

IDS NXT Kommunikation

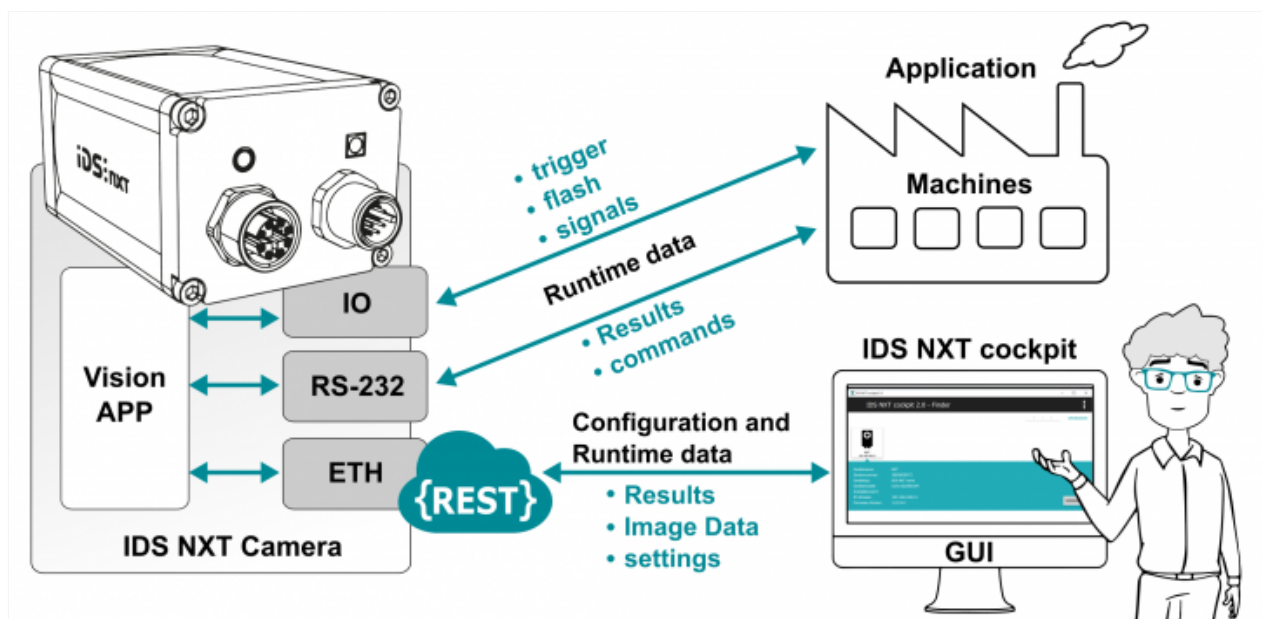
IDS NXT Kameras in die eigene Fertigungsautomation integrieren

Mit IDS NXT haben wir eine Plattform für eine neue Generation von Vision Systemen für industrielle Anwendungen geschaffen, die beispielsweise als leistungsstarke Vision Sensoren in der Fertigungsautomation eingesetzt werden können. In Verbindung mit IDS NXT ocean ist der Umgang mit Deep Learning zudem besonders einfach. Die All-in-One-Lösung bringt alles mit, um IDS NXT Kameras in leistungsstarke Inferenzkameras zu verwandeln, indem individuelle neuronale Netze direkt on-the-edge ausgeführt werden. Von der Bildaufnahme und Verwaltung über das Training der KI bis zur fertig arbeitenden Inferenzkamera sind es nur wenige Schritte. Doch über welche Schnittstellen wird die Inferenzkamera in die eigene Umgebung eingebunden?

i In diesem Techtipp gehen wir insbesondere auf die Kommunikationsmöglichkeiten von IDS NXT Kameras ein und zeigen nachvollziehbar, wie Sie Inferenzergebnisse in Ihrer eigenen Umgebung nutzen können.

IDS NXT Schnittstellen

Als kompakte Embedded-Vision-Systeme erledigen IDS NXT Kameras Bildverarbeitungsaufgaben direkt on-the-edge und können Ergebnisse ohne zusätzliche PCs bereitstellen. Folglich können sie als autonom arbeitende Vision-Sensoren eingesetzt werden. Was die Anforderungen an den Zugriff und Datenaustausch für externe Systeme angeht, unterscheiden sie sich grundlegend von Standard Industriekameras, die hauptsächlich als Bildlieferanten für PC-basierte Bildverarbeitung ihren Dienst tun. Daher haben wir unsere IDS NXT Kameras mit Schnittstellen ausgestattet, die in IoT-artigen Einsatzszenarien einfacher und flexibler einsetzbar sind.



i **IDS NXT cockpit**
Das IDS NXT cockpit bietet Zugriff auf alle wichtigen Einstellungen und Funktionen der IDS NXT Industriekameras und setzt dabei vollständig



auf deren REST-Interface auf. Zusätzlich können Sie darüber Vision Apps installieren oder konfigurieren. Egal über welche Schnittstelle Sie später mit der Kamera arbeiten – mit dem IDS NXT cockpit steht Ihnen eine **einfach bedienbare GUI** zur Verfügung, um Kameraanwendungen komfortabel **einzurichten** und für den Runtime-Betrieb zu **konfigurieren**.

IO und RS-232

Auch im Zeitalter von Industrie 4.0 bieten digitale Ein-/Ausgänge und UART-basierte serielle Schnittstellen mit den darauf aufbauenden Feldbussen zahlreiche Vorteile gegenüber ihren moderneren Nachfolgern. Auch für komplexe Teileprüfungen und bildbasierte Kontrollen genügen meist die vermeintlich einfachen Ergebnisse IO bzw. NIO, um Produkte in großen und modernen Industriestraßen schnell und unkompliziert zu lenken. Kaum eine andere Schnittstelle lässt sich so einfach und preiswert implementieren wie digitale In- bzw. Outputs. Ähnlich verhält es sich mit der seriellen Gerätekommunikation mit RS-232. Sie wird von so vielen Herstellern und Anwendern unterstützt und für keine andere existiert ein umfangreicheres Angebot an bewährtem Zubehör. Wodurch RS-232 auch heute noch zu den Standards in der Industriellen Kommunikation zählt.



Mit IO- und RS-232 Schnittstellen ergänzen IDS NXT Kameras viele Bestandsanlagen mit kompakten und leistungsfähigen Embedded-Vision-Lösungen, ohne die bestehende Infrastruktur ändern zu müssen.

IDS NXT REST Schnittstelle



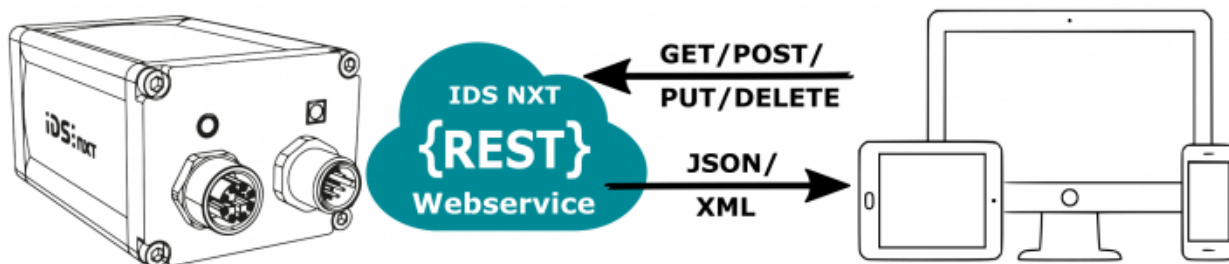
Mit einem REST (Representational State Transfer) Webservice bieten IDS NXT Kameras eine standardisierte offene Schnittstelle zur Maschine-zu-Maschine-Kommunikation, die eine einfache Vernetzung erlaubt. Darüber können Parameter und Ergebnisse der Kameras verändert bzw. abgefragt werden. Der REST Webservice basiert auf dem HTTP Protokoll und ist damit in den meisten Netzwerken verfügbar.

Als einheitliche Schnittstelle werden die Standard-HTTP-Methoden wie GET, POST, PUT, PATCH, usw. verwendet. Damit ist die IDS NXT REST Schnittstelle nicht nur plattformunabhängig, sondern durch die weite Verbreitung der HTTP-Methoden auch sehr vielseitig in unterschiedlichsten Clients (PCs, Smartphones, Tablets) nutzbar.

Während man für C++ Anwendungen eine komplett eingerichtete Entwicklungsumgebung benötigt, können REST(ful)-APIs von webfähigen Geräteklassen ohne zusätzliche Software oder Gateways bedient werden. Dies erleichtert nicht nur den Einstieg, sondern ermöglicht es auch, umfangreiche Setup-Aktionen in simple Skripten zu verpacken. Für die generische Kommunikation mit autonom arbeitenden Vision-Sensoren kann REST deshalb in der Welt des IoT (Internet der Dinge) ebenso wie im IIoT (Industrial Internet of Things) einfacher und mit weniger Overhead eingesetzt werden als eine PC-basierte Schnittstelle, wie beispielsweise GenlCam. Für IDS NXT Kameras ist REST dabei nicht nur die "Runtime"-Schnittstelle zur Daten- und Ergebnisübertragung, sondern kommt auch zur Konfiguration (z.B. IDS NXT cockpit) der Kamera, ihrer Apps und Schnittstellen (IO, RS232) zum Einsatz.

Warum REST?

- **Zustandslos**
Durch die zustandslose Kommunikation ist jede Client-Server Nachricht in sich abgeschlossen und enthält alle notwendigen Informationen über den aktuellen Anwendungszustand. Dies bedeutet, dass der REST-Service zwischen den Ausführungen nichts speichern muss. Eine Sitzungsverwaltung im Server ist damit nicht notwendig, wodurch der Webservice beliebig im Netzwerk skaliert werden kann. Damit ist REST ideal im IoT einsetzbar.
- **Einheitlich und plattformunabhängig**
Mit REST bieten Systeme eine standardisierte offene Schnittstelle basierend auf Webtechnologie, die eine einfache Vernetzung erlaubt. Daten und Ergebnisse können ohne Protokollkonvertierung oder Gateways zwischen allen (!)IoT-Geräten ausgetauscht werden.
- **Einfach und Ressourcenschonend**
Aufbau, Übermittlung und gegebenenfalls die Verarbeitung von Ergebnissen einfacher REST-API-Aufrufe, benötigen in den meisten Fällen nur wenig Rechenkapazität. Die notwendigen technischen Voraussetzungen für die REST-Kommunikation sind durch die weite Verbreitung der Standard-HTTP-Methoden zudem auf den meisten webfähigen Geräten bereits ohne zusätzliche Software nutzbar.
- **Ortsunabhängig**
Clients und Server können getrennt voneinander betrieben werden ("verteiltes System") . Auch wenn diese durch verschiedene Server gehostet werden, ist die Kommunikation über REST dennoch möglich.
- **Industrie- und Web-kompatibel**
Leichter Zugang und Transfer von Daten ist ein wesentlicher Bestandteil des IIoT. Mit ihrer Flexibilität und der Fähigkeit, weit verfügbare Ressourcen zu nutzen, kann REST auch die industrielle Automatisierung vereinfachen. Viele Systeme und Geräte können durch die anwendungs- und geräteübergreifende Kommunikation schnell und einfach integriert werden. Die Web-Kompatibilität ist ein Tür-Öffner für eine umfassendere Integrationsstrategie. Von der Offenheit für neue Systeme und Technologien profitiert auch die Evolution und Entwicklung der Industrie.



Der IDS NXT REST Webservice ist mit folgenden Eigenschaften implementiert:

- Der REST Webservice ist während des regulären Betriebs immer aktiv.
- Die Zugriffssicherheit ist über die HTTP-Basic-Authentifizierung umgesetzt. D.h. mit jeder Anfrage sind zur Authentifizierung die Benutzerdaten eines berechtigten Benutzers anzugeben, die dann Base64-encodiert übertragen werden.
- Die Antworten werden im JSON- oder XML-Format übertragen.



Dokumentation

Die **Dokumentation der IDS NXT REST Schnittstelle** und Informationen zur Anmeldung an IDS NXT Kameras (**Handbuch zum IDS NXT cockpit, Kapitel "Am Gerät anmelden"**) finden Sie im [Downloadbereich der IDS Webseite](#) zu Ihrem IDS NXT Kameramodell.

IDS NXT Kommunikation über I/Os und RS-232



Um Bildverarbeitungsergebnisse der Kamera über die digitalen Ein-/Ausgänge bzw. die serielle Schnittstelle zu kommunizieren, müssen Sie lediglich die beiden IDS NXT Vision Apps "GPIO Gateway" und "RS-232 Gateway" über das IDS NXT cockpit starten und einstellen, von welcher Vision App Sie spezifische Ergebnisse ausgeben möchten. Dazu müssen Sie nichts programmieren. Alle Einstellmöglichkeiten können über die GUI vorgenommen werden.

Als Beispiel können Sie die digitalen Ausgänge so konfigurieren, dass sie jeweils für eine bestimmte Zeit aktiv sind, wenn die Kamera KI (CNN manager mit "ImageNet1000") bestimmte Objekte klassifiziert hat (z.B. "Gut" oder "Schlecht"). Oder Sie kommunizieren die erkannte Objektklasse und die Wahrscheinlichkeit der Inferenz als Ergebniswerte über die serielle Verbindung.



Dokumentation

Eine detaillierte Beschreibung der beiden Vision Apps und ihre Konfiguration finden Sie im Downloadbereich zu Ihrer IDS NXT Kamera auf der IDS Webseite.

↓ [GPIO Gateway" Dokumentation \(DE\)](#)

↓ [RS-232 Gateway" Dokumentation \(DE\)](#)

IDS NXT REST Kommunikation

Der große Vorteil von REST liegt für Entwickler in der Web-Fähigkeit und damit in der Einfachheit der benötigten Tools. Deshalb zeigen wir Beispiele, wie Sie die HTTP-Methoden GET, POST, PUT und DELETE mit einfachen Mitteln nutzen können, um Daten einer IDS NXT Kamera abzurufen.

ⓘ Bitte beachten Sie, dass dieser Techtipp nicht als umfassende Dokumentation für die Verwendung der REST API gedacht ist. Vielmehr möchten wir grundlegende Hinweise geben, um Ihnen den Einstieg zu erleichtern. Der "Rest" liegt bei Ihnen!

Damit Sie die folgenden Beispiele sinnvoll nachvollziehen zu können, benötigen Sie eine IDS NXT Kamera mit Zugriffsmöglichkeit über einen Windows PC im selben Netzwerk. Um nicht nur stupide Parameter abzufragen, lassen wir die Kamera auch etwas Sinnvolles tun. Wir nutzen die Kamera-KI und das vorinstallierte CNN "KritzelINN". Die Kamera-KI erkennt damit Zeichnungen von Personen, Hunden, Katzen und Kameras. Wir fragen über die REST-API, was sie auf einer selbst gemalten Zeichnung erkennt.

TestszENARIO "Ich sehe was, was du nicht siehst":

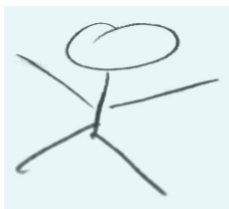
- Netzwerk mit IDS NXT rio oder rome Kamera und einem Windows PC
- Vision App "CNN-Manager" läuft mit dem vorinstallierten neuronalen Netz "KritzelINN".
- Verschiedene Zeichnungen von einer Person, einer Kamera, einer Katze oder einem Hund.

ⓘ **Dokumentation**
 Eine Dokumentation zur Verwendung des CNN-Manager inkl. der benötigten REST-URLs und wie Sie das "KritzelNet" starten, finden Sie im Downloadbereich zu Ihrer IDS NXT Kamera auf der IDS Webseite.
[↓ "CNN manager" Dokumentation \(DE\)](#)

REST-Aufruf mit "cURL" (Kommandozeile)

cURL ist ein Kommandozeilen-Tool zur Übertragung von Daten über URLs, das Sie einfach in der Eingabeaufforderung (auch cmd.exe) ausführen können. Um das Inferenzergebnis (KritzelINN) des letzten Kamerabildes abzufragen, senden wir einen 'GET' Request mit der gewünschte Ressource `./vapps/cnnmanager/resultsources/last`.

Kamerabild:



```
C:\Users\ids>curl -sX GET http://192.168.2.102/vapps/cnnmanager/resultsources/last --user admin:ids
{
  "inference": {
    "Top1": "person",
    "Top2": "camera",
    "Top3": "cat",
    "Top4": "dog"
  },
  "inference_propability": {
    "Top1": "0.99",
    "Top2": "0.01",
    "Top3": "0.00",
    "Top4": "0.00"
  },
  "inferencetime": {
    "Content": "43"
  }
}
```

Wenn Sie die **JSON Antwortdaten** filtern möchten, können Sie ein Zusatztool wie **.jq** verwenden. Der leichte und flexible Kommandozeilen-JSON-Prozessor benötigt keine sonstigen Abhängigkeiten. Um damit lediglich das **Top1 Inferenzergebnis** und seine Wahrscheinlichkeit auszugeben, erweitern Sie den Request um:

```
| jq ".inference.Top1, .inference_propability.Top1, .inferencetime.Content"
```

```
C:\Users\ids>curl -sX GET http://192.168.2.102/vapps/cnnmanager/resultsources/last --user admin:ids | jq ".inference.Top1,
.inference_propability.Top1"
"person"
"0.95"
```

Legen Sie nun zum Testen verschiedene Zeichnungen vor die Kameralinse und rufen Sie das jeweilige Inferenzresultate ab.

REST-Aufruf im Browser

Direkt über die Adresszeile

Die Inferenzabfrage lässt sich auch sehr einfach über Web-Browser abfragen (z.B. Firefox oder Google Chrome). Geben Sie dazu folgende URL in die Adresszeile ein:

<http://admin:ids@192.168.188.21/vapps/cnnmanager/resultsources/last>

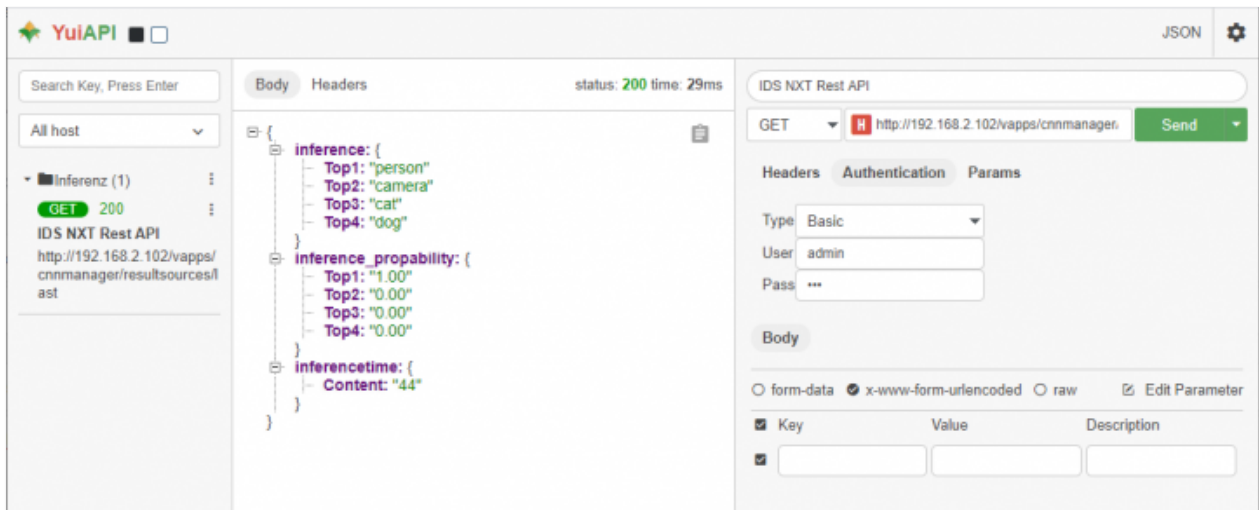
Die Anmeldeinformationen werden im Standard-HTTP Authorization Header übertragen.

⚠ Leider unterstützen aus Sicherheitsgründen nicht mehr alle Browser (z.B. IE) eingebettete Anmeldeinformationen in der URL!

Mit Browser Plugins

✓ Rest API Browser Plugins
Es gibt ebenfalls eine Reihe sehr guter **Browser Plugins**, die Sie beim Testen von REST Aufrufen mit sehr übersichtlichen GUIs unterstützen.

Wir testen unsere Abfrage des Inferenzresultats mit dem Google Chrome Browser und der Erweiterung "[YuiAPI](#)".



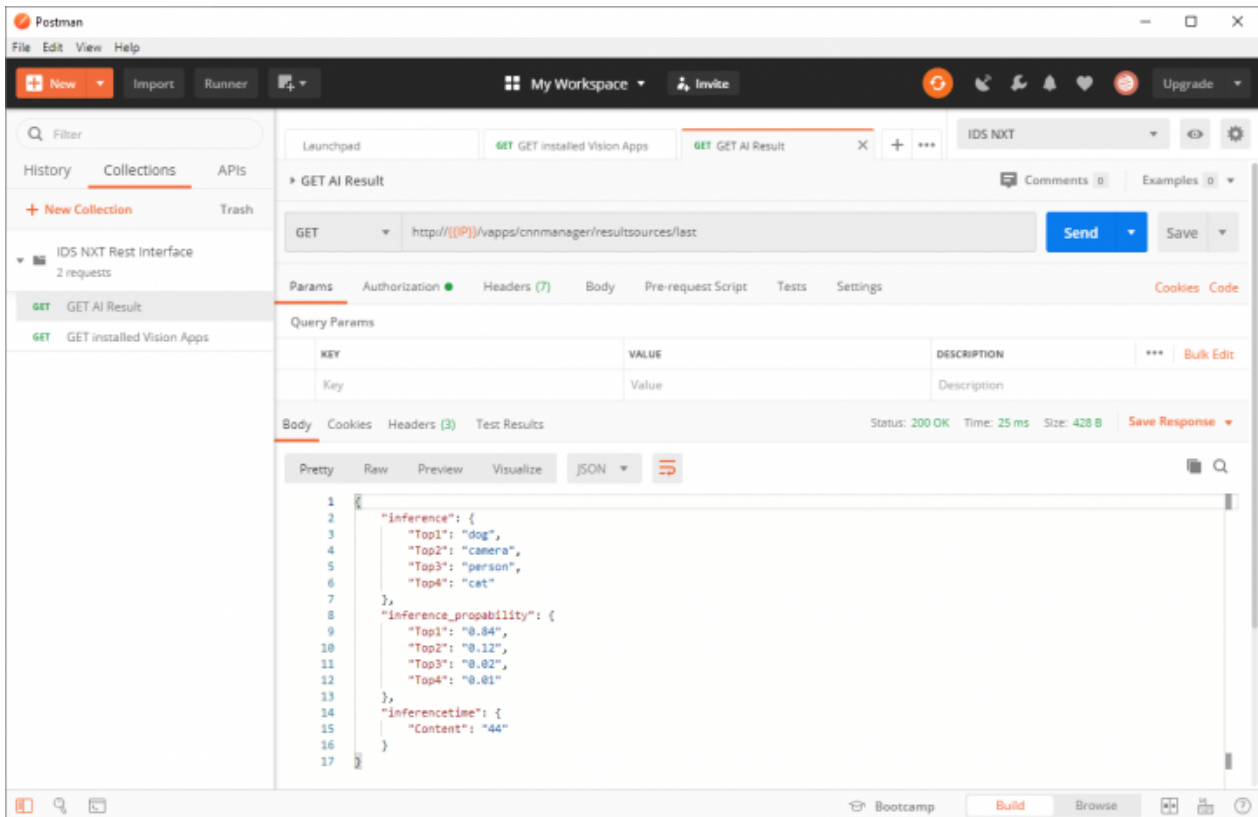
Komfortable GUI-Oberfläche für die Rest Kommunikation direkt im Browser

REST-Aufruf in speziellem Client

Die wohl komfortabelste Möglichkeit zur Kommunikation mit einem REST Webservice sind Clients wie beispielsweise "[Postman](#)". Mit der übersichtlichen GUI und viel Unterstützung durch Statuscodes, Antwortzeiten, einer automatischen Spracherkennung sowie Link- und Syntax-Highlighting beim Erstellen von REST Aufrufen finden auch Laien schnell einen Zugang zu REST. Auch die Analyse und Filterung von Antwortdaten wird damit vereinfacht. Ein weiterer

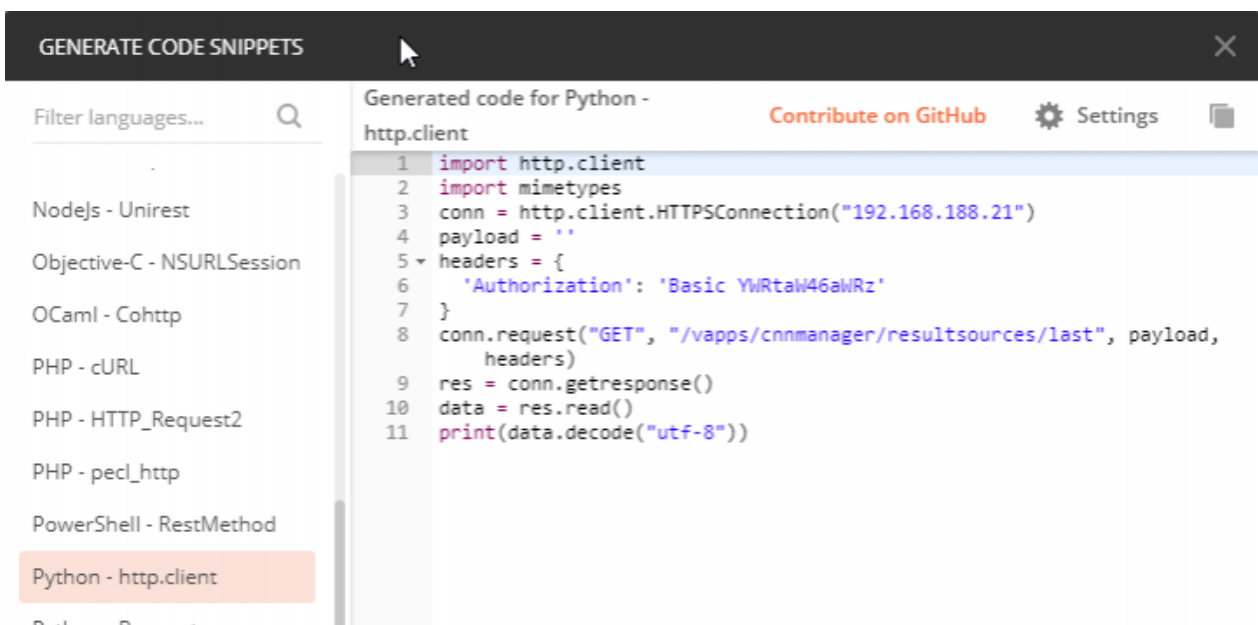
Vorteil liegt in der Möglichkeit, vollständige API-Beschreibungen zu importieren bzw. selbst zu erstellen, indem eine der bekannten API Spezifikation wie beispielsweise **RAML**, **WADL** oder **OpenAPI** verwendet wird.

Postman ist eine **visuelle Entwicklungsumgebung** für die Kommunikation mit dem REST Webservice von IDS NXT Kameras. REST Aufrufe lassen sich damit nicht nur erstellen, sondern auch in Kollektionen abspeichern und wieder verwenden.



Postman ist eine Entwicklungs- und Testumgebung für Rest APIs

Daten, die Sie immer wieder verwenden, können Sie damit komfortabel in **Variablen** bereitstellen. Aber die wohl wichtigste Funktion von Postman, ist die **Code-Erstellung**. Per Knopfdruck erstellt Ihnen der Client aus Ihren getesteten und gespeicherten REST Aufrufen vollständig nutzbare Code-Snippets für viele gängige Programmiersprachen und andere Clients, wie beispielsweise cURL, C, C#, Python, Java oder Swift.



✓ Die Code-Snippets aus Postman stellen Ihnen eine gute Basis zur Verfügung, um IDS NXT Kameras in selbst programmierte Anwendungen zu integrieren.

REST API Integration in programmierte Anwendungen

Obwohl Ihnen für IDS NXT Kamera kein Entwicklungs-SDK für beispielsweise C++ Anwendungen bereitgestellt wird, bedeutet das nicht, dass eine programmatische Integration der Kamera nicht möglich ist. Die Verwendung von REST- bzw. HTTP-Verbindungen ist in den meisten Programmiersprachen ohne Zusatzsoftware vom Hersteller möglich. Deshalb ist es für einen REST-Client wie Postman überhaupt möglich, Ihnen vollständig funktionsfähige Code-Snippets bereitzustellen, die Sie direkt verwenden können.


Am Beispiel unseres KI-Kamera-Ergebnisses zeigen wir Ihnen hier exemplarisch für Python, wie Sie den REST-Aufruf programmieren können.

REST-Aufruf mit Python

```
import http.client
import mimetypes
conn = http.client.HTTPSConnection("192.168.188.21")
payload = ''
headers = {
    'Authorization': 'Basic YWRtaW46aWRz'
}
conn.request("GET", "/vapps/cnnmanager/resultsources/last", payload, headers)
res = conn.getresponse()
data = res.read()
# output camera response
print(data.decode("utf-8"))
```

✓ Python stellt eine sehr gebräuchliche Programmiersprache im IoT dar. Wir möchten Ihnen deshalb die Integration unserer IDS NXT Kameras in eine Fertigungsautomation damit etwas näher erklären.

Interaktive Entwicklung mit Python

 Mit dem Techtipp stellen wir Ihnen weitere Python Code-Snippets mit einem Softwarepaket zur Verfügung, die den grundlegenden Umgang mit IDS NXT Kameras und ihren Schnittstellen zeigen. Mit den enthaltenen Jupyter-Notebooks können Sie in einer [Open-Source-Webanwendung](#) die Python Beispiele und den Datenaustausch mit Ihrer IDS NXT Kamera live testen und nachvollziehen.

1. Log-In und Bildübertragung (*connect_and_get_image.ipynb*)

Im ersten Beispiel erfahren Sie, wie der GET-Befehl verwendet wird, um sich zu autorisieren und ein Bild von Ihrer IDS NXT Kamera abzurufen.

2. Belichtungszeit ändern (*change-exposure.ipynb*)

Das zweite Beispiel zeigt, wie Kameraparameter abgefragt und modifiziert. Dazu verändern wir den Belichtungswert mit Hilfe des PATCH-Befehls.

3. Bildaufnahme triggern (*trigger-image-acquisition.ipynb*)

Mit Hilfe des POST-Befehls zeigen wir im dritten Beispiel, wie Sie eine Bildaufnahme auslösen, um den Bildinhalt einer veränderten Szene abzurufen.

4. IO-Gateway konfigurieren (*set_gpio_gateway.ipynb*)

Im vierten Beispiel zeigen wir Ihnen, wie Sie die digitalen Ausgänge der Kamera abhängig von den Ergebnissen der Vision App CNN manager schalten.



Python Code-Beispiele

↓ [Download der Python Code-Beispiele als Jupyter-Notebooks](#)

Wie Sie die Jupyter-Notebooks verwenden und die Beispiele testen können, wird im [Anhang: "Talking to the IDS NXT with Python"](#) erklärt.

Zusammenfassung

Mit digitalen Ein- und Ausgängen, RS-232 und einem REST Webservice stellen IDS NXT Kameras Standardschnittstellen zur Verfügung, mit denen sie sehr flexibel einsetzbar sind. Embedded Vision Geräte, die Bildverarbeitungsaufgaben selbständig ohne einen Host-PC erledigen, benötigen damit auch für die Kommunikation mit Gegenstellen keine PC-Unterstützung. Das zustandslose REST-Interface ermöglicht die Verbindung zu allen Web-kompatiblen Geräten, wodurch die Kameras für eine IoT-Netzwerkstruktur optimal aufgestellt sind. Mit geeigneten REST-Clients (wie z.B. Postman) lassen sich zudem Software-Wrappers für viele gängige Programmiersprachen per Knopfdruck erstellen.

Mit den vorhandenen Tools und Schnittstellen der IDS NXT Inferenzkamera Plattform ist die Integration in Bestandsanlagen mit bewährter Technik genauso einfach möglich, wie in zukünftige Anlagen mit web-kompatibler IoT-Struktur. Für weitere Informationen über IDS NXT oder deren Integration, besuchen Sie die [Knowledge Base](#) unserer Webseite oder setzen Sie sich mit unserer [Systemberatung in Verbindung](#).



Tipp: IDS NXT Webinar-Video

In unserem Webinar-Video "[Integration von IDS NXT in die Fertigungsautomation](#)" wird die Kommunikation über die IDS NXT Kameraschnittstellen mit Hilfe der Python Beispielcodes und den Jupiter-Notebooks demonstriert.



Weiterlesen Tipp: IDS NXT Kameras mittels OPC UA in die Fertigungsautomation integrieren

In der Fortsetzung "[IDS NXT OPC UA Kommunikation](#)" gehen wir insbesondere auf die neuen Kommunikationsmöglichkeiten von IDS NXT Kameras durch ihren integrierten OPC UA Server ein. Anhand eines KI-Beispielworkflows zeigen wir, wie Sie über einen OPC UA Client gespeicherte Kameraeinstellungen aktivieren, eine Inferenzaufgabe starten und Ergebnisse abfragen können. Damit integrieren Sie IDS NXT Kameras noch einfacher in OPC UA basierte Industrieanlagen.