

# **ATCS Software Interface für die Ansteuerung von DFI- Anzeigen**

---



## **Technische Beschreibung**

Druckdatum 17.12.2020

**Rechteinhaber: © ATRON electronic GmbH 1997-2020**

**Gestaltung: ATRON electronic GmbH**

Diese Dokumentation einschließlich ihrer Teile ist urheberrechtlich geschützt. Jede Verwertung bzw. Nutzung ist ohne die Zustimmung der ATRON electronic GmbH unzulässig und wird rechtlich verfolgt. Dies gilt insbesondere für die elektronische oder sonstige Vervielfältigung, Übersetzung, Verbreitung und öffentliche Zugänglichmachung. ATRON, DEBAS, ATRIES und andere in der Dokumentation genannte Warenprodukte sind geschützte Warenzeichen der ATRON electronic GmbH. Andere genannte Marken und Produkte unterliegen dem Warenzeichenschutz des jeweiligen Herstellers.

## Inhalt

<b>1</b>	<b>Zu diesem Dokument.....</b>	<b>3</b>
<b>1.1</b>	<b>Alternativer Zugriff auf die Schnittstelle .....</b>	<b>3</b>
<b>2</b>	<b>Schnittstelle.....</b>	<b>4</b>
<b>2.1</b>	<b>Backend Schnittstellenversion .....</b>	<b>4</b>
<b>2.2</b>	<b>IP Adresse und Port des Displays .....</b>	<b>4</b>
<b>2.3</b>	<b>Schnittstellen-Daten .....</b>	<b>4</b>

# **1 Zu diesem Dokument**

Dieses Dokument beschreibt die Software-Schnittstelle zwischen dem Backend (RBL Zentrale) und dem Frontend (Display Controller).

Bei direkter Kommunikation der Anzeiger Einheiten mit dem ATRON ITCS empfehlen wir die vorgängige Abstimmung auf IT Ebene in Bezug auf die VPN / Firewall Einstellungen.

## **1.1 Alternativer Zugriff auf die Schnittstelle**

Wir empfehlen bei Anschluss von Drittsystemen und wenn keine Möglichkeit zur direkten Kommunikation mit dem entsprechenden Anzeiger möglich oder gewollt ist, die Nutzung der Endpunkte „Trips“, „Specialtext“ und „Status“.

Die Daten können in Polling Zyklen beliebig oft abgefragt werden, wobei wir für die Trips ein dynamisches Polling, abhängig von der Prognose der nächsten Abfahrt vorstellen können.

Das Schreiben von Statusinformationen sollte spätestens alle 60 Sekunden erfolgen, damit die Status Anzeige im ITCS System korrekt anzeigen kann.

## 2 Schnittstelle

Das Backend und Frontend kommunizieren über http-get und http-post.

Http-get kann Argumente in der uri haben. Die Antwort sind Daten im JSON-Format.

Http-post enthält JSON-Daten und wird in den Antwortdaten mit JSON beantwortet.

Bei mehreren Daten ist es möglich, dass das Frontend die Daten per http-get aus dem Backend bezieht oder dass das Backend diese Daten per http-post an das Frontend sendet. In diesem Fall wird dasselbe JSON-Format verwendet.

### 2.1 Backend Schnittstellenversion

Die URL enthält die Schnittstellenversion. Wenn die Version vom Server nicht unterstützt wird, wird der Fehlercode 500 zurückgegeben.

### 2.2 IP Adresse und Port des Displays

Das Backend verwaltet die Tabelle der Display-ID, IP-Adresse und Port. Die IP-Adresse wird aus den HTTP-Verbindungsinformationen gelesen. Da der Port von der HTTP-Verbindung nicht lesbar ist, wird er als Konfigurationswert festgelegt.

Beachten Sie, dass jede Anzeige denselben Port verwenden muss und dass die Weiterleitung über einen beliebigen Proxy-Server nicht unterstützt wird.

### 2.3 Schnittstellen-Daten

Die Daten sind im JSON-Format, UTF-8 encoded.

Alle Zeitangaben sind im Format: yyyy-mm-ddTHH:MM:SS+hh:mm.

#### Trips data

Description	<p>The trips data contains trips to be shown on the display the procedure to determine the valid data is:</p> <ol style="list-style-type: none"> <li>1. Display starts.</li> <li>2. Display removes all Data and fetches current data by http...getTripsData from backend.</li> <li>3. While time passes the backend sends updated TripsData and new TripsData by http... post ... TripsData.</li> <li>4. Display deletes old TripData (scheduled departure has passed (offline) or Trip was removed (online)).</li> <li>5. When display detects that it is offline. it shows the Trips according to scheduled departure for a while (parameter), then switches to "Offline screen".</li> <li>6. When display gets back connection, it will continue like 2. it <b>removes all data</b> and fetch current data.</li> </ol>
GetUrl	<p>http://&lt;backend-ip&gt;:&lt;backend-port&gt;/&lt;backend-web-dfu-url&gt;/&lt;api-version&gt;/trips/&lt;display-id&gt;</p> <p>for example: http://localhost:9999/webdfi/dpic/api/1/trips/150</p>
PostUrl	http://<display-ip>:<display-port>/display.fcgi/trips
JSON Data	{

<b>Example</b> (Answer for Get / Data in Post)	<pre> "validUntilTime" : "1970-01-01T01:00:00+01:00", "previewTime" : 0, "trips" : [   {     "tripKey" : "Key_0",     "icon" : "\${_img_2_.png;}",     "lineName" : "Linie_0",     "direction" : "Ziel_0",     "stopPosition" : "Gleis_0",     "viaHst" : [ "via stop 1", "via stop 2" ],     "scheduledDeparture" : "2016-10-17T15:24:16+02:00",     "expectedDeparture" : "2016-10-17T15:24:16+02:00",     "atDisPoint" : 0,     "isActive" : 0,     "edpiTimeFormat" : 0,     "specialTextTrip" : "special text for this trip.",     "specialTextType" : 0,     "specialTextStyle" : 0   },   {     "tripKey" : "Key_1",     "icon" : "\${bus.png;}",     "lineName" : "Linie_1",     "direction" : "Ziel_1",     "stopPosition" : "Gleis_1",     "viaHst" : [],     "scheduledDeparture" : "2016-10-17T15:25:56+02:00",     "expectedDeparture" : "2016-10-17T15:25:56+02:00",     "atDisPoint" : 0,     "isActive" : 0,     "edpiTimeFormat" : 0,     "specialTextTrip" : "",     "specialTextType" : 0,     "specialTextStyle" : 0   } ] </pre>
<b>JSON Post Response</b>	standard response see below
<b>JSON-Comments</b>	<p>icon: optional, symbol like bus or train, might be an extra column on the DFI</p> <p>previewTime: trips should be shown when they leave within previewTime minutes.</p> <p>validUntilTime: The data is valid until - data will not be used if no new data was received after validUntilTime</p> <p>atDisPoint:          Vehicle is at station possible Values: 0,1,2              0: far              1: near              2: at</p> <p>isActive:          0: not active (should be shown with absolute time HH:MM always - expectedDeparture and scheduledDeparture have the same value - not active trips will be removed from display when scheduled time has passed)</p>

	<p>1: active (will be shown)  2: active (will be shown)  3: hidden (will not be shown)</p> <p>edpiTimeFormat:  time format - how the departure time is displayed:  0: relative (up tp 60min); HH:MM (60min +)  1: relative (up tp 30min); HH:MM (30min +)  2: relative (always)  3: HH:MM  4: (not yet implemented HH:MM + fpa in minutes: 13:47 + 2')</p> <p>specialTextType:  0: special text will replace destination text - it will be shown in the field of destination text  1: special text will be concatenated to destination text - it will be shown in the field of destination text  2: alternate - special text and destination text will be shown in the field of destination text alternating  3: (not yet implemented - below special text will be placed below destination text in a new field this will inserted a new field - the next trip will be moved one line down)</p> <p>specialTextStyle:  0: nothing  1: blink  2: invert  3: scroll  4: change</p> <p>direction, specialTextTrip, icon ,lineName , stopPosition can contain a reference to an image</p> <p>previewTime:  in minutes - preview when trip will occur on the display</p>
--	--

### Special text message data

Description	
GetUrl	<p><code>http://&lt;backend-ip&gt;:&lt;backend-port&gt;/&lt;backend-web-dfu-url&gt;/&lt;api-version&gt;/specialTexts/&lt;display-id&gt;</code></p> <p>for example:  <code>http://localhost:9999/webdfi/dpic/api/1/specialTexts/150</code></p>
PostUrl	<code>http://&lt;display-ip&gt;:&lt;display-port&gt;/display.fcgi/specialTexts</code>
<b>JSON Data Example</b> (Answer for Get / Data in Post)	<pre>{   "specialTexts" : [     {       "specialText" : "This is the special text",       "specialTextStyle" : 0,       "specialTextLayout" : 1,       "validity" : {</pre>

	<pre>         "from" : "2016-10-17T15:51:41+02:00",         "to"   : "2016-10-17T15:53:21+02:00"       }     }   ] }</pre>
JSON Post Response	standard response
Comments	<p>There is only one special text message stored in the controller – there can not be stored several messages at the same time.</p> <p>SpecialTextStyle: see trip specialtext</p> <p>specialTextLayout:</p> <p>0 : full screen</p> <p>1: one line at the bottom</p> <p>2: two lines at the bottom</p>

### Hide Trip data (schnelles Abmelden)

Description	
GetUrl	
PostUrl	http://<display-ip>:<display-port>/display.fcgi/hide
JSON Data Example (Answer for Get / Data in Post)	<pre> {   "tripKey" : "Key_12124343" }</pre>
JSON Post Response	standard response see below
Comments	

### Maintenance get display screenshot

Description	<p>Fetching a screenshot needs two steps 1st take a screenshot and return the path to the file 2nd download the file.</p> <p>The screenshot will be in xwd format.</p>
GetUrl	http://<display-ip>:<display-port>/display.fcgi/maintenance?cmd=get-content&panel=front
JSON Data Example (Answer for Get / Data in Post)	<pre> {   "path": "screenshot/front.png" }</pre>

GetUrl	http://<display-ip>:<display-port>/screenshot/front.png
Comments	Result image will be in png format

### Maintenance deactivate DPI

Description	
GetUrl	http://<display-ip>:<display-port>/display.fcgi/maintenance?cmd=deactivate
JSON Data Example (Answer for Get / Data in Post)	<pre>{   "url" : "/display.fcgi/maintenance?cmd=deactivate",   "result" : "OK",   "resultCode" : 0 }</pre>
Comments	resultCode: 0: OK other: not OK

### Maintenance activate DPI

Description	
GetUrl	http://<display-ip>:<display-port>/display.fcgi/maintenance?cmd=activate
JSON Data Example (Answer for Get / Data in Post)	<pre>{   "url": "/display.fcgi/maintenance?cmd=activate",   "result" : "OK",   "resultCode" : 0 }</pre>
Comments	resultCode: 0: OK other: not OK

### Status update data

Description	The display periodically posts the status to the backend. If the post is not successful the display will recognize offline mode
GetUrl	
PostUrl	http://<backend-ip>:<backend-port>/<backend-web-dfu-url>/<api-version>/status/<display-id>  for example: http://localhost:9999/webdfi/dpic/api/1/status/150
JSON Data	<pre>{   "timestamp" : "2016-12-19T15:10:33+01:00",</pre>



<b>Example</b> (Answer for Get / Data in Post)	<pre> "swVersion" : "457.1.0", "globalState" : 2, "details" : [   {     "status" : 1,     "statusCode" : 10005,     "statusText" : "Service connected"   },   {     "status" : 2,     "statusCode" : 10007,     "statusText" : "No rear display "   } ] </pre>
<b>JSON Post Response</b>	standard response see below
<b>Comments</b>	status: 0: OK 1: Notice 2: Warning 3: Error  globalState: 0: OK 1: Notice 2: Warning 3: Error

## Response data

<b>Description</b>	All http posts will receive this answer data
<b>JSON Data Example</b> (Answer for Get / Data in Post)	<pre> {   "url" : "/display.fcgi/trips",   "result" : "OK",   "resultCode" : 0 } </pre>
<b>JSON Post Response</b>	standard response see below
<b>Comments</b>	The tag <url> is the url of the command that is answered e.g. "/display.fcgi/trips". For defined error codes see table below.

result	resultCode	Description
"OK"	0	Command executed without error

result	resultCode	Description
"INVALID"	1	Invalid JSON (Format error or JSON does not fit to the URL )
"NOTIMPLEMENTED"	2	The Command is not yet implemented