



ermittelten IP-Adresse und der Dienstnummer 80 öffnen (Socket-Adressierung) und ruft zu diesem Zweck eine Funktion der Transportschicht auf.

Die Transportschicht und die darunter liegende Vermittlungsschicht kümmern sich um den Verbindungsaufbau. Hierzu wird unter anderem über ARP die Ethernet-Adresse des Rechners ermittelt, an den die Pakete weitergeleitet werden (in der Regel ein Router). Diese Vorgänge sind für die Anwendungsschicht jedoch unsichtbar; sie wird erst verständigt, sobald die TCP-Verbindung zum genannten Serverprogramm mit der Dienstnummer 80 hergestellt wurde.

15.2.1 Ablauf von HTTP-Anfragen

Nun kann im nächsten Schritt eine HTTP-Anfrage an diesen Server geschickt werden, um die genannte HTML-Seite (index.html) anzufordern. Diese Anfrage ist eine HTTP-Meldung, deren Struktur durch die Spezifikation von HTTP definiert wird.

Jede HTTP-Meldung (engl.: HTTP *message*) besteht aus einem Kopfteil (engl.: *header*), einer Trennzeile (engl.: *separator line*) und einem Nutzdatenteil (engl.: *body*). Der Kopfteil der Meldung enthält die Steuerinformation.

Eine HTTP-Meldung ist entweder eine HTTP-Anfrage oder eine HTTP-Antwort. Eine HTTP-Anfrage eines HTTP-Clients wird durch eine HTTP-Antwort eines HTTP-Servers beantwortet. Die Abb. zeigt ein Beispiel einer HTTP-Anfrage und einer HTTP-Antwort.

Eine HTTP-Anfrage (engl.: HTTP *request*) besteht aus einer Kopfzeile, optionalen Anfrageparametern (engl.: *request header fields*) und einem Nutzdatenteil, der auch leer sein kann. Die Kopfzeile der HTTP-Anfrage enthält die HTTP-Methode (engl.: HTTP *method*), einen Bezeichner für die angeforderte Ressource und die Bezeichnung der verwendeten Version des HTTP-Protokolls.

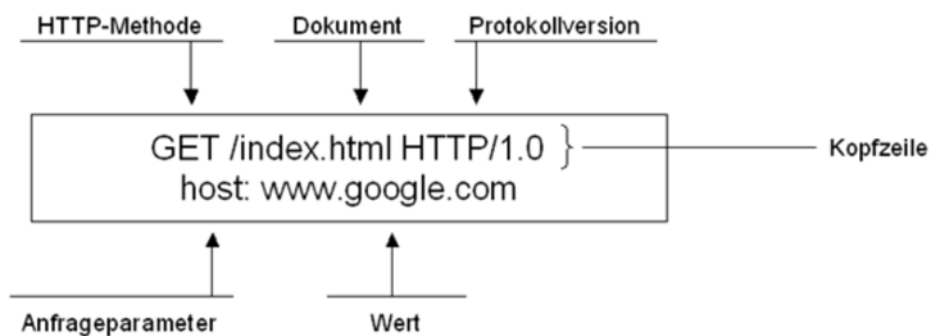
Anfrage von Client an Server

```
GET /index.html HTTP/1.0
host: www.google.com
```

Antwort von Server an Client

```
HTTP/1.0 200 OK
Date: Tue, 26 May 2005 13:47:53 GMT
Content-Length: 4255
Content-Type: text/html
<HTML>
  <HEAD>
  <TITLE>Google</TITLE>
...
</HTML>
```

HTTP definiert mehrere „Methoden“, die den konkreten Diensten eines Webservers entsprechen. Die HTTP-Methode, um ein Dokument anzufordern, ist die Methode GET. Abhängig von der verwendeten HTTP-Methode können unterschiedliche Anfrageparameter angegeben werden. Jeder Anfrageparameter besteht aus einer Zeile, die mit der Bezeichnung des Anfrageparameters gefolgt von einem Doppelpunkt



beginnt, und mit den Wert des Parameters abgeschlossen wird. Im weiteren Verlauf dieses Abschnitts wird noch genauer auf HTTP-Methoden eingegangen. Die folgende Abbildung zeigt die HTTP-Anfrage des obigen Beispiels in größerem Detail.

In diesem konkreten Beispiel ist der Parameter host (der den Zielrechner der Anfrage angibt) der einzige Anfrageparameter. Weitere Beispiele für Anfrageparameter sind if-modified-since oder range. Durch den erstgenannten der beiden Parameter kann ein Dokument unter der Bedingung transferiert werden, dass dieses nach dem im Parameterwert angegebenen Zeitpunkt verändert wurde. Durch den range-Parameter kann ein Teil (ein Ausschnitt) eines Dokuments transferiert werden.

Eine HTTP-Antwort (engl.: HTTP *reply*) besteht aus einer Kopfzeile, optionalen Antwortparametern (engl.: *reply header field*) und einem Nutzdatenteil, der auch leer sein kann. Die Kopfzeile der HTTP-Antwort enthält die Protokollversion, die der HTTP-Server unterstützt, gefolgt von einem Status-Code und einer Status-Meldung.

Der Status-Code der Antwort teilt dem Clienten mit, ob die gewünschte Operation durch den Server ausgeführt werden konnte, und gibt dem Clienten Aufschluss über die anschließend einzuleitenden Operationen. HTTP unterscheidet im Wesentlichen zwischen Erfolgsmeldungen, Warnungen, fehlerhaften Anfragen und serverseitigen Fehlern. Die nachfolgende Tabelle zeigt die wichtigsten im HTTP definierten

Status-Codes mit einer Kurzbeschreibung.

In der HTTP-Antwort werden als Antwortparameter *Date*, *Content-Length* und *Content-Type* verwendet. Der erstgenannte Parameter enthält den Zeitpunkt, an dem die Anfrage beantwortet wurde. Der Antwortparameter *Content-Length* besagt, wie viele Bytes der Nutzdatenteil enthält und *Content-Type* teilt dem Clienten mit, welchen Inhalt der Nutzdatenteil besitzt. Im konkreten Beispiel ist dies ein HTML-Text (der Wert ist text/html).

Die gültigen Werte für den Content-Type-Parameter sind durch den MIME-Standard (Abkürzung von engl.: *multipurpose Internet mail extension*) definiert, durch den beispielsweise Text-, Grafik- und anwendungsspezifische Datenformate beschrieben werden. Anhand der MIME-Formatangaben kann ein Webbrowser erkennen, wie die empfangenen Nutzdaten angezeigt werden sollen. Als wesentliche Optionen kommen hier die Anzeige durch den Webbrowser und die Anzeige durch ein browserexternes Hilfsprogramm (engl.: *external viewer*) in Betracht.

Beispiele für MIME-Formatangaben sind: text/html, text/plain, image/png, image/jpeg oder application/pdf.

15.2.2 HTTP-Methoden

Wie bereits erwähnt, definiert HTTP mehrere „Methoden“, die den konkreten Diensten eines Webservers entsprechen, also angeben, welche Aufgaben ein HTTP-Server erfüllen muss, um der Spezifikation von HTTP zu entsprechen.

	Status-Code	Status-Meldung	Beschreibung
Erfolgsmeldungen:	200	OK	Erfolgreiche Ausführung
Warnungen:	301	Moved Permanently	Ressource wurde auf Dauer an einen anderen Ort verschoben
	302	Moved Temporarily	Ressource wurde temporär an einen anderen Ort verschoben
	304	Not Modified	Ressource wurde nicht verändert (bei bedingter Anfrage)
Ungültige Anfragen:	400	Invalid Request	Ungültige Anfragesyntax
	401	Unauthorized	Benutzer ist nicht berechtigt
	402	Payment Required	Zahlung notwendig
	403	Forbidden	Zugriff ist nicht möglich
	404	Not found	Angefragte Ressource konnte nicht gefunden werden
Fehler des Servers:	500	Internal Server Error	Fehler auf Serverseite