Blank page

A RESTful approach: Clean UPnP without SOAP

Jan Newmarch

UPnP

- An industry standard for home networked devices
- Once FUD, now real technologies
- Includes SOAP as RPC mechanism

SOAP

- Many remote procedure call (RPC) systems exist: ONC, DCOM, CORBA, RMI, ...
- SOAP is an RPC mechanism over HTTP sending XML messages
- SOAP uses HTTP POST with content an XML message which contain procedure names and parameters
- The response is an XML document containing the return value
- Standardised by the WWW Consortium

REST

- REST was invented by Roy Fielding to describe the "ideal" architecture of the Web
- This includes a stateless protocol (no cookies)
- Every resource is addressable
- The addressing scheme is uniform (URLs)
- Small number of operations: PUT, GET, POST, HEAD, ...

REST criticism of SOAP

- SOAP uses HTTP as transport layer, adds its own layer above that - adds complexity
- SOAP ignores the semantic difference between GET and POST, and just uses POST
- SOAP responses are not addressable - a response is an XML document, not the *address* of an XML document
- SOAP as a layered protocol has an arbitrary number of operations

RPC criticisms of SOAP

- The mechanism is heavyweight, requiring XML parsers on client and server
- SOAP has no standard mechanism to return the address of an object, unlike all other RPC systems

UPnP and REST

- UPnP invents its own protcols: e.g. "multicast HTTP" - this is okay
- Devices are described by XML documents at URL addresses - this is okay
- Device descriptions contain URLs of vendor info, etc - this is okay
- Service descriptions contain info about parameters and return types, in an addressable XML document - this is okay
- UPnP specifies SOAP for RPC - this is not so good

UPnP with REST

- Queries for state info should be GET queries, not POST invocations - no need for XML
- Queries for state changes can use much simpler formats than XML documents

UPnP state query with SOAP

```
POST controlUrl HTTP/1.1
HOST: ...
CONTENT-LENGTH: ...
CONTENT-TYPE: ...
SOAP-ACTION: ...
<Envelope>
   <Body>
      <QueryStateVariable>
         <varName> vblName </varName>
      </QueryStateVariable>
   </Body>
</Envelope>
```


UPnP state query with REST

```
GET controlUrl/vblName HTTP/1.1
HOST: ...
```

UPnP state change with SOAP

```
POST controlUrl HTTP/1.1
HOST: ...
CONTENT-LENGTH: ...
CONTENT-TYPE: ...
SOAP-ACTION: ...
<Envelope>
  <Body>
    <actionName>
      <argumentName> value </argumentName>
    </actionName>
  </Body>
</Envelope>
```


UPnP state change with REST

```
POST controlUrl/actionName HTTP/1.1
HOST: ...
CONTENT-LENGTH: ...
CONTENT-TYPE: application/x-www-form-urlencoded

argumentName=value
```

Implementation

- CyberGarage Java implementation uses SOAP
- This was modified to use REST-style queries

Speed

	SOAP/Xerces	SOAP/kXML	REST
1000 queries (secs)	165	114	64

Message sizes

		SOAP	REST
Request	payload	350	4
	packet	535	84
Response	payload	365	11
	packet	526	171

Memory

	SOAP/Xerces	SOAP/kXML	REST
classes+data	416,336	114,808	6,072

Conclusion

- SOAP is claimed to be inappropriate for Web Services, where there is plenty of computing power
- This paper shows that SOAP is also inappropriate for UPnP services

