCV Web UI` handelt SDK. Wenn Sie beispielsweise `Webpack` verwenden, um Ihre Web-App zu bündeln, können Sie die Option [resolve.alias](#) wie folgt verwenden:

```
const path = require('path');

module.exports = {
  //...
```



```
resolve: {
  alias: {
    dcv: path.resolve('path', 'to', 'dcv.js'),
  },
},
};
```

Wenn Sie Rollup zum Bündeln verwenden, können Sie [@rollup/plugin-alias](#) installieren und es so verwenden:

```
import alias from '@rollup/plugin-alias';
const path = require('path');

module.exports = {
  //...
  plugins: [
    alias({
      entries: [
        { find: 'dcv', replacement: path.resolve('path', 'to', 'dcv.js') },
      ]
    })
  ]
};
```

Schritt 1: Bereite deine Seite vor HTML

Auf Ihrer Webseite müssen Sie die erforderlichen JavaScript Module laden und Sie sollten ein `<div>` HTML Element mit einem gültigen Wert haben, `id` in dem die Eingabekomponente Ihrer App gerendert wird.

Beispielsweise:

```
<!DOCTYPE html>
<html lang="en" style="height: 100%;">
  <head>
    <title>DCV first connection</title>
  </head>
  <body style="height: 100%;">
    <div id="root" style="height: 100%;"></div>
    <script type="module" src="index.js"></script>
  </body>
</html>
```

Schritt 2: Authentifizieren, verbinden und rendern Sie die **DCVViewer** React-Komponente.

In diesem Abschnitt wird gezeigt, wie Sie den Benutzerauthentifizierungsprozess abschließen, wie Sie den DCV Amazon-Server verbinden und wie Sie die DCVViewer React-Komponente rendern.

Zuerst aus der `index.js` Datei, dem Import React ReactDOM und Ihrer App Top-Level-Komponente.

```
import React from "react";
import ReactDOM from 'react-dom';
import App from './App';
```

Rendern Sie den Containerknoten der obersten Ebene Ihrer App.

```
ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById("root")
);
```

Importieren Sie in der `App.js` Datei den Amazon DCV Web Client SDK als ESM Modul, die DCVViewer React-Komponente aus der Amazon DCV Web UI SDK React und das Cloudscape Design Global Styles Paket.

```
import React from "react";
import dcv from "dcv";
import "@cloudscape-design/global-styles/index.css";
import {DCVViewer} from "./dcv-ui/dcv-ui.js";
```

Im Folgenden finden Sie ein Beispiel, das zeigt, wie Sie sich beim Amazon DCV Server authentifizieren und die DCVViewer React-Komponente über die Amazon DCV Web UI rendernSDK, sofern die Authentifizierung erfolgreich war.

```
const LOG_LEVEL = dcv.LogLevel.INFO;
const SERVER_URL = "https://your-dcv-server-url:port/";
const BASE_URL = "/static/js/dcvjs";

let auth;
```

```
function App() {
  const [authenticated, setAuthenticated] = React.useState(false);
  const [sessionId, setSessionId] = React.useState('');
  const [authToken, setAuthToken] = React.useState('');
  const [credentials, setCredentials] = React.useState({});

  const onSuccess = (_, result) => {
    var { sessionId, authToken } = { ...result[0] };

    console.log("Authentication successful.");

    setSessionId(sessionId);
    setAuthToken(authToken);
    setAuthenticated(true);
    setCredentials({});
  }

  const onPromptCredentials = (_, credentialsChallenge) => {
    let requestedCredentials = {};

    credentialsChallenge.requiredCredentials.forEach(challenge =>
requestedCredentials[challenge.name] = "");
    setCredentials(requestedCredentials);
  }

  const authenticate = () => {
    dcv.setLogLevel(LOG_LEVEL);

    auth = dcv.authenticate(
      SERVER_URL,
      {
        promptCredentials: onPromptCredentials,
        error: onError,
        success: onSuccess
      }
    );
  }

  const updateCredentials = (e) => {
    const { name, value } = e.target;
    setCredentials({
      ...credentials,
      [name]: value
    });
  }
}
```

```
});
}

const submitCredentials = (e) => {
  auth.sendCredentials(credentials);
  e.preventDefault();
}

React.useEffect(() => {
  if (!authenticated) {
    authenticate();
  }
}, [authenticated]);

const handleDisconnect = (reason) => {
  console.log("Disconnected: " + reason.message + " (code: " + reason.code + ")");
  auth.retry();
  setAuthenticated(false);
}

return (
  authenticated ?
  <DCVViewer
    dcv={{
      sessionId: sessionId,
      authToken: authToken,
      serverUrl: SERVER_URL,
      baseUrl: BASE_URL,
      onDisconnect: handleDisconnect,
      logLevel: LOG_LEVEL
    }}
    uiConfig={{
      toolbar: {
        visible: true,
        fullscreenButton: true,
        multimonitorButton: true,
      },
    }}
  />
  :
  <div
    style={{
      height: window.innerHeight,
      backgroundColor: "#373737",
```

```

        display: 'flex',
        alignItems: 'center',
        justifyContent: 'center',
    }}
  >
  <form>
    <fieldset>
      {Object.keys(credentials).map((cred) => (
        <input
          key={cred}
          name={cred}
          placeholder={cred}
          type={cred === "password" ? "password" : "text"}
          onChange={updateCredentials}
          value={credentials[cred]}
        />
      ))}
    </fieldset>
    <button
      type="submit"
      onClick={submitCredentials}
    >
      Login
    </button>
  </form>
</div>
);
}

const onError = (_, error) => {
  console.log("Error during the authentication: " + error.message);
}

export default App;

```

Die `success` Funktionen `promptCredentialError`, und sind obligatorische Callback-Funktionen, die im Authentifizierungsprozess definiert werden müssen.

Wenn der DCV Amazon-Server zur Eingabe von Anmeldeinformationen auffordert, empfängt die `promptCredentials` Callback-Funktion die angeforderte Aufforderung zur Eingabe der Anmeldeinformationen vom Amazon-Server. DCV Wenn der DCV Amazon-Server für die Verwendung der Systemauthentifizierung konfiguriert ist, müssen die Anmeldeinformationen in Form eines Benutzernamens und eines Kennworts bereitgestellt werden.

Wenn die Authentifizierung fehlschlägt, empfängt die `error` Callback-Funktion ein Fehlerobjekt vom DCV Amazon-Server.

Wenn die Authentifizierung erfolgreich ist, empfängt die `success` Callback-Funktion ein Array von Paaren, das die Sitzungs-ID (`sessionId`) und die Autorisierungstoken (`authToken`) für jede Sitzung enthält, zu der der Benutzer auf dem DCV Amazon-Server eine Verbindung herstellen darf. Das obige Codebeispiel aktualisiert den React-Status, sodass die `DCVViewer` Komponente bei erfolgreicher Authentifizierung gerendert wird.

Weitere Informationen zu den von dieser Komponente akzeptierten Eigenschaften finden Sie in der [Amazon DCV Web SDK UI-Referenz](#).

Weitere Informationen zu selbstsignierten Zertifikaten finden Sie in den [Erläuterungen zur Umleitung bei selbstsignierten Zertifikaten](#).

Aktualisierung von AWS -UI auf Cloudscape Design System

[Ab SDK Version 1.3.0 haben wir unsere `DCVViewer` Komponente von AWS -UI auf ihre Weiterentwicklung aktualisiert: Cloudscape Design.](#)

Cloudscape verwendet ein anderes visuelles Thema als AWS -UI, aber die zugrunde liegende Codebasis bleibt dieselbe. Daher sollte die Migration Ihrer Anwendung auf der Basis von einfach `DCVViewer` sein. Um zu migrieren, ersetzen Sie die AWS -UI-bezogenen NPM Pakete, die Sie installiert haben, durch die zugehörigen Cloudscape-Pakete:

AWS-UI-Paketname	Name des Cloudscape-Pakets
<code>@awsui /components-react</code>	<code>@cloudscape -design/Komponenten</code>
<code>@awsui /global-styles</code>	<code>@cloudscape -design/globale-styles</code>
<code>@awsui /collection-hooks</code>	<code>@cloudscape -design/collection-hooks</code>
<code>@awsui /design-token</code>	<code>@cloudscape -design/design-token</code>

[Weitere Informationen zur Migration finden Sie auf der -UI-Dokumentationsseite. AWS GitHub](#)

SDKReferenz

Dieser Abschnitt enthält Beschreibungen, Syntax und Anwendungsbeispiele für den Amazon DCV Web ClientSDK.

Themen

- [DCVModul](#)
- [Verbindungsklasse](#)
- [Authentifizierungsklasse](#)
- [Ressourcenklasse](#)
- [DCVAmazon-Webbenutzeroberfläche SDK](#)

DCVModul

Ein Modul, das die Client-Seite des DCV Protokolls implementiert.

Enthüllt

- [Methoden](#)
- [Mitglieder](#)
- [Definitionen für Typ und Rückruf](#)

Methoden

Auflisten

- [authentifizieren \(URL, Rückrufe\) → {Authentifizierung}](#)
- [verbinden \(config\) → {Versprechen. < Verbindung >|Promise. < {Code: ConnectionErrorCode, Nachricht: Zeichenfolge} >}](#)
- [setLogHandler\(Handler\) → {void}](#)
- [setLogLevel\(Ebene\) → {void}](#)

[authentifizieren \(URL, Rückrufe\) → {Authentifizierung}](#)

Startet den Authentifizierungsprozess für den angegebenen DCV Amazon-Serverendpunkt.

Parameter:

Name	Typ	Beschreibung
url	Zeichenfolge	Der Hostname und der Port des laufenden DCV Amazon-Servers im folgenden Format: <code>https://dcv_host_address:port</code> . Beispiel: <code>https://my-dcv-server:8443</code> .
callbacks	authenticationCallbacks	Die Callbacks, die während des Authentifizierungsprozesses aufgerufen werden können.

Rückgabe:

- Das Authentifizierungsobjekt.

Typ

[Authentifizierung](#)

verbinden (config) → {Versprechen. < [Verbindung](#) >|Promise. < {Code: [ConnectionErrorCode](#), Nachricht: Zeichenfolge} >}

Stellt eine Verbindung zum angegebenen DCV Amazon-Serverendpunkt her. Wenn die Verbindung erfolgreich ist, wird ein Connection-Objekt zurückgegeben. Wenn die Verbindung fehlschlägt, wird ein Fehlerobjekt zurückgegeben.

Parameter:

Name	Typ	Beschreibung
config	ConnectionConfig	Das ConnectionConfig Objekt.

Rückgabe:

- Ein Connection-Objekt oder ein Fehlerobjekt.

Typ

Verspricht es. < [Verbindung](#) > | Versprechen. < {Code: [ConnectionErrorCode](#), Nachricht: Zeichenfolge} >

setLogHandler(Handler) → {void}

Legt eine benutzerdefinierte Log-Handler-Funktion fest. Beim Überschreiben des Standard-Log-Handlers geht beim Debuggen mit der Browserkonsole die ursprüngliche Position des Logeintrags verloren.

Parameter:

Name	Typ	Beschreibung
handler	function	Die benutzerdefinierte Log-Handler-Funktion. Die Handlerfunktion enthält Level (Zahl), levelName (Zeichenfolge), Domain (Zeichenfolge) und Nachricht (Zeichenfolge).

Rückgabe:

Typ

void

setLogLevel(Ebene) → {void}

Legt die Protokollebene fest. Dies ist nur erforderlich, wenn der Standard-Log-Handler verwendet wird.

Parameter:

Name	Typ	Beschreibung
level	LogLevel	Die zu verwendende Protokollstufe.

Rückgabe:

Typ

void

Mitglieder

Auflisten

- [\(Konstante\) AudioError: AudioErrorCode](#)
- [\(konstant\) AuthenticationError: AuthenticationErrorCode](#)
- [\(konstant\) ChannelError: ChannelErrorCode](#)
- [\(konstant\) ClosingReasonError: ClosingReasonErrorCode](#)
- [\(konstant\) ConnectionError: ConnectionErrorCode](#)
- [\(konstant\) CustomChannelError: CustomChannelErrorCode](#)
- [\(konstant\) DisplayConfigError: DisplayConfigErrorCode](#)
- [\(konstant\) FileStorageError: FileStorageErrorCode](#)
- [\(konstant\) LogLevel: LogLevel](#)
- [\(konstant\) MultiMonitorError: MultiMonitorErrorCode](#)
- [\(konstant\) ResolutionError: ResolutionErrorCode](#)
- [\(konstant\) TimezoneRedirectionError: TimezoneRedirectionErrorCode](#)
- [\(konstant\) TimezoneRedirectionSetting: TimezoneRedirectionSettingCode](#)
- [\(konstant\) TimezoneRedirectionStatus: TimezoneRedirectionStatusCode](#)
- [\(konstante\) Version](#)
- [\(konstant\): ScreenshotError ScreenshotErrorCode](#)
- [\(konstant\) WebcamError: WebcamErrorCode](#)

(Konstante) AudioError: [AudioErrorCode](#)

Die AudioError Codes enumerieren.

Type:

- [AudioErrorCode](#)

(konstant) AuthenticationError: [AuthenticationErrorCode](#)

Die AuthenticationError Codes enumerieren.

Type:

- [AuthenticationErrorCode](#)

(konstant) ChannelError: [ChannelErrorCode](#)

Die ChannelError Codes enumerieren.

Type:

- [ChannelErrorCode](#)

(konstant) ClosingReasonError: [ClosingReasonErrorCode](#)

Die ClosingReasonError Codes enumerieren.

Type:

- [ClosingReasonErrorCode](#)

(konstant) ConnectionError: [ConnectionErrorCode](#)

Die ConnectionError Codes enumerieren.

Type:

- [ConnectionErrorCode](#)

(konstant) CustomChannelError: [CustomChannelErrorCode](#)

Die CustomChannelError Codes enumerieren.

Type:

- [CustomChannelErrorCode](#)

(konstant) DisplayConfigError: [DisplayConfigErrorCode](#)

Die DisplayConfigError Codes enumerieren.

Type:

- [DisplayConfigErrorCode](#)

(konstant) FileStorageError: [FileStorageErrorCode](#)

Die FileStorageError Codes enumerieren.

Type:

- [FileStorageErrorCode](#)

(konstant) LogLevel: [LogLevel](#)

Die verfügbaren SDK Protokollebenen.

Type:

- [LogLevel](#)

(konstant) MultiMonitorError: [MultiMonitorErrorCode](#)

Die MultiMonitorError Codes enumerieren.

Type:

- [MultiMonitorErrorCode](#)

(konstant) ResolutionError: [ResolutionErrorCode](#)

Die ResolutionError Codes enumerieren.

Type:

- [ResolutionErrorCode](#)

(konstant) TimezoneRedirectionError: [TimezoneRedirectionErrorCode](#)

Die TimezoneRedirectionError Codes enumerieren.

Type:

- [TimezoneRedirectionErrorCode](#)

(konstant) TimezoneRedirectionSetting: [TimezoneRedirectionSettingCode](#)

Die TimezoneRedirectionSetting Codes enumerieren.

Type:

- [TimezoneRedirectionSettingCode](#)

(konstant) TimezoneRedirectionStatus: [TimezoneRedirectionStatusCode](#)

Die TimezoneRedirectionStatus Codes enumerieren.

Type:

- [TimezoneRedirectionStatusCode](#)

(konstante) Version

Die DCV Amazon-Version mit Major, Minor, Patch, Revision, Extended und versionStr.

Eigenschaften:

Name	Typ	Beschreibung
major	Ganzzahl	Die Hauptversionsnummer.
minor	Ganzzahl	Die Nummer der Nebenversion.
patch	Ganzzahl	Die Versionsnummer des Patches.
revision	Ganzzahl	Die Revisionsnummer.
extended	Zeichenfolge	Die erweiterte Zeichenfolge.
versionStr	Zeichenfolge	Eine Verkettung der Haupt-, Neben-, Patch- und Revisionsnummern im Formular. major.minor.patch+build.revision

(konstant): ScreenshotError [ScreenshotErrorCode](#)

Die ScreenshotError Codes enumerieren.

Type:

- [ScreenshotErrorCode](#)

(konstant) WebcamError: [WebcamErrorCode](#)

Die WebcamError Codes enumerieren.

Type:

- [WebcamErrorCode](#)

Definitionen für Typ und Rückruf

Auflisten

- [AudioErrorCode](#)
- [authenticationCallbacks](#)
- [AuthenticationErrorCode](#)
- [authErrorCallback\(Authentifizierung, Fehler\)](#)
- [authPromptCredentialsRückruf \(Authentifizierung, Herausforderung\)](#)
- [authSuccessCallback\(Authentifizierung,authenticationData\)](#)
- [Kanal](#)
- [ChannelErrorCode](#)
- [clipboardEventCallback\(Ereignis\)](#)
- [ClosingReasonErrorCode](#)
- [Farbraum](#)
- [connectionCallbacks](#)
- [ConnectionConfig](#)
- [ConnectionErrorCode](#)
- [createDirectory\(Pfad\)](#)
- [CustomChannelErrorCode](#)
- [dataChannelCallback\(Information\)](#)
- [deleteFile\(Pfad\)](#)
- [deviceChangeEventRückruf \(\)](#)
- [disconnectCallback\(Grund\)](#)
- [displayAvailabilityCallback\(Status,displayId\)](#)
- [DisplayConfigErrorCode](#)
- [displayLayoutCallback\(serverWidth,serverHeight, Köpfe\)](#)
- [Merkmal](#)
- [featuresUpdateCallback\(featuresList\)](#)
- [fileDownloadCallback\(fileResource\)](#)
- [filePrintedCallback\(printResource\)](#)

- [Dateispeicher](#)
- [filestorageEnabledCallback\(aktiviert\)](#)
- [FileStorageErrorCode](#)
- [firstFrameCallback\(resizeEnabled, relativeMouseMode Aktiviert,displayId\)](#)
- [idleWarningNotificationRückruf \(\) disconnectionDateTime](#)
- [collaboratorListCallback\(Mitarbeiter\)](#)
- [licenseNotificationCallback\(Benachrichtigung\)](#)
- [Liste \(Pfad\)](#)
- [LogLevel](#)
- [Überwachen](#)
- [MultiMonitorErrorCode](#)
- [qualityIndicatorStateRückruf \(Status\)](#)
- [renameDirectory\(src, dest\)](#)
- [renameFile\(src, dest\)](#)
- [ResolutionErrorCode](#)
- [retrieveFile\(Pfad\)](#)
- [screenshotCallback\(Bildschirmfoto\)](#)
- [ScreenshotErrorCode](#)
- [serverInfo](#)
- [stats](#)
- [storeFile\(Datei, Verzeichnis\)](#)
- [TimezoneRedirectionErrorCode](#)
- [TimezoneRedirectionSettingCode](#)
- [TimezoneRedirectionStatusCode](#)
- [WebcamErrorCode](#)

AudioErrorCode

Die im AudioError Modul verfügbaren Code-Aufzählungen DCV

- SETTING_AUDIO_FAILED
- CHANNEL_NOT_AVAILABLE

Type:

- Zahl

authenticationCallbacks

Rückrufe zur Authentifizierung

Type:

- Object

Eigenschaften:

Name	Typ	Beschreibung
promptCredentials	authPromptCredentialsRückruf	Die Callback-Funktion, die aufgerufen wird, wenn der Benutzer zur Eingabe von Anmeldeinformationen aufgefordert wird.
error	authErrorCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn die Authentifizierung fehlschlägt.
success	authSuccessCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn die Authentifizierung erfolgreich ist.

AuthenticationErrorCode

Die im AuthenticationError Modul verfügbaren Code-Aufzählungen DCV

- INVALID_MESSAGE
- UNKNOWN_AUTH_MODE

- SESSION_NOT_AVAILABLE
- NO_SESSIONS
- WRONG_CREDENTIALS
- SASL_CHALLENGE
- SASL_AUTH_MECHANISM
- FAILED_COMMUNICATION
- AUTHENTICATION_REJECTED
- GENERIC_ERROR
- WRONG_CREDENTIALS_FORMAT
- WRONG_CREDENTIALS_TYPE
- UNREQUESTED_CREDENTIALS
- MISSING_CREDENTIAL

Type:

- Zahl

authErrorCallback(Authentifizierung, Fehler)

Die Callback-Funktion, die aufgerufen werden soll, wenn die Authentifizierung fehlschlägt.

Parameter:

Name	Typ	Beschreibung						
authentication	Authentifizierung	Das Authentifizierungsobjekt.						
error	Object	Das Fehlerobjekt, das durch den Authentifizierungsprozess ausgelöst wurde. <table border="1" data-bbox="1068 1583 1528 1885"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>code</td> <td>AuthenticationErrorCode</td> <td>Der Fehlercode.</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	code	AuthenticationErrorCode	Der Fehlercode.
Name	Typ	Beschreibung						
code	AuthenticationErrorCode	Der Fehlercode.						

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
		message	Zeichenfolge	Die Fehlermeldung.

authPromptCredentialsRückruf (Authentifizierung, Herausforderung)

Die Callback-Funktion, die aufgerufen wird, wenn der Benutzer zur Eingabe von Anmeldeinformationen aufgefordert wird. Der Benutzer muss die Aufforderung beantworten, indem er die angeforderten Anmeldeinformationen eingibt.

Parameter:

Name	Typ	Beschreibung		
authentication	Authentifizierung	Das Authentifizierungsobjekt.		
challenge	Object	Die Herausforderung.		
		Name	Typ	Beschreibung
		requires	Reihe. <Object>	Eine Reihe von angeforderten Anmeldeinformationssubjekten.

Name	Typ	Beschreibung															
		<table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td> <table border="1"> <thead> <tr> <th>N</th> <th>T</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>n</td> <td>Z</td> <td>Das Info-Objekt Name der angeforderten Anmeldeinformationen.</td> </tr> <tr> <td>t</td> <td>Z</td> <td>Das Info-Objekt Typ der angeforderten Anmeldeinformationen.</td> </tr> </tbody> </table> </td> </tr> </tbody> </table>	Name	Typ	Beschreibung			<table border="1"> <thead> <tr> <th>N</th> <th>T</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>n</td> <td>Z</td> <td>Das Info-Objekt Name der angeforderten Anmeldeinformationen.</td> </tr> <tr> <td>t</td> <td>Z</td> <td>Das Info-Objekt Typ der angeforderten Anmeldeinformationen.</td> </tr> </tbody> </table>	N	T	Beschreibung	n	Z	Das Info-Objekt Name der angeforderten Anmeldeinformationen.	t	Z	Das Info-Objekt Typ der angeforderten Anmeldeinformationen.
Name	Typ	Beschreibung															
		<table border="1"> <thead> <tr> <th>N</th> <th>T</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>n</td> <td>Z</td> <td>Das Info-Objekt Name der angeforderten Anmeldeinformationen.</td> </tr> <tr> <td>t</td> <td>Z</td> <td>Das Info-Objekt Typ der angeforderten Anmeldeinformationen.</td> </tr> </tbody> </table>	N	T	Beschreibung	n	Z	Das Info-Objekt Name der angeforderten Anmeldeinformationen.	t	Z	Das Info-Objekt Typ der angeforderten Anmeldeinformationen.						
N	T	Beschreibung															
n	Z	Das Info-Objekt Name der angeforderten Anmeldeinformationen.															
t	Z	Das Info-Objekt Typ der angeforderten Anmeldeinformationen.															

authSuccessCallback(Authentifizierung,authenticationData)

Die Callback-Funktion, die aufgerufen werden soll, wenn die Authentifizierung erfolgreich ist.

Parameter:

Name	Typ	Beschreibung
authentication	Authentifizierung	Das Authentifizierungsobjekt.

Name	Typ	Beschreibung									
authenticationData	Array. <Object>	Eine Reihe von Objekten, die DCV Amazon-Sitzungs IDs - und Authentifizierungstoken enthalten.									
		<table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>sessionId</td> <td>Zeichenfolge</td> <td>Die DCV Amazon-Sitzungs-ID.</td> </tr> <tr> <td>authToken</td> <td>Zeichenfolge</td> <td>Das Authentifizierungstoken für die DCV Amazon-Sitzung.</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	sessionId	Zeichenfolge	Die DCV Amazon-Sitzungs-ID.	authToken	Zeichenfolge	Das Authentifizierungstoken für die DCV Amazon-Sitzung.
Name	Typ	Beschreibung									
sessionId	Zeichenfolge	Die DCV Amazon-Sitzungs-ID.									
authToken	Zeichenfolge	Das Authentifizierungstoken für die DCV Amazon-Sitzung.									

Kanal

Die verfügbaren Kanäle, die angegeben werden können.

Type:

- „Zwischenablage“ | „Anzeige“ | „Eingabe“ | „Audio“ | „Dateispeicher“

ChannelErrorCode

Die im Modul verfügbaren ChannelError Code-Aufzählungen DCV

- ALREADY_OPEN
- INITIALIZATION_FAILED
- REJECTED

Type:

- Zahl

clipboardEventCallback(Ereignis)

Die Callback-Funktion, die aufgerufen werden soll, wenn a generiert clipboardEvent wird.

Parameter:

Name	Typ	Beschreibung			
event	Object	Informationen über das Zwischenablage-Ereignis.			
		Name	Typ	Attribu	Beschreib
					ung
		name	einger tet kopier einfüg dataSi lert autoC one 		Immer präsent. Der Name des Ereigniss es.

Name	Typ	Beschreibung			
		Name	Typ	Attribu	Beschreib ung
			newD: ailable autoP: Done remote or paste/ lableD		
		clip: Data	Objekt Zeiche lge		Die Daten in der Zwischena blage.
		autoC	booles	<optio >	Gibt an, ob das automatis che Kopieren aus der Zwischena blage der Sitzung in die

Name	Typ	Beschreibung			
		Name	Typ	Attribu	Beschreib ung
					Zwischena blage des lokalen Clients aktiviert ist.
		maxDä ze	Zahl	<optio >	Die maximale Datenmeng e, die in der Zwischena blage platziert werden kann.
		error	Zeiche lge	<optio >	Fehlerinf ormatione n, falls zutreffen d.

ClosingReasonErrorCode

Die im ClosingReasonError Modul verfügbaren Code-Aufzählungen DCV

- TRANSPORT_ERROR
- NO_ERROR
- GENERIC_ERROR
- INTERNAL_SERVER_ERROR
- PROTOCOL_ERROR
- AUTHORIZATION_DENIED
- AUTHORIZATION_REVOKED
- ACCESS_REJECTED
- IDLE_TIMEOUT_EXPIRED
- DISCONNECT_BY_OWNER
- DISCONNECT_BY_USER
- EVICTED
- EXTERNAL_PROTOCOL_CONNECTION_EVICTED
- DISCONNECTION_REQUESTED

Type:

- Zahl

Farbraum

Die verfügbaren Farbräume, die angegeben werden können.

Type:

- "|RGB" YUV _ REC6 01" | "_ 09" YUV REC7

connectionCallbacks

Die Callbacks, die im Falle eines Verbindungsfehlers aufgerufen werden können.

Type:

- Object

Eigenschaften:

Name	Typ	Beschreibung
<code>disconnect</code>	disconnectCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn die Verbindung beendet wird.
<code>displayLayout</code>	displayLayoutCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn das Display-Layout oder die Auflösung geändert werden.
<code>displayAvailability</code>	displayAvailabilityCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn sich die Verfügbarkeit eines Displays ändert.
<code>firstFrame</code>	firstFrameCallback	Die Callback-Funktion, die aufgerufen wird, wenn der erste Frame vom DCV Amazon-Server empfangen wird.
<code>filePrinted</code>	filePrintedCallback	Die Callback-Funktion, die aufgerufen wird, wenn eine Datei auf dem DCV Amazon-Server gedruckt wird.
<code>fileDownload</code>	fileDownloadCallback	Die Callback-Funktion, die aufgerufen wird, wenn eine Datei bereit ist, vom DCV Amazon-Server heruntergeladen zu werden.
<code>dataChannel</code>	dataChannelCallback	Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server

Name	Typ	Beschreibung
		eine Benachrichtigung über die Verfügbarkeit eines Datenkanals sendet.
licenseNotification	licenseNotificationCallback	Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server eine Benachrichtigung über den Lizenzstatus sendet.
idleWarningNotification	idleWarningNotificationRückruf	Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server eine Timeout-Warnung im Leerlauf sendet.
collaboratorList	collaboratorListCallback	Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server die Liste der Mitarbeiter sendet (seit Amazon DCV Web Client SDK Version 1.1.0).
qualityIndicatorState	qualityIndicatorStateRückruf	Die Callback-Funktion, die aufgerufen werden soll, wenn sich der Status der Verbindungsqualitätsanzeige ändert.
filestorageEnabled	filestorageEnabledCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn die Dateispeicherung aktiviert oder deaktiviert ist.

Name	Typ	Beschreibung
featuresUpdate	featuresUpdateCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn sich der Status einer Funktion ändert.
clipboardEvent	clipboardEventCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn eine generiert clipboard Event wird.
deviceChangeEvent	deviceChangeEventRückruf	Die Callback-Funktion, die aufgerufen wird, wenn ein deviceChange Ereignis ausgelöst wird.
screenshot	screenshotCallback	Die Callback-Funktion, die aufgerufen werden soll, wenn a verfügbar screenshot ist.

ConnectionConfig

Die Konfiguration für eine DCV Amazon-Verbindung.

Type:

- Object

Eigenschaften:

Name	Typ	Beschreibung
url	Zeichenfolge	Der Hostname und der Port des laufenden DCV Amazon-Servers im folgenden Format:https://dcv_host_address:po

Name	Typ	Beschreibung
		rt . Beispiel: https://my-dcv-server:8443 .
sessionId	Zeichenfolge	Die DCV Amazon-Sitzungs-ID.
authToken	Zeichenfolge	Das Authentifizierungstoken, das für die Verbindung mit dem Server verwendet werden soll.
baseUrl	Zeichenfolge	Das Absolute oder Relative, URL aus dem SDK Dateien geladen werden sollen.
resourceBaseUrl	Zeichenfolge	Das Absolute oder Relative, URL von dem aus auf DCV Ressourcen zugegriffen werden soll.
enabledChannels	Array. < Kanal >	Zeigt die Liste der Kanäle an, die aktiviert werden können. Wenn nicht angegeben oder ein leeres Array bereitgestellt wird, werden standardmäßig alle verfügbaren Kanäle verwendet.
losslessColorspace	Farbraum	Gibt den Farbraum an, der verwendet wird. Falls nicht angegeben, wird standardmäßig "" RGB verwendet.
divId	Zeichenfolge	Die ID des div Objekts in dem Bereich, in HTML DOM dem die Zeichenfläche mit dem Remote-Stream erstellt werden SDK soll.

Name	Typ	Beschreibung
<code>volumeLevel</code>	Ganzzahl	Die bevorzugte Lautstärke. Der gültige Bereich liegt zwischen 0 und 100.
<code>clipboardAutoSync</code>	boolesch	Gibt an, ob das automatische Kopieren von der Zwischenablage der DCV Amazon-Sitzung in die Zwischenablage des lokalen Clients für kompatible Webbrowser aktiviert ist.
<code>dynamicAudioTuning</code>	boolesch	Gibt an, ob das Audio basierend auf den Audioeinstellungen des DCV Amazon-Servers dynamisch eingestellt werden soll, wenn eine Verbindung hergestellt wird.
<code>clientHiDpiScaling</code>	boolesch	Gibt an, ob die Leinwand auf der Grundlage der des Clients skaliert werden soll DPI.
<code>highColorAccuracy</code>	boolesch	Gibt an, ob eine hohe Farbgenauigkeit verwendet werden sollte, sofern verfügbar. Wenn nicht angegeben, ist die Standardinstellung <code>false</code> .
<code>enableWebCodecs</code>	Boolesch	Gibt an, ob verwendet werden WebCodecs soll, falls verfügbar. Es gilt der Standardwert <code>false</code> , wenn keine Angabe gemacht wird.

Name	Typ	Beschreibung
observers	connectionCallbacks	Die Callback-Funktionen zum Aufrufen von Ereignissen, die sich auf die Verbindung beziehen.
callbacks	connectionCallbacks	Entspricht der observers Eigenschaft, aber jeder Callback enthält das Connection-Objekt als ersten Parameter.

ConnectionErrorCode

Die im ConnectionError Modul verfügbaren Code-Aufzählungen DCV

- ALREADY_OPEN
- INVALID_CONFIG
- INITIALIZATION_FAILED
- REJECTED
- MAIN_CHANNEL_ALREADY_OPEN
- GENERIC_ERROR(seit DCV Server 2021.0)
- INTERNAL_SERVER_ERROR(seit DCV Server 2021.0)
- AUTHENTICATION_FAILED(seit DCV Server 2021.0)
- PROTOCOL_ERROR(seit DCV Server 2021.0)
- INVALID_SESSION_ID(seit DCV Server 2021.0)
- INVALID_CONNECTION_ID(seit DCV Server 2021.0)
- CONNECTION_LIMIT_REACHED(seit DCV Server 2021.0)
- SERVER_UNREACHABLE(seit DCV Server 2022.1)
- GATEWAY_BUSY
- UNSUPPORTED_CREDENTIAL(seit DCV Server 2022.2)
- TRANSPORT_ERROR

Type:

- Zahl

createDirectory(Pfad)

Parameter:

Name	Typ	Beschreibung
path	Zeichenfolge	Der absolute Pfad auf dem Server, auf dem wir ein Verzeichnis erstellen möchten. Er sollte auch den Namen des Zielverzeichnisses enthalten.

CustomChannelErrorCode

Die im CustomChannelError Modul verfügbaren Code-Aufzählungen DCV

- TRANSPORT_ERROR

Type:

- Zahl

dataChannelCallback(Information)

Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server eine Benachrichtigung über die Verfügbarkeit eines Datenkanals sendet.

Parameter:

Name	Typ	Beschreibung
info	Object	Informationen über den Datenkanal.

Name	Typ	Beschreibung									
		<table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>Zeichenfolge</td> <td>Der Name des Datenkanals.</td> </tr> <tr> <td>token</td> <td>Zeichenfolge</td> <td>Das Authentifizierungstoken für den Datenkanal.</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	name	Zeichenfolge	Der Name des Datenkanals.	token	Zeichenfolge	Das Authentifizierungstoken für den Datenkanal.
Name	Typ	Beschreibung									
name	Zeichenfolge	Der Name des Datenkanals.									
token	Zeichenfolge	Das Authentifizierungstoken für den Datenkanal.									

deleteFile(Pfad)

Parameter:

Name	Typ	Beschreibung
path	Zeichenfolge	Der absolute Pfad auf dem Server, der die Datei identifiziert, die wir löschen möchten.

deviceChangeEventRückruf ()

Die Callback-Funktion, die aufgerufen wird, wenn ein deviceChange Ereignis ausgelöst wird.

disconnectCallback(Grund)

Die Callback-Funktion, die aufgerufen werden soll, wenn die Verbindung beendet wird.

Parameter:

Name	Typ	Beschreibung									
reason	Object	Der Grund für die Unterbrechung der Verbindung.									
		<table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>code</td> <td>Zahl</td> <td>Der Ursachencode.</td> </tr> <tr> <td>message</td> <td>Zeichenfolge</td> <td>Die Ursachenmeldung.</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	code	Zahl	Der Ursachencode.	message	Zeichenfolge	Die Ursachenmeldung.
Name	Typ	Beschreibung									
code	Zahl	Der Ursachencode.									
message	Zeichenfolge	Die Ursachenmeldung.									

`displayAvailabilityCallback(Status,displayId)`

Die Callback-Funktion, die aufgerufen wird, wenn sich die Verfügbarkeit eines Displays ändert.

Parameter:

Name	Typ	Beschreibung						
status	Object	Der Status der Anzeige.						
		<table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>enabled</td> <td>boolesch</td> <td>Zeigt an, ob die Anzeige</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	enabled	boolesch	Zeigt an, ob die Anzeige
Name	Typ	Beschreibung						
enabled	boolesch	Zeigt an, ob die Anzeige						

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				aktiviert ist.
		closed	boolesch	Zeigt an, ob das Display geschlossen ist.
displayId	Zahl	Die Kennung für das Display.		

DisplayConfigErrorCode

Die im DisplayConfigError Modul verfügbaren Code-Aufzählungen DCV

- INVALID_ARGUMENT
- UNSUPPORTED_OPERATION
- NO_CHANNEL

Type:

- Zahl

displayLayoutCallback(serverWidth,serverHeight, Köpfe)

Die Callback-Funktion, die aufgerufen werden soll, wenn das Display-Layout oder die Auflösung geändert werden.

Parameter:

Name	Typ	Beschreibung
serverWidth	Zahl	Die Breite (in Pixeln) der primären Anzeige.
serverHeight	Zahl	Die Höhe (in Pixeln) des primären Displays.
heads	Reihe. < Überwachen >	Die vom DCV Amazon-Server unterstützten Displayköpfe.

Merkmal

Die Merkmalswerte.

- `display`- Zeigt die Verfügbarkeit eines Videostreams mit einem einzigen Display an.
- `display-multi`- Zeigt die Verfügbarkeit eines Videostreams mit mehreren Displays an.
- `high-color-accuracy`- Zeigt die Verfügbarkeit einer hohen Farbgenauigkeit an (seit Amazon DCV Web Client SDK Version 1.1.0).
- `mouse`- Zeigt die Verfügbarkeit von Mausfunktionen an.
- `keyboard`- Zeigt die Verfügbarkeit von Tastaturfunktionen an.
- `keyboard-sas`- Zeigt die Verfügbarkeit der SAS Sequenzfunktion (Strg + Alt + Löschen) an.
- `relative-mouse`- Zeigt die Verfügbarkeit des relativen Mausmodus an.
- `clipboard-copy`- Zeigt die Verfügbarkeit der Funktion zum Kopieren der Zwischenablage vom DCV Amazon-Server zum Client an.
- `clipboard-paste`- Zeigt an, ob die Funktion zum Einfügen in die Zwischenablage vom Client auf den DCV Amazon-Server verfügbar ist.
- `audio-in`- Zeigt die Verfügbarkeit der Audioeingabefunktion über das Mikrofon an.
- `audio-out`- Zeigt die Verfügbarkeit der Audiowiedergabefunktion an.
- `webcam`- Zeigt die Verfügbarkeit der Webcam-Streaming-Funktionalität an.
- `file-download`- Zeigt die Verfügbarkeit der Funktion zum Herunterladen von Dateien vom DCV Amazon-Server zum Client an.
- `file-upload`- Zeigt die Verfügbarkeit der Datei-Upload-Funktionalität vom Client auf den DCV Amazon-Server an.

- `timezone-redirection`— Zeigt die Verfügbarkeit der Funktion zur Zeitzonenumleitung an (seit Amazon DCV Web Client SDK Version 1.3.0).

Type:

- Zeichenfolge

featuresUpdateCallback(featuresList)

Die Callback-Funktion, die aufgerufen wird, wenn sich der Status einer Funktion ändert.

Parameter:

Name	Typ	Beschreibung
featuresList	Reihe. < Funktion >	Eine Reihe von Funktionen, die sich geändert haben.

fileDownloadCallback(fileResource)

Die Callback-Funktion, die aufgerufen wird, wenn eine Datei bereit ist, vom DCV Amazon-Server heruntergeladen zu werden.

Parameter:

Name	Typ	Beschreibung						
fileResource	Object	Informationen über die Datei, die zum Herunterladen bereit ist. <table border="1" data-bbox="1068 1602 1529 1879"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>id</td> <td>Zeichenfolge</td> <td>Der Bezeichner für</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	id	Zeichenfolge	Der Bezeichner für
Name	Typ	Beschreibung						
id	Zeichenfolge	Der Bezeichner für						

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				die Datei.
		url	Zeichenfolge	Der URL, der zum Herunterladen der Datei verwendet werden soll.
		domain	Zeichenfolge	Die Ressourcendomäne.
		token	Zeichenfolge	Das Authentifizierungstoken, das zum Herunterladen der Datei verwendet werden soll. Das Token

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				ist auch in der enthaltenen URL.

filePrintedCallback(printResource)

Die Callback-Funktion, die aufgerufen wird, wenn eine Datei auf dem DCV Amazon-Server gedruckt wird.

Parameter:

Name	Typ	Beschreibung		
printResource	Object	Informationen über die gedruckte Datei.		
		Name	Typ	Beschreibung
		id	Zeichenfolge	Die Kennung für die gedruckte Datei.
		url	Zeichenfolge	Der URL, der zum Herunterladen

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				der gedruckten Datei verwendet werden soll.
		domain	Zeichensfolge	Die Ressourcendomäne. In diesem Fallprinter.
		token	Zeichensfolge	Das Authentifizierungstoken, das zum Herunterladen der gedruckten Datei verwendet werden soll. Das Token ist auch in der

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				enthalten URL.

Dateispeicher

Objekt, das das Erkunden und Ausführen von Aktionen im Dateisystem ermöglicht.

Type:

- Object

Eigenschaften:

Name	Typ	Beschreibung
list	Liste	Funktion, die es ermöglicht, die Elemente (Dateien und Verzeichnisse) aufzulisten, die sich im angegebenen Pfad auf dem Server befinden.
createDirectory	createDirectory	Funktion, die es ermöglicht, ein Verzeichnis im angegebenen Pfad auf dem Server zu erstellen.
retrieveFile	retrieveFile	Funktion, die es ermöglicht, eine Datei lokal im angegebenen Pfad auf dem Server herunterzuladen.
deleteFile	deleteFile	Funktion, die es ermöglicht, eine Datei im angegebenen

Name	Typ	Beschreibung
		en Pfad auf dem Server zu löschen.
renameFile	renameFile	Funktion, die es ermöglicht, eine Datei vom angegebenen Quellpfad in den angegebenen Zielpfad umzubenennen.
renameDirectory	renameDirectory	Funktion, die es ermöglicht, ein Verzeichnis vom angegebenen Quellpfad in den absoluten Zielpfad umzubenennen.
storeFile	storeFile	Funktion, die es ermöglicht, eine lokale Datei in den angegebenen Pfad auf dem Server hochzuladen.

filestorageEnabledCallback(aktiviert)

Die Callback-Funktion, die aufgerufen werden soll, wenn die Dateispeicherung aktiviert ist. Lazy-Channel nur in Internet Explorer 11.

Parameter:

Name	Typ	Beschreibung
enabled	boolesch	Zeigt an, ob der Dateispeicher aktiviert ist.

FileStorageErrorCode

Die im FileStorageError Modul verfügbaren Code-Aufzählungen DCV

- CANCELLED

- ABORTED
- INVALID_ARGUMENT
- NOT_IMPLEMENTED
- ERROR
- ALREADY_EXIST
- NOT_FOUND

Type:

- Zahl

`firstFrameCallback(resizeEnabled, relativeMouseMode Aktiviert, displayId)`

Die Callback-Funktion, die aufgerufen wird, wenn der erste Frame vom DCV Amazon-Server empfangen wird. Wird für jedes Display ausgegeben.

Parameter:

Name	Typ	Beschreibung
<code>resizeEnabled</code>	boolesch	Gibt an, ob der Server die Größenänderung des Client-Display-Layouts unterstützt.
<code>relativeMouseModeEnabled</code>	boolesch	Gibt an, ob der Server den relativen Mausmodus unterstützt.
<code>displayId</code>	Zahl	Der Bezeichner für das Display.

`idleWarningNotificationRückruf () disconnectionDateTime`

Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server eine Timeout-Warnung im Leerlauf sendet.

Parameter:

Name	Typ	Beschreibung
<code>disconnectionDateTime</code>	Datum	Datum und Uhrzeit der Verbindungsunterbrechung.

`collaboratorListCallback(Mitarbeiter)`

Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server die Liste der Mitarbeiter sendet.

Parameter:

Name	Typ	Beschreibung									
<code>collaborators</code>	Array. <Object>	Eine Liste von Objekten, die Informationen über Mitarbeiter enthält. <table border="1" data-bbox="1068 1129 1529 1871"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td><code>username</code></td> <td>Zeichenfolge</td> <td>Der Benutzername des Mitarbeiters.</td> </tr> <tr> <td><code>owner</code></td> <td>boolesch</td> <td>Gibt an, ob der Mitarbeiter der Sitzungs-</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	<code>username</code>	Zeichenfolge	Der Benutzername des Mitarbeiters.	<code>owner</code>	boolesch	Gibt an, ob der Mitarbeiter der Sitzungs-
Name	Typ	Beschreibung									
<code>username</code>	Zeichenfolge	Der Benutzername des Mitarbeiters.									
<code>owner</code>	boolesch	Gibt an, ob der Mitarbeiter der Sitzungs-									

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				esitzer ist.
		connect nId	Zahl	Gibt die ID an, die der Server der Verbindung zugewiesen hat.

licenseNotificationCallback(Benachrichtigung)

Die Callback-Funktion, die aufgerufen wird, wenn der DCV Amazon-Server eine Benachrichtigung über den Lizenzstatus sendet.

Parameter:

Name	Typ	Beschreibung		
notification	Object	Die Benachrichtigung.		
		Name	Typ	Beschreibung
		product	Zeichenfolge	Das DCV Produkt.

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
		status	Zeichenfolge	Der Status der Lizenz.
		message	Zeichenfolge	Eine Nachricht.
		leftDay	Zahl	Die Anzahl der Tage, bevor die Lizenz abläuft.
		isDemo	boolesch	Gibt an, ob es sich bei der Lizenz um eine Demo-Lizenz handelt.
		numUnused	Zahl	Die Anzahl der nicht

Name	Typ	Beschreibung												
		<table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td>lizenzierten Verbindungen.</td> </tr> <tr> <td>licenseMode</td> <td>Zeichenfolge</td> <td>Der Lizenzierungsmodus.</td> </tr> <tr> <td>documentationUrl</td> <td>Zeichenfolge</td> <td>Der URL für die Dokumentation.</td> </tr> </tbody> </table>	Name	Typ	Beschreibung			lizenzierten Verbindungen.	licenseMode	Zeichenfolge	Der Lizenzierungsmodus.	documentationUrl	Zeichenfolge	Der URL für die Dokumentation.
Name	Typ	Beschreibung												
		lizenzierten Verbindungen.												
licenseMode	Zeichenfolge	Der Lizenzierungsmodus.												
documentationUrl	Zeichenfolge	Der URL für die Dokumentation.												

Liste (Pfad)

Parameter:

Name	Typ	Beschreibung
path	Zeichenfolge	Der absolute Pfad auf dem Server, dessen Inhalt wir auflisten möchten.

LogLevel

Die verfügbaren SDK Protokollebenen.

Type:

- TRACE | DEBUG | INFO | WARN | ERROR | SILENT

Überwachen

Type:

- Object

Eigenschaften:

Name	Typ	Beschreibung												
name	Zeichenfolge	Der Name des Displaykopfs.												
rect	Object	Informationen über den Displaykopf. <table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>x</td> <td>Zahl</td> <td>Die x Anfangskoordinate für den Displaykopf.</td> </tr> <tr> <td>y</td> <td>Zahl</td> <td>Die y Anfangskoordinate für den Displaykopf.</td> </tr> <tr> <td>width</td> <td>Zahl</td> <td>Die Breite (in Pixeln) des</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	x	Zahl	Die x Anfangskoordinate für den Displaykopf.	y	Zahl	Die y Anfangskoordinate für den Displaykopf.	width	Zahl	Die Breite (in Pixeln) des
Name	Typ	Beschreibung												
x	Zahl	Die x Anfangskoordinate für den Displaykopf.												
y	Zahl	Die y Anfangskoordinate für den Displaykopf.												
width	Zahl	Die Breite (in Pixeln) des												

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				Displaykopfs.
		height	Zahl	Die Höhe (in Pixeln) des Displaykopfs.
primary	boolesch	Gibt an, ob der Displaykopf der primäre Displaykopf ist. Dies wird anhand des Remote-Betriebssystems ermittelt, sofern verfügbar.		
dpi	Zahl	Der DPI des Displaykopfs.		

MultiMonitorErrorCode

Die im MultiMonitorError Modul verfügbaren Code-Aufzählungen DCV

- NO_DISPLAY_CHANNEL
- MAX_DISPLAY_NUMBER_REACHED
- INVALID_ARGUMENT
- DISPLAY_NOT_OPENED_BY_SERVER
- REQUEST_TIMEOUT
- GENERIC_ERROR
- NO_ERROR

Type:

- Zahl

qualityIndicatorStateRückruf (Status)

Die Callback-Funktion, die aufgerufen werden soll, wenn sich der Status der Verbindungsqualitätsanzeige ändert.

Parameter:

Name	Typ	Beschreibung												
state	Reihe. <Object>	Informationen über die Verbindungsqualität. <table border="1" data-bbox="1068 892 1526 1810"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>Zeichenfolge</td> <td>Der Name des Indikators.</td> </tr> <tr> <td>status</td> <td>NORMAL WARNIN CRITICA</td> <td>Beschreibung des Status.</td> </tr> <tr> <td>changed</td> <td>boolesch</td> <td>Gibt an, ob sich der Status</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	name	Zeichenfolge	Der Name des Indikators.	status	NORMAL WARNIN CRITICA	Beschreibung des Status.	changed	boolesch	Gibt an, ob sich der Status
Name	Typ	Beschreibung												
name	Zeichenfolge	Der Name des Indikators.												
status	NORMAL WARNIN CRITICA	Beschreibung des Status.												
changed	boolesch	Gibt an, ob sich der Status												

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				geändert hat.

renameDirectory(src, dest)

Parameter:

Name	Typ	Beschreibung
src	Zeichenfolge	Der absolute Quellpfad auf dem Server, der das Verzeichnis identifiziert, das wir umbenennen möchten.
dest	Zeichenfolge	Der absolute Zielpfad auf dem Server, der den Zielpfad und den Verzeichnisnamen angibt.

renameFile(src, dest)

Parameter:

Name	Typ	Beschreibung
src	Zeichenfolge	Der absolute Quellpfad auf dem Server, der die Datei identifiziert, die wir umbenennen möchten.
dest	Zeichenfolge	Der absolute Zielpfad auf dem Server, der den Zielpfad und den Dateinamen angibt.

ResolutionErrorCode

Die im Modul verfügbaren ResolutionError Codeaufzählungen DCV

- INVALID_ARGUMENT
- NO_CHANNEL
- NOT_IMPLEMENTED

Type:

- Zahl

retrieveFile(Pfad)

Parameter:

Name	Typ	Beschreibung
path	Zeichenfolge	Der absolute Pfad auf dem Server, der die Datei identifiziert, die wir lokal herunterladen möchten.

screenshotCallback(Bildschirmfoto)

Die Callback-Funktion, die aufgerufen werden soll, wenn ein Screenshot verfügbar ist.

Parameter:

Name	Typ	Beschreibung
screenshot	byte[]	Screenshot-Puffer im PNG Format oder null wenn das Abrufen des Screenshots fehlgeschlagen ist.

ScreenshotErrorCode

Die im ScreenshotError Modul verfügbaren Code-Aufzählungen DCV

- NO_CHANNEL
- GENERIC_ERROR

Type:

- Zahl

serverInfo

Type:

- Object

Eigenschaften:

Name	Typ	Beschreibung									
name	Zeichenfolge	Der Name der Software.									
version	Object	Die Versionsnummer der Software. <table border="1" data-bbox="1068 1367 1528 1875"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>major</td> <td>Zahl</td> <td>Die Hauptversionsnummer.</td> </tr> <tr> <td>minor</td> <td>Zahl</td> <td>Die Nummer der</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	major	Zahl	Die Hauptversionsnummer.	minor	Zahl	Die Nummer der
Name	Typ	Beschreibung									
major	Zahl	Die Hauptversionsnummer.									
minor	Zahl	Die Nummer der									

Name	Typ	Beschreibung									
		<table border="1"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td>Nebenversion.</td> </tr> <tr> <td>revision</td> <td>Zahl</td> <td>Die Versionsnummer der Revision.</td> </tr> </tbody> </table>	Name	Typ	Beschreibung			Nebenversion.	revision	Zahl	Die Versionsnummer der Revision.
Name	Typ	Beschreibung									
		Nebenversion.									
revision	Zahl	Die Versionsnummer der Revision.									
os	Zeichenfolge	Das Betriebssystem.									
arch	Zeichenfolge	Die Architektur.									
hostname	Zeichenfolge	Der Hostname									

stats

Type:

- Object

Eigenschaften:

Name	Typ	Beschreibung
fps	Zahl	Die aktuellen Frames pro Sekunde.
traffic	Zahl	Der aktuelle Verkehr in Bit/s.
peakTraffic	Zahl	Der Spitzenwert des Datenverkehrs in Bit/s seit dem Verbindungsaufbau.

Name	Typ	Beschreibung
latency	Zahl	Die aktuelle Latenz in ms.
currentChannels	Zahl	Die Anzahl der Kanäle, die seit dem Verbindungsaufbau geöffnet wurden.
openedChannels	Zahl	Die Anzahl der aktuell geöffneten Kanäle.
channelErrors	Zahl	Die Anzahl der Kanäle, die einen Fehler gemeldet haben.

storeFile(Datei, Verzeichnis)

Parameter:

Name	Typ	Beschreibung
file	Datei	Das Datei-Objekt (weitere Informationen finden Sie unter https://developer.mozilla.org/en-US/docs/Web/API/File), das wir auf den Server hochladen wollen.
dir	Zeichenfolge	Der absolute Pfad auf dem Server, auf den wir die Datei hochladen möchten.

TimezoneRedirectionErrorCode

Die im TimezoneRedirectionError Modul verfügbaren Code-Aufzählungen DCV

- INVALID_ARGUMENT
- NO_CHANNEL

- USER_CANNOT_CHANGE

Type:

- Zahl

TimezoneRedirectionSettingCode

Die im TimezoneRedirectionSetting Modul verfügbaren Code-Aufzählungen DCV

- ALWAYS_OFF
- ALWAYS_ON
- CLIENT_DECIDES

Type:

- Zahl

TimezoneRedirectionStatusCode

Die im TimezoneRedirectionStatus Modul verfügbaren Code-Aufzählungen DCV

- SUCCESS
- PERMISSION_ERROR
- GENERIC_ERROR

Type:

- Zahl

WebcamErrorCode

Die im WebcamError Modul verfügbaren Code-Aufzählungen DCV

- SETTING_WEBCAM_FAILED
- CHANNEL_NOT_AVAILABLE

Type:

- Zahl

Verbindungsklasse

Die Verbindungsklasse, die durch den Aufruf der [connectMethode](#) des dcv Moduls abgerufen wurde. Ein Beispiel zur Verwendung finden Sie im Abschnitt [Erste Schritte](#).

Enthält

- [Methoden](#)

Methoden

Auflisten

- [attachDisplay\(SiegidisplayConf\)](#), → {Versprechen. <number>|Versprechen. < {Code: MultiMonitorErrorCode, Nachricht: Zeichenfolge} >}
- [captureClipboardEvents\(aktiviert, Win,displayId\)](#) → {void}
- [detachDisplay\(displayId\)](#) → {void}
- [disconnect \(\)](#) → {void}
- [disconnectCollaborator\(connectionId\)](#) → {void}
- [enableDisplayQualityUpdates \(aktivieren\)](#) → {void}
- [enableHighPixelDichte \(aktivieren\)](#) → {void}
- [enableTimezoneRedirection\(aktivieren\)](#) → {Promise|Promise. < {Code: TimezoneRedirectionErrorCode, Nachricht: Zeichenfolge} >}
- [enterRelativeMouseModus \(\)](#) → {void}
- [getConnectedDevices\(\)](#) → {Versprechen. <Reihe. < MediaDeviceInfo >>|Versprechen. < {message: string} >}
- [getFileExplorer\(\)](#) → {Versprechen. < Dateispeicher >|Promise. < {Code: ChannelErrorCode, Nachricht: Zeichenfolge} >}
- [getServerInfo\(\)](#) → {serverInfo}
- [getScreenshot\(\)](#) → {Versprechung|Versprechen. < {Code: ScreenshotErrorCode, Nachricht: Zeichenfolge} >}

- [getStats\(\)](#) → {Statistiken}
- [latchModifierKey\(Schlüssel, Ort,isDown\)](#) → {boolean}
- [openChannel\(Name, RückrufeauthToken, Namespace\)](#) → {Promise|Promise. < {Code: ChannelErrorCode, Nachricht: Zeichenfolge} >}
- [queryFeature\(featureName\)](#) → {Versprechen. < {aktiviert: boolean, remote? : Zeichenfolge,? autoCopy : boolesch,? autoPaste : boolesch,? serviceStatus : Zeichenfolge, verfügbar? : boolean} >|Versprechen. < {message: string} >}
- [registerKeyboardShortcuts\(Abkürzungen\)](#) → {void}
- [requestDisplayConfig\(highColorAccuracy\)](#) → {Versprechung|Versprechen. < {Code: DisplayConfigErrorCode, Nachricht: Zeichenfolge} >}
- [requestDisplayLayout\(Layout\)](#) → {Versprechen|Versprechen. < {Code: ResolutionErrorCode, Nachricht: Zeichenfolge} >}
- [requestResolution\(Breite, Höhe\)](#) → {Promise|Promise. < {Code: ResolutionErrorCode, Nachricht: Zeichenfolge} >}
- [sendKeyboardEvent\(Ereignis\)](#) → {boolean}
- [sendKeyboardShortcut\(Abkürzung\)](#) → {void}
- [setDisplayQuality\(min, maxopt\)](#) → {void}
- [setDisplayScale\(scaleRatio,displayId\)](#) → {Versprechen|Versprechen. < {Code: ResolutionErrorCode, Nachricht: Zeichenfolge} >} () DEPRECATED
- [setKeyboardQuirks\(Macken\)](#) → {void}
- [setMaxDisplayAuflösung \(maxWidth,maxHeight\)](#) → {void}
- [setMicrophone\(aktivieren\)](#) → {Promise|Promise. < {Code: AudioErrorCode, Nachricht: Zeichenfolge} >}
- [setMinDisplayAuflösung \(minWidth,minHeight\)](#) → {void}
- [setUploadBandwidth\(Wert\)](#) → {Zahl}
- [setVolume\(Volumen\)](#) → {void}
- [setMicrophone\(aktivieren,deviceId\)](#) → {Promise|Promise. < {Code: AudioErrorCode, Nachricht: Zeichenfolge} >}
- [setWebcam\(aktivieren,deviceId\)](#) → {Versprechen|Versprechen. < {Code: WebcamErrorCode, Nachricht: Zeichenfolge} >}
- [syncClipboards\(\)](#) → {boolean}

`attachDisplay(SiegdisplayConf)`, → {Versprechen. <number>|Versprechen. < {Code: [MultiMonitorErrorCode](#), Nachricht: Zeichenfolge} >}

Fügt einem Fenster eine bestimmte Anzeige hinzu. Sie können das Hauptdisplay nicht anschließen. Bei Erfolg gibt die Funktion den `zurückdisplayId`.

Parameter:

Name	Typ	Beschreibung			
<code>win</code>	Object	Das Fenster, an das das Display angeschlossen werden muss.			
<code>displayConf</code>	Object	Die Konfiguration des Displays.			
		Name	Typ	Attribu	Beschreib ung
		<code>displ</code>	Zahl	<optio >	Die ID des Displays.
		<code>displ vName</code>			Der Name des Display-D ivs.

Rückgabe:

Verspricht es. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen. <number>| Versprechen. < {Code: [MultiMonitorErrorCode](#), Nachricht: Zeichenfolge}>

`captureClipboardEvents(aktiviert, Win,displayId) → {void}`

Startet oder beendet das Abhören von Copy-Paste-Ereignissen. Bei interaktiven Zwischenablagen (immer beim Einfügen) müssen wir anfangen, die Copy/Paste-Ereignisse abzuhören. Es könnte nützlich sein, das Abhören nur dann zu starten und zu beenden, wenn es nötig ist, zum Beispiel wenn ein Modal angezeigt wird.

Parameter:

Name	Typ	Attribute	Beschreibung
<code>enabled</code>	boolesch		Um mit dem Abhören von Ereignissen zu beginnen, geben Sie <code>true</code> . Um das Abhören von Ereignissen zu beenden, geben Sie <code>false</code> .
<code>win</code>	Object	<optional>	Das Fenster, in dem Sie nach Ereignissen Ausschau halten können. Wenn es weggelassen wird, wird das Standardfenster verwendet.
<code>displayId</code>	Zahl	<optional>	Die ID des Displays, das die Ereignisse abhören soll. Wenn nicht angegeben, wird die Standard-

Name	Typ	Attribute	Beschreibung
			nzeige des Fensters verwendet.

Rückgabe:

Typ

void

`detachDisplay(displayId) → {void}`

Trennt ein bestimmtes Display ab. Das Hauptdisplay kann nicht abgenommen werden.

Parameter:

Name	Typ	Beschreibung
displayId	Zahl	Die ID des Displays, das abgenommen werden soll.

Rückgabe:

Typ

void

`disconnect () → {void}`

Trennt die Verbindung zum DCV Amazon-Server und schließt die Verbindung.

Rückgabe:

Typ

void

disconnectCollaborator(connectionId) → {void}

Fordert die Trennung eines Mitarbeiters an, der mit der angegebenen Verbindungs-ID verbunden ist (seit Amazon DCV Web Client SDK Version 1.1.0).

Parameter:

Name	Typ	Beschreibung
connectionId	boolesch	Die ID der Verbindung, die getrennt wird.

Rückgabe:

Typ

void

enableDisplayQualityUpdates (aktivieren) → {void}

Aktiviert oder deaktiviert Updates zur Anzeigequalität für Streaming-Bereiche, die keine Updates erhalten. Das Deaktivieren von Updates zur Anzeigequalität reduziert die Bandbreitennutzung, verringert aber auch die Anzeigequalität.

Parameter:

Name	Typ	Beschreibung
enable	boolesch	Um Aktualisierungen der Anzeigequalität zu aktivieren, geben Sie an <code>true</code> . Geben Sie an, um Aktualisierungen der Anzeigequalität zu deaktivieren <code>false</code> .

Rückgabe:

Typ

void

`enableHighPixelDichte (aktivieren) → {void}`

Aktiviert oder deaktiviert die hohe Pixeldichte auf dem Client.

Parameter:

Name	Typ	Beschreibung
<code>enable</code>	boolesch	Ob eine hohe Pixeldichte aktiviert werden soll oder nicht.

Rückgabe:

Typ

void

`enableTimezoneRedirection(aktivieren) → {Promise|Promise. < {Code: TimezoneRedirectionErrorCode, Nachricht: Zeichenfolge} >}`

Aktiviert oder deaktiviert die Zeitzonenumleitung. Sobald sie aktiviert ist, fordert der Client den Server auf, die Zeitzone des Server-Desktops an die Zeitzone des Clients anzupassen.

Parameter:

Name	Typ	Beschreibung
<code>enable</code>	boolesch	Geben Sie an, um die Zeitzonenumleitung zu aktivieren. <code>true</code> Geben Sie

Name	Typ	Beschreibung
		an, um die Zeitzonenumleitung zu deaktivieren. <code>false</code>

Rückgabe:

Verspricht es. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen. `<number>` | Versprechen. `< {Code: TimezoneRedirectionErrorCode, Nachricht: Zeichenfolge} >`

`enterRelativeMouseModus () → {void}`

Aktiviert den relativen Mausmodus.

Rückgabe:

Typ

`void`

`getConnectedDevices() → {Versprechen. <Reihe. < MediaDeviceInfo >>|Versprechen. < {message: string} >}`

Fordert eine Liste der Mediengeräte an, die mit dem Client-Computer verbunden sind.

Rückgabe:

Bei Erfolg wird ein Promise zurückgegeben, das in ein Array von `MediaDeviceInfo` Objekten aufgelöst wird. Weitere Informationen finden Sie unter <https://developer.mozilla.org/en-US/docs/Web/API/MediaDeviceInfo>. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen. `<Reihe. < MediaDeviceInfo >> | Versprochen. < {message: string} >`

`getFileExplorer()` → {Versprechen. < [Dateispeicher](#) > | [Promise](#). < {Code: [ChannelErrorCode](#), Nachricht: Zeichenfolge} >}

Ruft ein Objekt zur Verwaltung des Dateispeichers des DCV Amazon-Servers ab.

Rückgabe:

Verspricht es. Löst das Ergebnis in das Datei-Explorer-Objekt auf, wenn es erfüllt ist, oder in ein Fehlerobjekt, wenn es abgelehnt wird.

Typ

Verspricht es. < [Dateispeicher](#) > | [Versprechen](#). < {Code: [ChannelErrorCode](#), Nachricht: Zeichenfolge} >

`getServerInfo()` → {[serverInfo](#)}

Ruft Informationen über den DCV Amazon-Server ab.

Rückgabe:

Informationen über die Serversoftware.

Typ

[serverInfo](#)

`getScreenshot()` → {Versprechung | Versprechen. < {Code: [ScreenshotErrorCode](#), Nachricht: Zeichenfolge} >}

Ruft den Screenshot des Remote-Desktops im PNG Format ab. Der Screenshot wird im [screenshotCallback](#)Observer zurückgegeben. `null` wird stattdessen im Falle von Fehlern zurückgegeben.

Rückgabe:

Versprechen, das gelöst wird, wenn die Anfrage bearbeitet wird. Bei Ablehnung erhalten wir ein Fehlerobjekt.

Typ

Versprechen | Versprechen. < {Code: [ScreenshotErrorCode](#), Nachricht: Zeichenfolge} >

`getStats()` → [{Statistiken}](#)

Ruft Statistiken über den DCV Amazon-Server ab.

Rückgabe:

Informationen zu den Streaming-Statistiken.

Typ

[Statistiken](#)

`latchModifierKey(Schlüssel, Ort,isDown)` → `{boolean}`

Sendet eine einzelne Tastatur keydown oder ein einzelnes keyup Ereignis für einen zulässigen Modifikator.

Parameter:

Name	Typ	Beschreibung
<code>key</code>	Steuerung Alt Meta AltGraph OS Shift	Der Schlüssel zum Senden.
<code>location</code>	KeyboardEvent. Standort	Der Standort des Schlüssels. Weitere Informationen finden Sie unter https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/location .
<code>isDown</code>	boolesch	Wenn das einzufügende Schlüsselereignis ein Keydown (<code>true</code>) oder ein Keyup (<code>isDown</code>) ist. <code>false</code>

Rückgabe:

Wenn die angeforderte Kombination gültig ist, kehrt die Funktion zurück `true`, andernfalls kehrt sie zurück. `false`

Typ

boolesch

`openChannel(Name, RückrufeauthToken, Namespace) → {Promise|Promise. < {Code: ChannelErrorCode, Nachricht: Zeichenfolge} >}`

Öffnet einen benutzerdefinierten Datenkanal für die Verbindung, falls er auf dem DCV Amazon-Server erstellt wurde.

Parameter:

Name	Typ	Beschreibung
name	Zeichenfolge	Der Name des Kanal.
authToken	Zeichenfolge	Das Authentifizierungstoken, das für die Verbindung mit dem Kanal verwendet werden soll.
callbacks	Object	Die onMessage onClose aufzurufenden Funktionen und Callbacks.
namespace	Zeichenfolge	Der Namespace des Kanals. Verfügbar seit Amazon DCV Web Client SDK 1.2.0 und Amazon DCV Server 2022.1.

Rückgabe:

Versprochen. Bei Ablehnung erhalten wir ein Fehlerobjekt.

Typ

Versprechen | Versprechen. < {Code: [ChannelErrorCode](#), Nachricht: Zeichenfolge} >

```
queryFeature(featureName) → {Versprechen. < {aktiviert: boolean, remote? :
Zeichenfolge,? autoCopy : boolesch,? autoPaste : boolesch,? serviceStatus :
Zeichenfolge, verfügbar? : boolean} >|Versprechen. < {message: string} >}
```

Fragt den Status einer bestimmten DCV Amazon-Serverfunktion ab.

Parameter:

Name	Typ	Beschreibung
featureName	Funktion	Der Name des Features, das abgefragt werden soll.

Rückgabe:

Verspricht es. Bei Auflösung gibt die Funktion ein `status` Objekt zurück, das immer eine `enabled` Eigenschaft und möglicherweise auch andere Eigenschaften enthält. Bei Ablehnung gibt die Funktion ein `error` Objekt zurück.

Typ

```
{Versprechen. < {aktiviert: boolean, remote? : Zeichenfolge,? autoCopy : boolesch,? autoPaste :
boolesch,? serviceStatus : Zeichenfolge, verfügbar? : boolean} > | Versprechen. < {message:
string} >
```

```
registerKeyboardShortcuts(Abkürzungen) → {void}
```

Registriert Tastenkombinationen.

Parameter:

Name	Typ	Beschreibung
shortcuts	Reihe. <Object>	Das Array von Schlüsseln und Zuordnungen, die registriert werden sollen.

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
		sequenc	Reihe. <Object>	Die Tastenkombination zur Registrierung.
		k k v	k k v	<p>BoardE Key der vom Benutzer gedrückte n Taste. Weitere Informationen finden Sie unter <a 120="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://d
eveloper.
mozilla.o
rg/
en-
US/</p> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Methoden</p> </p>

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				N T Beschreibung ung docs/ Web/ API/ Keybo ardEvent/ key. 1 k DiscardE v Array ti der zu sendender Schlüssel . Die Position der Taste auf der Tastatur. Weitere Informati onen finden Sie unter <a 120="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://d
eveloper.</td></tr></tbody></table></div><div data-bbox="><p>Methoden</p>

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				N T Beschreibung mozilla.org/en-US/docs/Web/API/KeyboardEvent.location.
		output	Reihe. <Object>	Die beabsichtigte Aktion, die mit der Tastenkombination ausgeführt werden soll.

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				<p>NT Beschreibung</p> <p>kk BoardE</p> <p>vv Wert</p> <p>ε Schlüssel vom Benutzer gedrückte n Taste. Weitere Informationen finden Sie unter https://d developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/key.</p>

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				<p>NT Beschreibung</p> <p>1 k DisardE</p> <p>v Array</p> <p>ti der</p> <p>zu</p> <p>sendenden</p> <p>Schlüssel</p> <p>.</p> <p>Die</p> <p>Position</p> <p>der</p> <p>Taste</p> <p>auf</p> <p>der</p> <p>Tastatur.</p> <p>Weitere</p> <p>Informati</p> <p>onen</p> <p>finden</p> <p>Sie</p> <p>unter</p> <p><a 121="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://d</p> <p>eveloper.</p> <p>mozilla.o</p> <p>rg/</p> <p>en-</p> <p>US/</p> <p>docs/</p> <p>Web/</p> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Methoden</p> </p>

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				N T Beschreibung
				API/ KeyboardEvent/ location.

Rückgabe:

Typ

void

`requestDisplayConfig(highColorAccuracy) → {Versprechung|Versprechen. < {Code: DisplayConfigErrorCode, Nachricht: Zeichenfolge} >}`

Fordert eine aktualisierte Display-Konfiguration vom DCV Amazon-Server an. Verfügbar seit Amazon DCV Web Client SDK 1.1.0 und Amazon DCV Server 2022.0.

Parameter:

Name	Typ	Beschreibung
<code>highColorAccuracy</code>	boolesch	Ob eine hohe Farbgenauigkeit verlangt werden soll oder nicht.

Rückgabe:

Verspricht es. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen | Versprechen. < {Code: [DisplayConfigErrorCode](#), Nachricht: Zeichenfolge} >

requestDisplayLayout(Layout) → {Versprechen|Versprechen. < {Code: [ResolutionErrorCode](#), Nachricht: Zeichenfolge} >}

Fordert ein aktualisiertes Display-Layout für die Verbindung an.

Parameter:

Name	Typ	Beschreibung
layout	Reihe. < Überwachen >	Die angeforderten Anzeigen werden im Layout angezeigt.

Rückgabe:

Verspricht es. Bei Ablehnung erhalten wir ein Fehlerobjekt.

Typ

Versprechen | Versprechen. < {Code: [ResolutionErrorCode](#), Nachricht: Zeichenfolge} >

requestResolution(Breite, Höhe) → {Promise|Promise. < {Code: [ResolutionErrorCode](#), Nachricht: Zeichenfolge} >}

Fordert eine aktualisierte Bildschirmauflösung vom DCV Amazon-Server an.

Parameter:

Name	Typ	Beschreibung
width	Zahl	Die anzufordernde Breite in Pixeln. Der zulässige Mindestwert ist 0.

Name	Typ	Beschreibung
height	Zahl	Die anzufordernde Höhe in Pixeln. Der zulässige Mindestwert ist 0.

Rückgabe:

Verspricht es. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen | Versprechen. < {Code: [ResolutionErrorCode](#), Nachricht: Zeichenfolge} >

sendKeyboardEvent(Ereignis) → {boolean}

Sendet ein Tastaturkürzel-Ereignis. Weitere Informationen zu Tastaturereignissen finden Sie unter <https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent>. Zu den gültigen Tastaturereignissen gehören: `keydown`, `keypress`, und `keyup`. Weitere Informationen zu diesen Ereignissen finden Sie unter <https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent#events>.

Parameter:

Name	Typ	Beschreibung
event	KeyboardEvent	Das zu sendende Tastaturereignis.

Rückgabe:

Wenn das Ereignis nicht gültig ist, kehrt die Funktion zurück `false`. Wenn das Ereignis gültig ist, kehrt die Funktion zurück `true`.

Typ

boolesch

sendKeyboardShortcut(Abkürzung) → {void}

Sendet eine Tastenkombination. Verwenden Sie diese Funktion, um eine vollständige keydown oder eine keyup Sequenz zu senden. Wenn Sie beispielsweise Strg + Alt + Del senden, werden die keydown Ereignisse für alle Tasten gesendet, gefolgt von den keyup Ereignissen. Verwenden Sie diese Funktion auch dann, wenn Sie eine einzelne Taste senden möchten.

Parameter:

Name	Typ	Beschreibung						
shortcut	Reihe. <Object>	Das Array der zu sendenden Schlüssel. <table border="1" data-bbox="1068 789 1528 1879"> <thead> <tr> <th>Name</th> <th>Typ</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>key</td> <td>Keyboardvent. Schlüssel</td> <td>Der Wert der vom Benutzer gedrückten Taste. Weitere Informationen finden Sie unter https:// developer. mozilla. org/en-US/</td> </tr> </tbody> </table>	Name	Typ	Beschreibung	key	Keyboardvent. Schlüssel	Der Wert der vom Benutzer gedrückten Taste. Weitere Informationen finden Sie unter https:// developer. mozilla. org/en-US/
Name	Typ	Beschreibung						
key	Keyboardvent. Schlüssel	Der Wert der vom Benutzer gedrückten Taste. Weitere Informationen finden Sie unter https:// developer. mozilla. org/en-US/						

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				docs/ Web/ API/ KeyboardEvent/ key.
		location	KeyboardEvent.location	Das Array der zu sendenden Schlüssel. Die Position der Taste auf der Tastatur. Weitere Informationen finden Sie unter https:// developer.mozilla.org/en-US/docs/Web/

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				API/KeyboardEvent/location.

Rückgabe:

Typ

void

`setDisplayQuality(min, maxopt) → {void}`

Legt die für die Verbindung zu verwendende Bildqualität fest. Gültiger Bereich ist 0 bis 100, 1 wobei die niedrigste Bildqualität und die höchste Bildqualität angegeben 100 werden. Geben Sie 0 an, ob der aktuelle Wert beibehalten werden soll.

Parameter:

Name	Typ	Attribute	Beschreibung
min	Zahl		Die minimale Bildqualität.
max	Zahl	<optional>	Die maximale Bildqualität.

Rückgabe:

Typ

void

`setDisplayScale(scaleRatio,displayId) → {Versprechen|Versprechen. < {Code: ResolutionErrorCode, Nachricht: Zeichenfolge} >} ()` DEPRECATED

Seit Version 1.3.0 veraltet. Die Anzeigeskala muss nicht mehr eingestellt werden. Mauskoordinaten werden automatisch intern verwaltet.

Informiert Amazon darüberDCV, dass die Anzeige auf der Client-Seite skaliert ist. Verwenden Sie diese Option, um den Server darüber zu informieren, dass er Mausereignisse skalieren muss, um sie an das Anzeigeverhältnis des Clients anzupassen.

Parameter:

Name	Typ	Beschreibung
<code>scaleRatio</code>	float	Das zu verwendende Skalierungsverhältnis. Muss eine strikt positive Zahl sein.
<code>displayId</code>	Zahl	Die ID des zu skalierenden Displays.

Rückgabe:

Versprech es. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen | Versprechen. < {Code: [ResolutionErrorCode](#), Nachricht: Zeichenfolge} >

`setKeyboardQuirks(Macken) → {void}`

Legt die Eigenheiten der Tastatur für den Client-Computer fest.

Parameter:

Name	Typ	Beschreibung
<code>quirks</code>	Object	Die Tastatur ist beim Aktivieren oder Deaktivieren eigenartig.

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
		macOptionsAlt	boolesch	Geben Sie an, um die Optionstate der Alt-Taste für macOS zuzuordnen. true. Andernfalls geben Sie false an.
		macCommandToControl	boolesch	Um die Befehlsstate Strg für macOS zuzuordnen, geben Sie an true. Andernfalls geben Sie

Name	Typ	Beschreibung		
		Name	Typ	Beschreibung
				false an.

Rückgabe:

Typ

void

`setMaxDisplayAuflösung (maxWidth,maxHeight) → {void}`

Legt die maximale Bildschirmauflösung fest, die für die Verbindung verwendet werden soll.

Parameter:

Name	Typ	Beschreibung
<code>maxWidth</code>	Zahl	Die maximale Anzeigebreite in Pixeln. Der zulässige Mindestwert ist 0.
<code>maxHeight</code>	Zahl	Die maximale Anzeigehöhe in Pixeln. Der zulässige Mindestwert ist 0.

Rückgabe:

Typ

void

`setMicrophone(aktivieren) → {Promise|Promise. < {Code: AudioErrorCode, Nachricht: Zeichenfolge} >}`

Aktiviert oder deaktiviert das Mikrofon.

Parameter:

Name	Typ	Beschreibung
<code>enable</code>	boolesch	Geben <code>true</code> Sie an, um das Mikrofon zu aktivieren. Um das Mikrofon zu deaktivieren, geben Sie <code>false</code> .

Rückgabe:

Versprech es. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen | Versprechen. < {Code: [AudioErrorCode](#), Nachricht: Zeichenfolge} >

`setMinDisplayAuflösung (minWidth,minHeight) → {void}`

Legt die minimale Bildschirmauflösung fest, die für die Verbindung verwendet werden soll. Für einige Anwendungen ist möglicherweise eine minimale Bildschirmauflösung erforderlich. Wenn die erforderliche Mindestauflösung größer als die vom Client unterstützte maximale Auflösung ist, wird eine Strategie zur Größenänderung verwendet. Verwenden Sie diese Funktion mit Bedacht. Die Strategie zur Größenänderung könnte zu einem weniger präzisen Maus- und Touch-Eingabesystem führen.

Parameter:

Name	Typ	Beschreibung
<code>minWidth</code>	Zahl	Die minimale Anzeigebreite in Pixeln. Der zulässige Mindestwert ist 0.

Name	Typ	Beschreibung
minHeight	Zahl	Die minimale Anzeigehöhe in Pixeln. Der zulässige Mindestwert ist 0.

Rückgabe:

Typ

void

setUpUploadBandwidth(Wert) → {Zahl}

Legt die maximale Bandbreite fest, die für das Hochladen von Dateien auf den Amazon-Server verwendet werden soll. DCV

Parameter:

Name	Typ	Beschreibung
value	Zahl	Die maximale Bandbreitenbeschränkung in kBit/s. Der gültige Bereich liegt zwischen 1024 kbit/s und 102400 kbit/s.

Rückgabe:

- Das festgelegte Bandbreitenlimit. null wenn die Dateispeicherfunktion auf dem Server deaktiviert ist.

Typ

Zahl

setVolume(Volumen) → {void}

Legt die Lautstärke fest, die für Audio verwendet werden soll. Der gültige Bereich liegt zwischen 0 und 100, wobei 0 die niedrigste Lautstärke und 100 die höchste Lautstärke ist.

Parameter:

Name	Typ	Beschreibung
volume	Zahl	Der zu verwendende Lautstärkepegel.

Rückgabe:

Typ

void

setMicrophone(aktivieren,deviceId) → {Promise|Promise. < {Code: [AudioErrorCode](#), Nachricht: Zeichenfolge} >}

[Experimentell — könnte sich in future ändern] Aktiviert oder deaktiviert das Mikrofon.

Parameter:

Name	Typ	Beschreibung
enable	boolesch	Um das Mikrofon zu aktivieren, geben Sie <code>true</code> . Um das Mikrofon zu deaktivieren, geben Sie <code>false</code> .
deviceId	Zeichenfolge	Die Geräte-ID des Mikrofons. Wenn kein angegeben <code>deviceId</code> ist, <code>default deviceId</code> wird das verwendet.

Rückgabe:

Verspricht es. Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen | Versprechen. < {Code: [AudioErrorCode](#), Nachricht: Zeichenfolge} >

`setWebcam(aktivieren,deviceId) → {Versprechen|Versprechen. < {Code: WebcamErrorCode, Nachricht: Zeichenfolge} >}`

Aktiviert oder deaktiviert die Webcam.

Parameter:

Name	Typ	Beschreibung
<code>enable</code>	boolesch	Geben <code>true</code> Sie an, um die Webcam zu aktivieren. Geben Sie an, um die Webcam zu deaktivieren <code>false</code> .
<code>deviceId</code>	Zeichenfolge	Die Geräte-ID der Webcam.

Rückgabe:

Versprechen Sie, dass bei Erfolg die Auflösung auf die angeschlossene/abgetrennte Webcam erfolgt. `deviceId` Wenn das Versprechen abgelehnt wird, gibt es ein Fehlerobjekt zurück.

Typ

Versprechen. <string> | Versprechen. < {Code: [WebcamErrorCode](#), Nachricht: Zeichenfolge} >

`syncClipboards() → {boolean}`

Synchronisiert die Zwischenablage des lokalen Clients mit der Zwischenablage des DCV Remote-Amazon-Servers. Autocopy muss vom Browser unterstützt werden.

Rückgabe:

Wenn die Zwischenablagen synchronisiert wurden, kehrt die Funktion zurück. `true` Wenn die Zwischenablagen nicht synchronisiert wurden oder wenn der Browser das automatische Kopieren nicht unterstützt, kehrt die Funktion zurück. `false`

Typ

boolesch

Authentifizierungsklasse

Die Authentifizierungsklasse muss verwendet werden, um ein Authentifizierungstoken zu erhalten, indem die [authenticateMethode](#) des dcv Moduls aufgerufen wird. Ein Beispiel zur Verwendung finden Sie im Abschnitt [Erste Schritte](#).

Enthält

- [Methoden](#)

Methoden

Auflisten

- [erneut versuchen \(\) → {void}](#)
- [sendCredentials\(Anmeldeinformationen\) → {void}](#)

[erneut versuchen \(\) → {void}](#)

Wiederholt den Authentifizierungsprozess.

Rückgabe:

Typ

void

[sendCredentials\(Anmeldeinformationen\) → {void}](#)

Sendet die vom Client bereitgestellten Authentifizierungsdaten an den DCV Amazon-Server.

Parameter:

Name	Typ	Beschreibung
<code>credentials</code>	Object	Das Objekt, das die angegebenen Anmeldeinformationen enthält. Die Anmeldeinformationen müssen denselben Namen und denselben Typ haben, der in der Abfrage angegeben wurde.

Rückgabe:

Typ

`void`

Ressourcenklasse

Die Ressourcenklasse kann die entsprechende Datei, die gerade gedruckt oder heruntergeladen wurde, abrufen oder verwerfen. Bei der Ausführung dieser Aktionen [fileDownload](#)würden die entsprechenden Observer-Funktionen [filePrinted](#)jeweils mit dem Ressourcenobjekt als einzigem Argument aufgerufen. Solche Ressourcen können akzeptiert oder abgelehnt werden, um die Datei, auf die sie verweisen, abzurufen oder zu verwerfen.

Enthüllt

- [Methoden](#)

Methoden

Auflisten

- [akzeptieren \(urlParameters\) → {void}](#)
- [ablehnen \(\) → {void}](#)

akzeptieren (urlParameters) → {void}

Lädt die Ressource lokal herunter.

Parameter:

Name	Typ	Beschreibung
<code>urlParameters</code>	Object	Das optionale Objekt, das die Schlüssel/Wert-Paare der URL Suchparameter enthält, die an die Anforderung zum Abrufen der Ressource übergeben wurden.

Rückgabe:

Typ

`void`

ablehnen () → {void}

Wirft die Ressource weg.

Rückgabe:

Typ

`void`

DCVAmazon-Webbenutzeroberfläche SDK

Eine JavaScript React-Komponentenbibliothek, die derzeit eine einzelne React-Komponente namens `exportiertDCVViewer`, die eine Verbindung zum DCV Amazon-Server herstellt und die Toolbar rendert, um mit dem Remote-Stream zu interagieren.

Macht verfügbar

- [Komponenten](#)

Komponenten

Auflisten

- [DCVViewer](#)

DCVViewer

Die React-Komponente rendert die Werkzeugleiste mit all ihren Funktionen, die für die Interaktion mit dem Remote-Stream nützlich sind.

Eigenschaften:

Auflisten

- [dcv](#)
- [uiConfig](#)

dcv

Name	Typ	Erforderlich	Beschreibung
dcv	Object	Ja	Das Objekt definiert die Eigenschaften, die zum Herstellen der Verbindung zum DCV Amazon-Server erforderlich sind, legt die Protokollebene fest und legt fest, URL von wo aus die Amazon DCV Web SDK Client-Ressourcen geladen und auf die DCV

Name	Typ	Erforderlich	Beschreibung																
			<p>Ressourcen zugegriffen werden sollen.</p> <table border="1" data-bbox="1187 352 1604 478"> <thead> <tr> <th data-bbox="1187 352 1268 478">Name</th> <th data-bbox="1268 352 1349 478">Typ</th> <th data-bbox="1349 352 1430 478">Erforderlich</th> <th data-bbox="1430 352 1604 478">Beschreibung</th> </tr> </thead> <tbody> <tr> <td data-bbox="1187 478 1268 846">ses</td> <td data-bbox="1268 478 1349 846">Strir</td> <td data-bbox="1349 478 1430 846">Ja</td> <td data-bbox="1430 478 1604 846">Die DCV Amazon-Sitzungs-ID.</td> </tr> <tr> <td data-bbox="1187 846 1268 1591">aut</td> <td data-bbox="1268 846 1349 1591">Strir</td> <td data-bbox="1349 846 1430 1591">Ja</td> <td data-bbox="1430 846 1604 1591">Das Authentifizierungstoken, das für die Verbindung mit dem Server verwendet werden soll.</td> </tr> <tr> <td data-bbox="1187 1591 1268 1848">ser</td> <td data-bbox="1268 1591 1349 1848">Strir</td> <td data-bbox="1349 1591 1430 1848">Ja</td> <td data-bbox="1430 1591 1604 1848">Der Hostname und der Port.</td> </tr> </tbody> </table>	Name	Typ	Erforderlich	Beschreibung	ses	Strir	Ja	Die DCV Amazon-Sitzungs-ID.	aut	Strir	Ja	Das Authentifizierungstoken, das für die Verbindung mit dem Server verwendet werden soll.	ser	Strir	Ja	Der Hostname und der Port.
Name	Typ	Erforderlich	Beschreibung																
ses	Strir	Ja	Die DCV Amazon-Sitzungs-ID.																
aut	Strir	Ja	Das Authentifizierungstoken, das für die Verbindung mit dem Server verwendet werden soll.																
ser	Strir	Ja	Der Hostname und der Port.																

Name	Typ	Erforderlich	Beschreibung			
			Name	Typ	Erforderlich	Beschreibung
			base	String	Ja	Das Absolute oder Relative, URL aus dem Dateien geladen werden

des laufenden DCV Amazon-Server im folgenden Format: `https://<dcv_host_address>:port`. Zum Beispiel: `https://8443-dcv-server`.

Name	Typ	Erforderlich	Beschreibung			
			Name	Typ	Erforderlich	Beschreibung
			resource	String	Nein	Das Absolute oder Relative, URL von dem aus auf DCV Ressourcen zugegriffen werden soll.
			onDemand	function	Nein	Die Callback-Funktion wird aufgerufen, wenn die Verbindung zum DCV Amazon-

Name	Typ	Erforderlich	Beschreibung
			<p>Server getrennt wird und die Verbindung geschlossen wird.</p> <p>log Log! Nein Die (Staprotokollebene, die 1. INF im Viewer verwendet werden soll.</p>

uiConfig

Name	Typ	Erforderlich	Beschreibung
uiConfig	Object	Nein (Standard:{})	Das Objekt, das die Eigenschaften definiert, um zu konfigurieren, ob die Werkzeugleiste

Name	Typ	Erforderlich	Beschreibung												
			<p>sichtbar ist und ob die Vollbild- und Multimonitorschaltflächen darauf angezeigt werden sollen.</p> <table border="1" data-bbox="1187 541 1511 667"> <thead> <tr> <th data-bbox="1187 541 1268 667">Name</th> <th data-bbox="1268 541 1349 667">Typ</th> <th data-bbox="1349 541 1430 667">Erforderlich</th> <th data-bbox="1430 541 1511 667">Beschreibung</th> </tr> </thead> <tbody> <tr> <td data-bbox="1187 667 1268 1360">true</td> <td data-bbox="1187 667 1268 1360">Boolean</td> <td data-bbox="1187 667 1268 1360">Nein</td> <td data-bbox="1187 667 1268 1360">Das Objekt, das die Konfigurationsoptionen für die Werkzeugleiste definiert.</td> </tr> <tr> <td data-bbox="1187 1360 1268 1881">true</td> <td data-bbox="1187 1360 1268 1881">Boolean</td> <td data-bbox="1187 1360 1268 1881">Nein</td> <td data-bbox="1187 1360 1268 1881">Das Objekt, das die Konfigurationsoptionen für die Werkzeugleiste definiert.</td> </tr> </tbody> </table>	Name	Typ	Erforderlich	Beschreibung	true	Boolean	Nein	Das Objekt, das die Konfigurationsoptionen für die Werkzeugleiste definiert.	true	Boolean	Nein	Das Objekt, das die Konfigurationsoptionen für die Werkzeugleiste definiert.
Name	Typ	Erforderlich	Beschreibung												
true	Boolean	Nein	Das Objekt, das die Konfigurationsoptionen für die Werkzeugleiste definiert.												
true	Boolean	Nein	Das Objekt, das die Konfigurationsoptionen für die Werkzeugleiste definiert.												

Name	Typ	Erforderlich	Beschreibung			
			Name	Typ	Erforderlich	Beschreibung
						Erforderlich
						Werkzeugeinstellungen werden ausgeblendet werden soll.
						Microsoft (Standard: true)
						zu definieren, ob die Vollbild-Schaltfläche in der Werkzeugleiste ein- oder ausgeblendet

Name	Typ	Erforderlich	Beschreibung			
			Name	Typ	Erforderlich	Beschreibung
						Erforderlich
						werden soll.
						Nichtsonst (Optional: true)
						zu definieren, ob die Multimonitor-Schaltfläche in der Werkzeugleiste ein- oder ausgeblendet werden soll.

Versionshinweise und Dokumentverlauf für Amazon DCV Web Client SDK

Diese Seite enthält die Versionshinweise und den Dokumentverlauf für Amazon DCV Web Client SDK.

Themen

- [SDK Versionshinweise zum Amazon DCV Web Client](#)
- [Dokumentverlauf](#)

SDK Versionshinweise zum Amazon DCV Web Client

Dieser Abschnitt enthält Versionshinweise für den Amazon DCV Web Client SDK nach Veröffentlichungsdatum.

Themen

- [1.8.4 — 1. Oktober 2024](#)
- [1.5.10 — 19. Dezember 2023](#)
- [1.5.6 — 9. November 2023](#)
- [1.4.4 — 29. Juni 2023](#)
- [1.4.0 — 28. März 2023](#)
- [1.3.1 — 9. Dezember 2022](#)
- [1.3.0 — 11. November 2022](#)
- [1.2.1 — 21. Juli 2022](#)
- [1.2.0 — 29. Juni 2022](#)
- [1.1.3 — 23. Mai 2022](#)
- [1.1.2 — 19. Mai 2022](#)
- [1.1.1 — 23. März 2022](#)
- [1.1.0 — 23. Februar 2022](#)
- [1.0.4 — 20. Dezember 2021](#)
- [1.0.3 — 01. September 2021](#)

- [1.0.2 — 30. Juli 2021](#)
- [1.0.1 — 31. Mai 2021](#)
- [1.0.0 — 24. März 2021](#)

1.8.4 — 1. Oktober 2024

Build-Nummern	Neue Features	Änderungen und Fehlerbehebungen
<ul style="list-style-type: none"> • Semantische Version: 1.8.4 • Bau: 840 	<p>Die folgenden Funktionen wurden hinzugefügt:</p> <ul style="list-style-type: none"> • Umbenannt in „Amazon DCV Web ClientSDK“ • Neue API enableHighPixel Dichte für Displays mit hohem DPI-Wert hinzugefügt • Es wurde ein Experiment API setMicrophone zur Auswahl des Mikrofons in kompatiblen Browsern hinzugefügt • Neue Verbindungsfehler GATEWAY _ BUSY CREDENTIAL, UNSUPPORTED _ und TRANSPORT _ hinzugefügt ERROR • Neue Abschlussgründe EXTERNAL _ PROTOCOL _ CONNECTION _ EVICTED und DISCONNECTION _ hinzugefügt REQUESTED 	<ul style="list-style-type: none"> • Verbesserte Webcam-Handhabung • Verbesserte Handhabung der Audiowiedergabe • Verbesserte WebCodecs Handhabung • Verbessertes Ein- und Ausstecken von Mikrophon und Webcam • Verbessertes Ziehen von Fenstern aus der Ferne bei Verwendung eines Multimonitors • Die Upload- und Download-Rechte für den Dateispeicher werden jetzt korrekt weitergegeben • Kleinere Korrekturen beim Rendern

Version	Versionshinweise
<ul style="list-style-type: none">Semantische Version: 1.8.4Bau: 840	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">

1.5.10 — 19. Dezember 2023

Version	Versionshinweise
<ul style="list-style-type: none">Semantische Version: 1.5.10Version: 684	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">Fehler bei der Stream-Dekodierung beheben

1.5.6 — 9. November 2023

Version	Versionshinweise
<ul style="list-style-type: none">Semantische Version: 1.5.6Version: 659	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">Leistungsverbesserungen beim Decodieren und Rendern von StreamsDie Unterstützung für Internet Explorer 11 wurde entfernt

1.4.4 — 29. Juni 2023

Version	Versionshinweise
---------	------------------

Version	Versionshinweise
<ul style="list-style-type: none">Semantische Version: 1.4.4Version: 573	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">Die Viewer-UI-Komponente verwendet jetzt die eigenen Browser, die sie unterstützen, um Sondertasten im navigator <code>.keyboard.lock</code> API Vollbildmodus zu verarbeiten.Es wurde ein Problem behoben, das bei Verwendung von Chrome 114 oder neuer zu falschen Farben führen konnte.Verbesserte WebCodecs Erkennung.Ein Problem mit dem Zustand der Maustaste beim Betreten des Fensters wurde behoben.Es wurde ein Problem behoben, das dazu führen konnte, dass die Sondertasten auf macOS gedrückt blieben.Die Robustheit der Audiowiedergabe gegenüber verschlechterten Netzwerkbedingungen wurde verbessert.Speicherlecks wurden behoben.Verbesserte Logs mit Angabe von Zeit und Level.

1.4.0 — 28. März 2023

Version	Versionshinweise
---------	------------------

Version	Versionshinweise
<ul style="list-style-type: none">• Semantische Version: 1.4.0• Version: 476	<p>Neue Features</p> <ul style="list-style-type: none">• Dem <code>FileStorage</code> Objekt wurde eine neue <code>uploadFiles</code> Methode hinzugefügt, um mehrere Dateien hochzuladen.• Die Viewer-UI-Komponente unterstützt jetzt Drag & Drop, um das Hochladen von Dateien zu starten.• Der WebCodecs Browser API wird jetzt auch für Audio und Webcam verwendet. <p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">• Speicherlecks im Zusammenhang mit wiederholten Verbindungen von derselben Seite wurden behoben.• <code>setUploadBandwidth</code> erlaubt jetzt Werte bis zu 1 Gbit/s.• Optimiertes Rendern von UI-Komponenten.• Die Unterstützung für animierte Cursor unter Windows wurde behoben.• Es wurde ein Problem mit der Unterstützung der Zwischenablage behoben, wenn sowohl Text- als auch Bilddaten für denselben Vorgang vorhanden waren.• Verbesserte Robustheit der WebcamAPI: Die Einstellungen können nicht geändert werden,

Version	Versionshinweise
	<p>solange eine Anfrage bereits bearbeitet wird. Sie verfolgt <code>webcam.setEnabled</code> jetzt die Geräte-ID für die Anfrage und gibt ein Promise zurück. Die Viewer-UI-Komponente zeigt im Fehlerfall eine Benachrichtigung an.</p>

1.3.1 — 9. Dezember 2022

Version	Versionshinweise
<ul style="list-style-type: none"> • Semantische Version: 1.3.1 • Version: 413 	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none"> • Es wurde ein Problem behoben, das dazu führen konnte, dass die Benutzeroberfläche für die Zeitzonenumleitung nicht mehr mit dem Server synchronisiert war. • Ein Speicherleck nach mehreren erneuten Verbindungen wurde behoben. • Es wurde ein Problem behoben, das beim Trennen der Verbindung zu einer leeren Seite führte. • Es wurde ein Fehler behoben, der zu Konsolenwarnungen beim Schließen des Audiodecoders führte.

1.3.0 — 11. November 2022

Version	Versionshinweise

Version	Versionshinweise
<ul style="list-style-type: none"> • Semantische Version: 1.3.0 • Version: 407 	<p>Neue Features</p> <ul style="list-style-type: none"> • Adoptierte Cloudscape (https://cloudscape.design) for the UI Viewer component. • Unterstützung für die Zeitzonenumleitung hinzugefügt. <p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none"> • Fehlende Aktualisierung in der asynchronen Zwischenablage behoben, wenn der DCV Viewer fokussiert war. • Die <code>setDisplayScale</code> Funktion wird nicht mehr benötigt, wenn die Anzeige auf der Client-Seite skaliert wird. • Die <code>DCVViewer</code> Komponente wird jetzt automatisch aufgerufen, <code>disconnect()</code> wenn sie nicht gemountet ist.

1.2.1 — 21. Juli 2022

Version	Versionshinweise
<ul style="list-style-type: none"> • Semantische Version: 1.2.1 • Version: 358 	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none"> • Es wurde ein Problem behoben, das dazu führte, dass keine Verbindung zum DCV

Version	Versionshinweise
	Amazon-Server 2019.1 und älter hergestellt werden konnte.

1.2.0 — 29. Juni 2022

Version	Versionshinweise
<ul style="list-style-type: none">Semantische Version: 1.2.0Version: 352	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">Ein Absturzfehler, wenn die empfangenen Frames größer als die maximal unterstützte Auflösung (4096x2160) waren, wurde behoben.Ressourcenobjekte (die als Argumente an <code>fileDownload</code> und <code>filePrinted</code> Beobachter übergeben werden) verfügen jetzt über die <code>decline</code> Methoden <code>accept</code> und, die für das Objekt aufgerufen werden können, um die Ressource herunterzuladen bzw. zu verwerfen.Kleinerer Fehler bei der automatischen Synchronisation der Zwischenablage beim Trennen der Verbindung behoben.

1.1.3 — 23. Mai 2022

Version	Versionshinweise
<ul style="list-style-type: none">Semantische Version: 1.1.3	Änderungen und Fehlerbehebungen

Version	Versionshinweise
<ul style="list-style-type: none"> Version: 329 	<ul style="list-style-type: none"> Es wurde ein Problem behoben, das eine erfolgreiche Verbindung bei der Angabe der <code>web-url-path</code> Option verhinderte.

1.1.2 — 19. Mai 2022

Version	Versionshinweise
<ul style="list-style-type: none"> Semantische Version: 1.1.2 Version: 322 	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none"> Es wurde ein Problem behoben, das dazu führen konnte, dass die Eingabe nach der Verbindung nicht richtig funktionierte. Die Mauskoordinaten wurden behoben, wenn das Skalierungsverhältnis größer als 1 ist.

1.1.1 — 23. März 2022

Version	Versionshinweise
<ul style="list-style-type: none"> Semantische Version: 1.1.1 Version: 309 	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none"> Melden <code>Transport Error</code> Sie, wenn die Kommunikation mit dem Server unterbrochen wird. Ein wiederkehrender Dekodierungsfehler beim Streamen großer Auflösungen wurde behoben.

1.1.0 — 23. Februar 2022

Version	Versionshinweise
<ul style="list-style-type: none">• Semantische Version: 1.1.0• Version: 295	<p data-bbox="829 380 1040 411">Neue Features</p> <ul style="list-style-type: none">• Veröffentlichen Sie die Amazon DCV Web SDK UI-Bibliothek mit der <code>DCVViewer</code> React-Komponente.• Exportieren Sie Amazon DCV Web Client SDK sowohl als als auch als UMD ES-Module.• Unterstützung für hohe Farbgenauigkeit hinzugefügt.• Es wurde die Möglichkeit hinzugefügt, Clients, die mit einer Sitzung verbunden sind, aufzulisten und mit ihnen zu interagieren. Es wurden Benachrichtigungen für Verbindungen und Verbindungsabbrüche hinzugefügt. <p data-bbox="829 1329 1341 1360">Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">• Verbesserte Unterstützung für die Dekodierung von Webcodecs.• Verschiedene Verbesserungen der Tastatur.• Behebung eines Fehlers, der das Öffnen eines zweiten Bildschirms verhinderte, wenn die Zwischenablage deaktiviert war.

1.0.4 — 20. Dezember 2021

Version	Versionshinweise
<ul style="list-style-type: none">• Semantische Version: 1.0.4• Version: 249	<p>Neue Features</p> <ul style="list-style-type: none">• Support das Öffnen mehrerer Verbindungen von derselben Seite aus.• Support das Laden SDK von CDN a.

1.0.3 — 01. September 2021

Version	Versionshinweise
<ul style="list-style-type: none">• Semantische Version: 1.0.3• Version: 202	<p>Neue Features</p> <ul style="list-style-type: none">• Experimentelle Unterstützung für WebCodecs. Dies ist standardmäßig deaktiviert und muss über das <code>ConnectionConfig</code> Objekt aktiviert werden, das die neue Eigenschaft <code>verwendetenableWebCodecs</code> .• Zwischenablage: Unterstützung für den <code>image/png</code> Datentyp in Chromium-basierten Browsern hinzugefügt.• Observer/Callback hinzugefügt, um den Screenshot des Servers als PNG Bild zu erhalten (erfordert DCV Amazon-Server 2021.2).

Version	Versionshinweise
	<p>Änderungen und Fehlerbehebungen</p> <ul style="list-style-type: none">• Die Handhabung von Tastaturmodifikatoren wurde verbessert.

1.0.2 — 30. Juli 2021

Version	Versionshinweise
<ul style="list-style-type: none">• Semantische Version: 1.0.2• Version: 167	<ul style="list-style-type: none">• Feste Druckererkennung für Eingabestifte.• Verbesserte Unterstützung für das koreanische Tastaturlayout in Chrome.

1.0.1 — 31. Mai 2021

Version	Versionshinweise
<ul style="list-style-type: none">• Semantische Version: 1.0.1• Version: 141	<ul style="list-style-type: none">• Die Ausbreitung von Verbindungsfehlern und Schließungsgründen wurde behoben• Aktualisierung des Dateispeicher-Chunk-Fortschritts behoben• Verbesserte Webcam-Handhabung• Verbesserte Audioeingangsverarbeitung

1.0.0 — 24. März 2021

Version	Versionshinweise
<ul style="list-style-type: none"> Semantische Version: 1.0.0 Version: 81 	Erste Version des Amazon DCV Web ClientSDK.

Dokumentverlauf

In der folgenden Tabelle wird die Dokumentation für diese Version von Amazon DCV Web Client beschriebenSDK.

Änderung	Beschreibung	Datum
Amazon DCV Web Client versie SDK 1.8.4	Amazon DCV Web Client SDK 1.8.4 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.8.4.	1. Oktober 2024
Amazon DCV Web Client, SDK Version 1.5.6	Amazon DCV Web Client SDK 1.5.6 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.5.6.	9. November 2023
Amazon DCV Web Client, SDK Version 1.4.4	Amazon DCV Web Client SDK 1.4.4 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.4.4.	29. Juni 2023
Amazon DCV Web Client versie SDK 1.4.0	Amazon DCV Web Client SDK 1.4.0 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.4.0.	28. März 2023

Änderung	Beschreibung	Datum
Amazon DCV Web Client, SDK Version 1.3.1	Amazon DCV Web Client SDK 1.3.1 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.3.1 .	9. Dezember 2022
Amazon DCV Web Client, SDK Version 1.3.0	Amazon DCV Web Client SDK 1.3.0 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.3.0 .	11. November 2022
Amazon DCV Web Client, SDK Version 1.2.0	Amazon DCV Web Client SDK 1.2.0 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.2.0 .	29. Juni 2022
Amazon DCV Web Client versie SDK 1.1.0	Amazon DCV Web Client SDK 1.1.0 ist jetzt verfügbar. Weitere Informationen finden Sie unter SDKv.1.1.0 .	23. Februar 2022
Amazon DCV Web Client, SDK Version 1.0.1	Einige Tippfehler wurden behoben. Kleinere Fehler wurden behoben, siehe SDKv.1.0.1 .	31. Mai 2021
Erstversion	Erstveröffentlichung dieses Inhalts.	24. März 2021

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.