

Prestandatest Mobilbanken 3.0

Utmaningar och styrkor med LR11

Christian Gerdes

Performance Engineer, Swedbank IT

Intro

- Lite kort om Mobilbanken 3.0
 - GUI-less API (REST, JSON)
 - Externa parter bygger klienten, endast ramverket tillgängligt internt
 - Klienter kommer vara touchphone native app's
- Utmaningar med tekniken
 - "REST is stateless" förutom säkerheten
 - REST nyttjar HTTP hårt, många statuskoder, olika svar beroende på Accept, HTTP metoder (PUT, POST, m.fl.)
 - JSON är javascript arrayer, svåra att korrelera

REST - Representational State Transfer

- State på klienten
- En tjänst har en och samma URI
- HTTP metoder avgör vad du vill göra
- Primärt statuskoder som svar
- Accept: headern avgör formatet på svaret
(text application/json, text/xml, text/html)

REST HTTP Metoder (exempel)

URI	Verb (metod)	Action
/Product	GET	Hämtar en lista med alla produkter
/Product	POST	Skapar en ny produkt
/Product/{ProductID}	GET	Returnerar en produkt
/Product/{ProductID}	PUT	Uppdaterar en produkt
/Product/{ProductID}	DELETE	Tar bort en produkt
/ProductGroup/{ProductGroupID}	GET	Returnerar en produktgrupp

Mest kända REST API: WebDAV

REST HTTP Status koder

- **1xx is all about information**

These are not used a lot.

- **2xx is all about success**

Whatever the client tried to do was successful up to the point that the response was send, for example **201 Created** means something on the server side was created successfully. Keep in mind that a status like **202 Accepted** doesn't say anything about the actual result, it only indicates that a request was accepted and is being processed asynchronously.

- **3xx is all about redirection**

These are all about sending the calling application somewhere else for the actual resource. The best known of these are the **303 See Other** and the **301 Moved Permanently** which are used a lot on the web to redirect a browser to another URL.

- **4xx is all about client errors**

With these status codes we indicate that the client has done something invalid and needs to fix the request before resending it, for example **400 Bad Request** often indicating a data input error or **409 Conflict** indicating the client state is out of sync. **404 Not found** can both indicate the URI i.e. the service was not found, or the application is trying to tell the client that a specific resource (with a given id) was not found. **401 Unauthorized** is used when the client needs or could not be authenticated.

- **5xx is all about service errors**

With these status codes we indicate that something went wrong in the service. For example a database connection failed. Typically a client application can retry the request. The server can even specify when the client is allowed to retry the command using a [Retry-After](#) HTTP header. Keep in mind that the webserver can respond with for example **500 Internal Server Error** as well, indicating an unhandled Exception or that the application crashed.

REST vs SOAP

- WSDL
 - SOAP tjänster publicerar sina metoder och datastruktur via WSDL, önskad metod anges tex i en SoapAction header
 - REST saknar motsvarigheten (är dock nått på g i wsdl 2.0)
- Envelope
 - SOAP har ett envelope isolerat från protokollet och kan därmed köras över fler protokoll (som SMTP, TCP)
 - REST är tajt integrerat med HTTP
- XML
 - SOAP är enbart XML
 - REST kan nyttja alla mime-type
- WS-Security
 - REST stödjer det HTTP stödjer (SSL och klient cert, WWW-Authenticate, Cookies)

JSON - JavaScript Object Notation

- Samma format som om du hade skapat datat i javascript kod. Kan därmed extremt enkelt användas av just JavaScript med `eval()`
- Extremt jobbigt att använda om man inte kodar i javascript... eller inte har en `json_eval()` funktion... som tex i Ansi C (LR skript)

JSON Response Example

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "id": "e91570a2-a571-44bf-a6e5-a20293262265",
    "streetAddress": "212ndStreet",
    "city": "NewYork",
    "state": "NY",
    "postalCode": "10021",
    "id": "3474c0e0-21a8-4980-88e0-1286abc951ab"
  },
  "allowedTo": [
    {
      "type": "phone",
      "number": "212555-1234",
      "id": "f528700f-6856-40d4-af5f-006146fc9c09"
    },
    {
      "type": "phone",
      "number": "646555-4567",
      "id": "545b2cd8-49a3-4883-9408-bddab7ae969a"
    }
  ],
  "allowedFrom": [
    {
      "type": "phone",
      "number": "330555-1234",
      "id": "b9df87a0-4bfd-4d6b-bb4f-8f3db45c7380"
    },
    {
      "type": "phone",
      "number": "330555-4567",
      "id": "bb34ac7f-fdb5-44c4-849f-3f05d6865695"
    }
  ]
}
```



```
1 {
2   "firstName": "John",
3   "lastName": "Smith",
4   "age": 25,
5   "address":
6     {
7       "id": "e91570a2-a571-44bf-a6e5-a20293262265"
8     }
9     {
10      "streetAddress": "21 2nd Street",
11      "city": "New York",
12      "state": "NY",
13      "postalCode": "10021",
14      "id": "3474c0e0-21a8-4980-88e0-1286abc951ab"
15    },
16    "allowedTo":
17      [
18        {
19          "type": "phone",
20          "number": "212 555-1234",
21          "id": "f528700f-6856-40d4-af5f-006146fc9c09"
22        },
23        {
24          "type": "phone",
25          "number": "646 555-4567",
26          "id": "545b2cd8-49a3-4883-9408-bddab7ae969a"
27        }
28      ]
29    "allowedFrom":
30      [
31        {
32          "type": "phone",
33          "number": "330 555-1234",
34          "id": "b9df87a0-4bfd-4d6b-bb4f-8f3db45c7380"
35        },
36        {
37          "type": "phone",
38          "number": "330 555-4567",
39          "id": "bb34ac7f-fdb5-44c4-849f-3f05d6865695"
40        }
41      ]
42    }
43  }
```

```
1 {
2   "firstName": "John",
3   "lastName": "Smith",
4   "age": 25,
5   "address":
6     {
7       "streetAddress": "21 2nd Street",
8       "city": "New York",
9       "state": "NY",
10      "postalCode": "10021",
11      "id": "3474c0e0-21a8-4980-88e0-1286abc951ab"
12    },
13   "allowedTo":
14     [
15       {
16         "type": "phone",
17         "number": "212 555-1234",
18         "id": "f528700f-6856-40d4-af5f-006146fc9c09"
19       },
20       {
21         "type": "phone",
22         "number": "646 555-4567",
23         "id": "545b2cd8-49a3-4883-9408-bddab7ae969a"
24       }
25     ]
26   "allowedFrom":
27     [
28       {
29         "type": "phone",
30         "number": "330 555-1234",
31         "id": "b9df87a0-4bfd-4d6b-bb4f-8f3db45c7380"
32       },
33       {
34         "type": "phone",
35         "number": "330 555-4567",
36         "id": "bb34ac7f-fdb5-44c4-849f-3f05d6865695"
37       }
38     ]
39 }
40 }
```

LoadRunner protokoll alternativ

- Ajax TruClient / Mobile TruClient
 - Klarar REST/JSON för JavaScript klienter
- Mobile App
 - Kräver inspelning, endast sniffer/emulator
- WebServices
 - Kräver en WSDL, endast SOAP
- Web DFE
 - Kräver inspelning, inget stöd för proxy-inspelning med DFE

DFE – Data Format Extensions

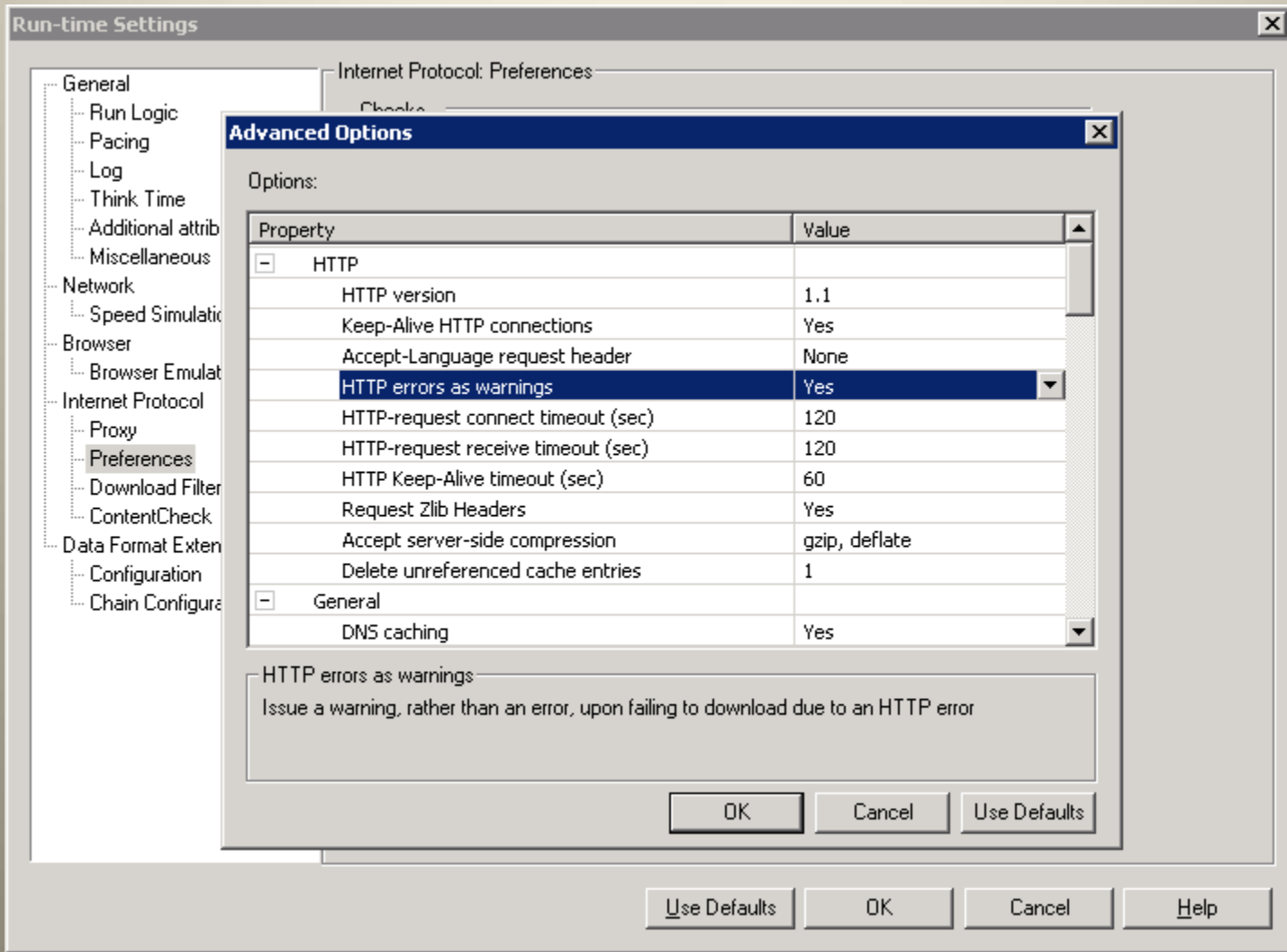
The screenshot shows the 'Recording Options' dialog box with the 'Chain Configuration' section selected under 'Data Format Extension'. A sub-dialog titled 'Add Data Format Extension' is open, displaying a table of available extensions. The 'Json to Xml Extension' is selected.

Name	Tag	Provider	Output Type
Base64 Extension	Base64	HP	text
GWT Extension	GWT	HP	XML
Json to Xml Extension	JsonXml	HP	XML
Prefix Postfix Extension	PrefixPostfix	HP	text
UrlEncoding Extension	UrlEncoding	HP	text
Xml Extension	Xml	HP	XML

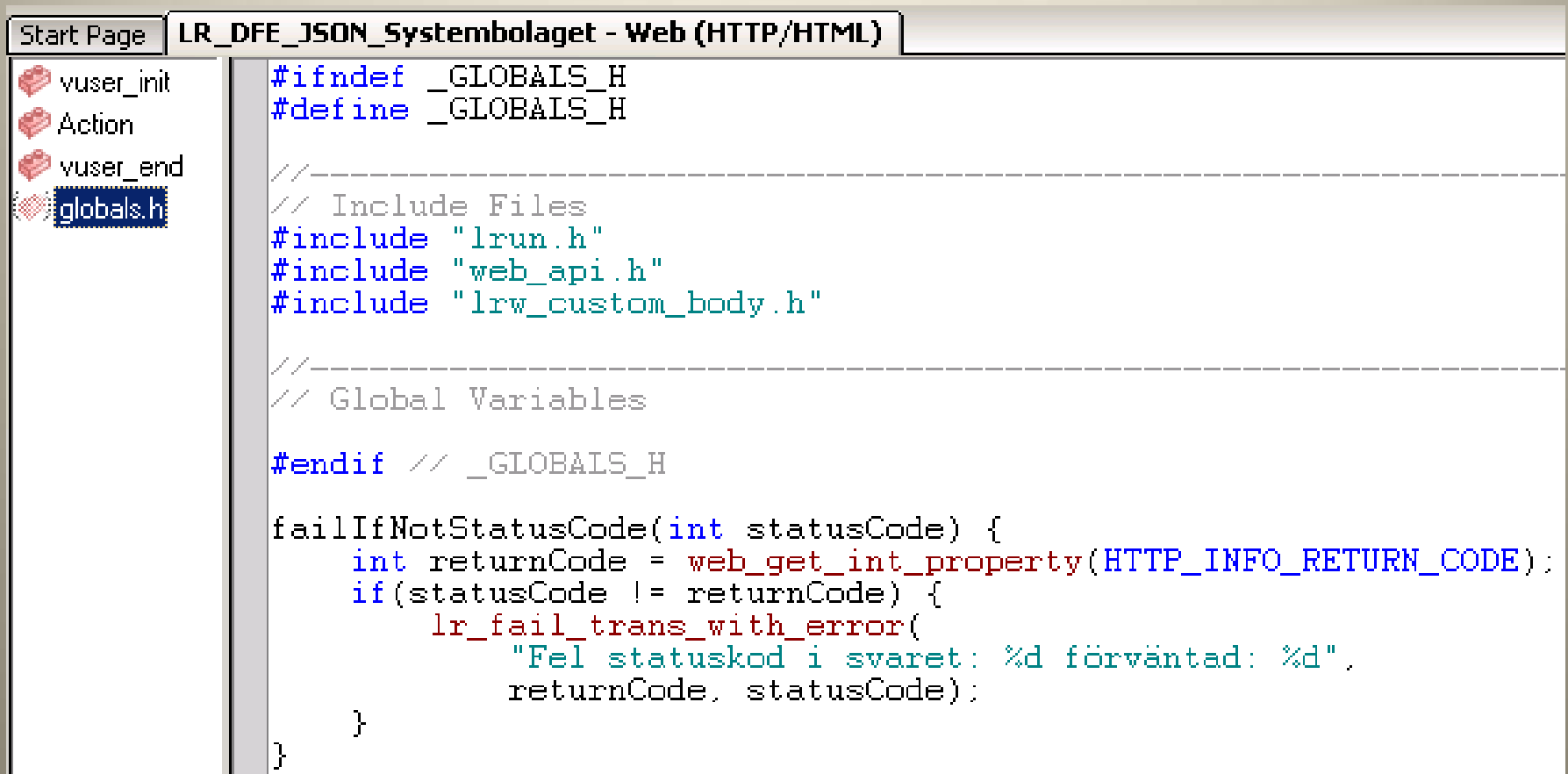
Description: JSON XML DFE

Buttons: Delete, OK, Cancel

Hantera HTTP Status Codes



Manuellt verifiera statuskoden



The screenshot shows a web browser window with a single tab titled "LR_DFE_JSON_Systembolaget - Web (HTTP/HTML)". On the left side, there is a sidebar with a tree view containing the following items: "vuser_init", "Action", "vuser_end", and "globals.h". The "globals.h" item is selected and highlighted. The main content area of the browser displays the source code of the "globals.h" file. The code is as follows:

```
#ifndef _GLOBALS_H
#define _GLOBALS_H

//-----
// Include Files
#include "lrun.h"
#include "web_api.h"
#include "lrw_custom_body.h"

//-----
// Global Variables

#endif // _GLOBALS_H

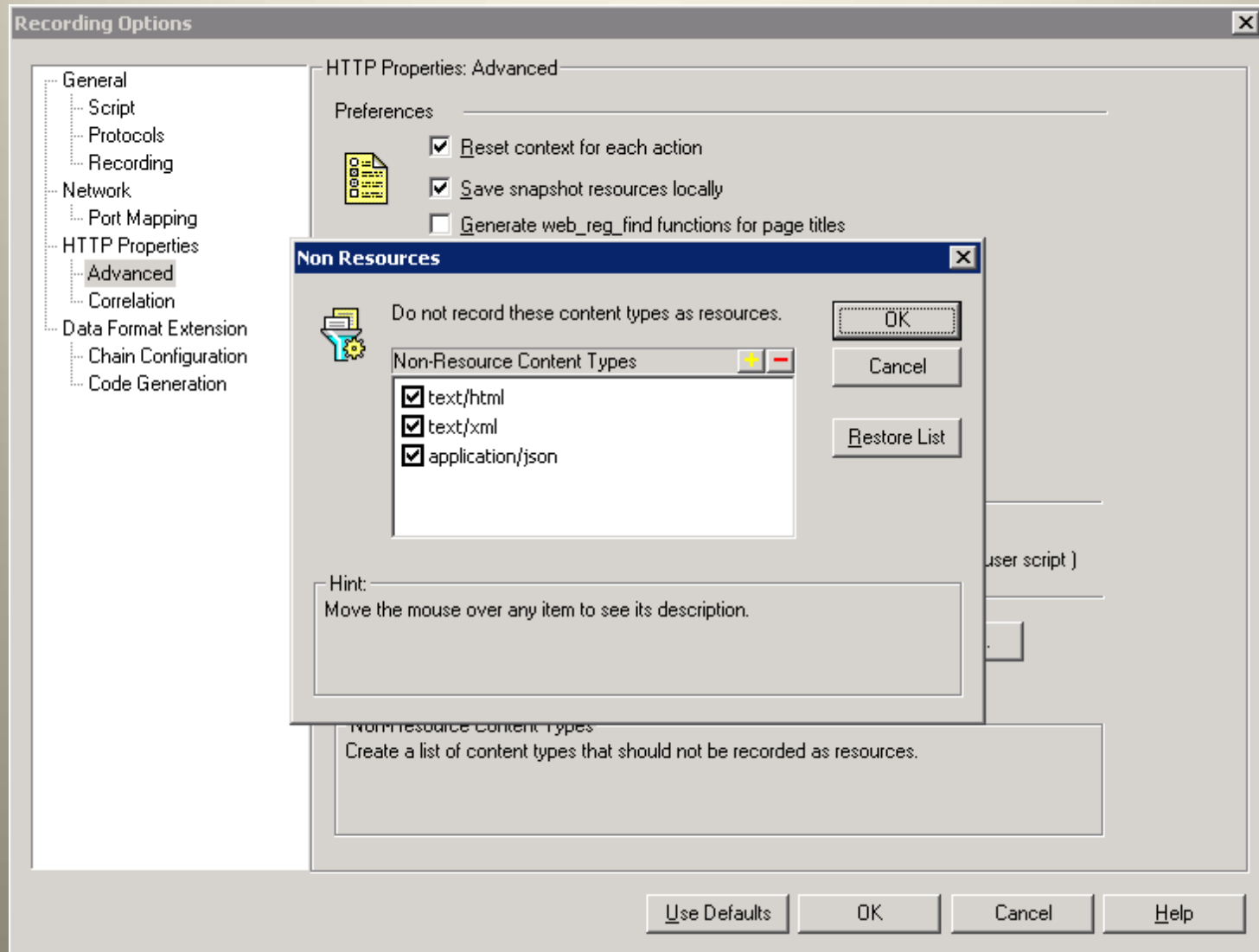
failIfNotStatusCode(int statusCode) {
    int returnCode = web_get_int_property(HTTP_INFO_RETURN_CODE);
    if(statusCode != returnCode) {
        lr_fail_trans_with_error(
            "Fel statuskod i svaret: %d förväntad: %d",
            returnCode, statusCode);
    }
}
```

Manuellt verifiera statuskoden

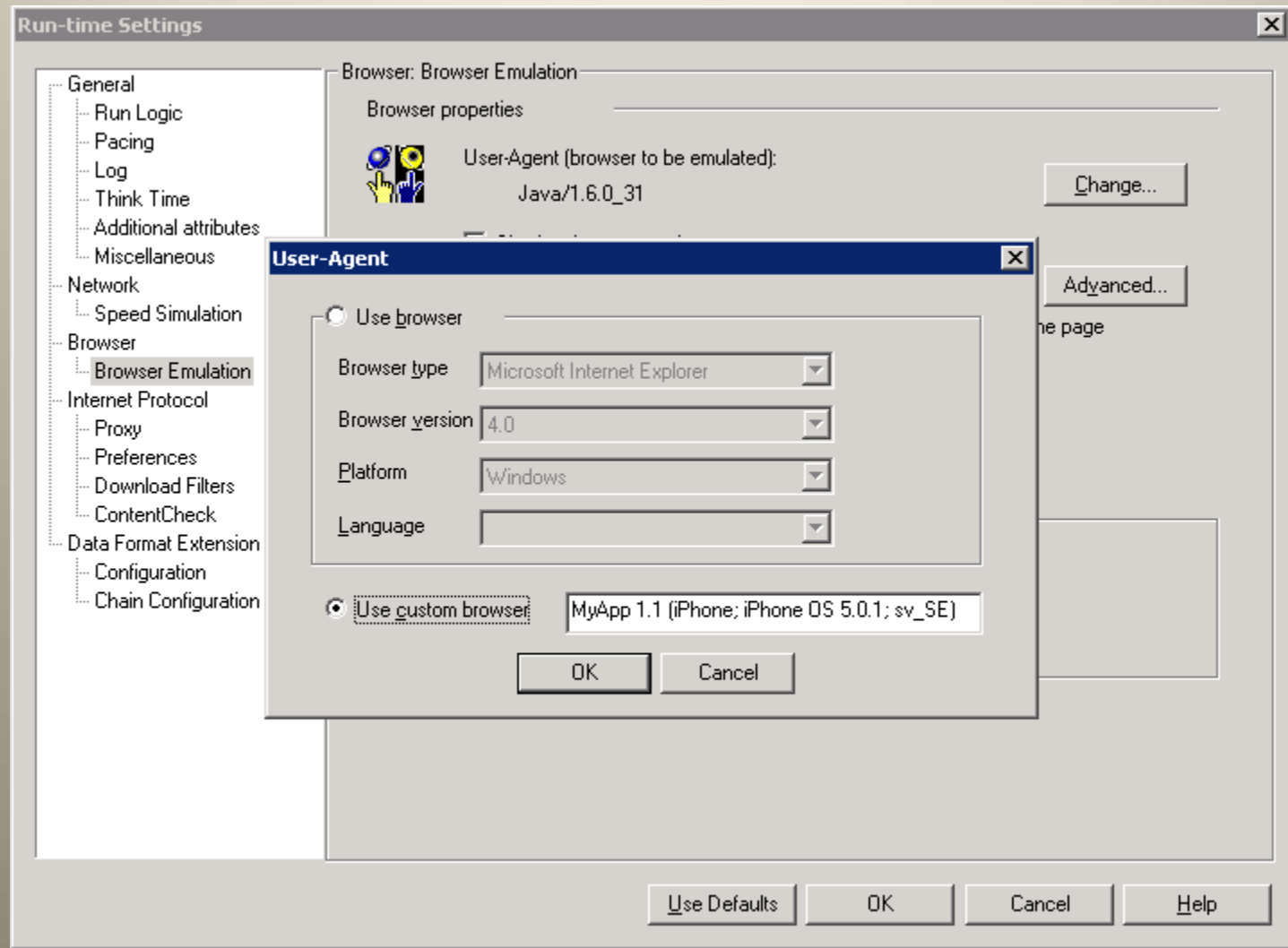
The image shows a software interface with two main components:

- Action List:** A tree view under the heading "Action" containing the following items:
 - Service: Set User
 - Url: SearchArtikel
 - Think Time - 5 (sec)
 - //Correlation comment - Do not ch
 - Web Reg Save Param XPath
 - Url: SearchArtikel_2
 - failIfNotStatusCode** (highlighted)
 - Think Time - 5 (sec)
 - lr_save_string
 - Url: GetArtikel
- User-defined Step Properties Dialog:** A modal dialog box with the following fields and controls:
 - Function Name:** A text box containing "failIfNotStatusCode".
 - Argument List:** A table with one row containing the value "201". To the right of the table are buttons for "Add", "Delete", "Edit Value", "Move Up", and "Move Down".
 - Argument Value:** A large empty text box with a vertical scrollbar on the right side.
 - Buttons:** "OK" and "Cancel" buttons at the bottom right.

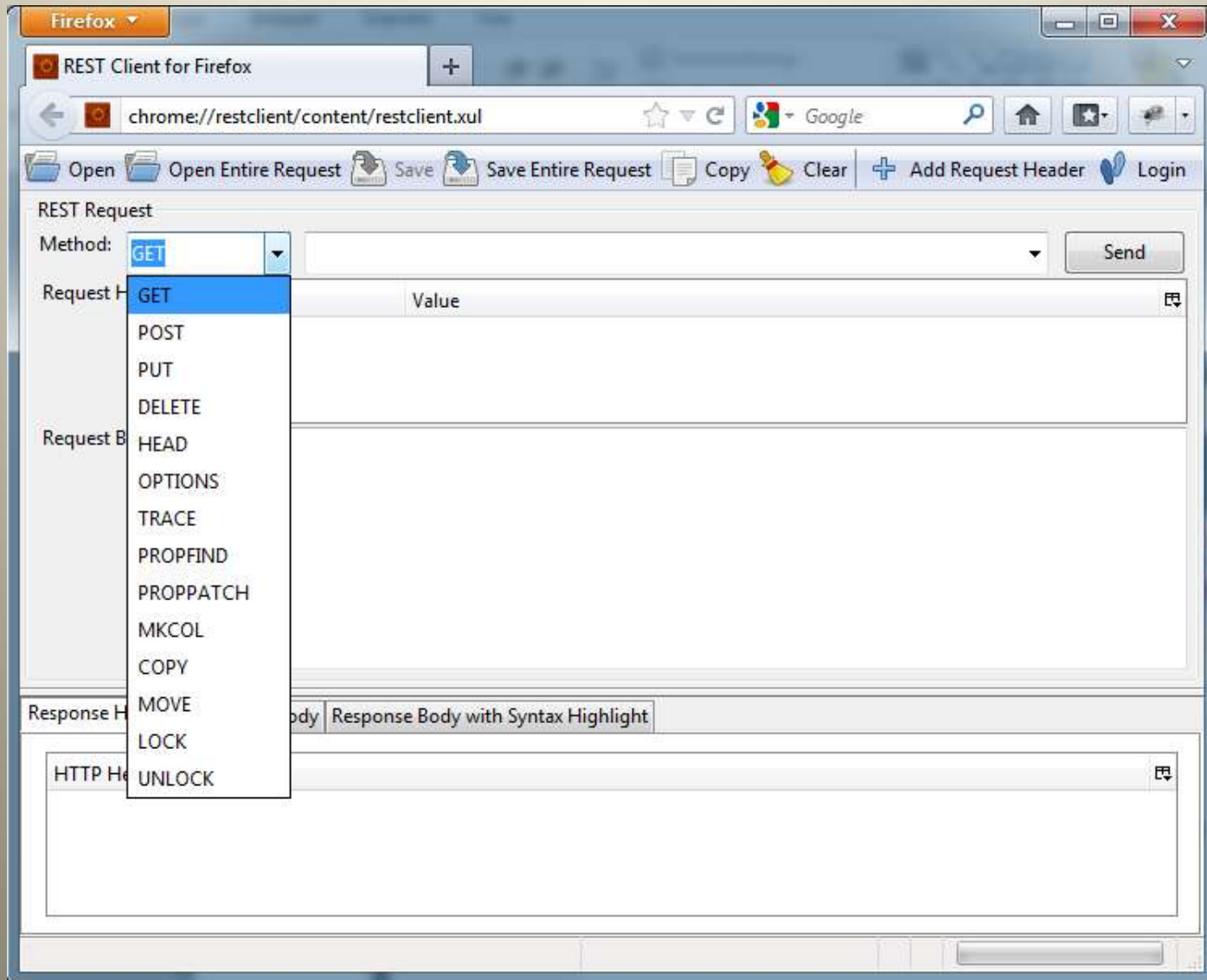
Hantera JSON Response



Emulera touchphonen



Hur spelar man in utan klient?



Hur spelar man in native appar?

- Spela in en SSL Proxy Server!
 - Charles Web Debugging Proxy
 - Microsoft Fiddler 2
 - M.fl.

VILL NI SE EN LIVE DEMO?
KLAPPA HÄNDERNA!