WebDAV

The good, the bad and the evil

TobiasSchlitt <toby@php.net>

IPC SE 2009

2009-05-30

About me

- Tobias Schlitt <toby@php.net>
- PHP since 2001
- Freelancing consultant
- Qualified IT Specialist
- Studying CS at TU Dortmund

  (expect to finish this year)
- OSS addicted
  - PHP
  - eZ Components
  - PHPUnit

Agenda

Outline

HTTP

- Network protocol driving the web
- Current version: 1.1
- RFC 2616 (June 1999)

  `http://tools.ietf.org/html/rfc2616`
- Originlally invented by Sir Tim Berners-Lee
- Client / server based
- Stateless communication
- Defines
  - Request / response
  - Headers / body
  - Formats / actions

Figure: Tim Berners-Lee [Dan]

WebDAV

WebDAV	HTTP Extensions for Distributed Authoring (RFC 2518)
WebDAV	HTTP Extensions for Web Distributed Authoring and Versioning (RFC 4918)

- IETF Standard
- Specified in RFC 2518 (February 1999)

  `http://tools.ietf.org/html/rfc2518`
- Refined in RFC 4918 (June 2007)

  `http://tools.ietf.org/html/rfc4918`

WebDAV

WebDAV	HTTP Extensions for Distributed Authoring (RFC 2518)
WebDAV	HTTP Extensions for Web Distributed Authoring and Versioning (RFC 4918)

- IETF Standard
- Specified in RFC 2518 (February 1999)

  `http://tools.ietf.org/html/rfc2518`
- Refined in RFC 4918 (June 2007)

  `http://tools.ietf.org/html/rfc4918`
- Extension to HTTP
- Distributed editing

WebDAV

WebDAV	HTTP Extensions for Distributed Authoring (RFC 2518)
WebDAV	HTTP Extensions for Web Distributed Authoring and Versioning (RFC 4918)

- IETF Standard
- Specified in RFC 2518 (February 1999)

  `http://tools.ietf.org/html/rfc2518`
- Refined in RFC 4918 (June 2007)

  `http://tools.ietf.org/html/rfc4918`
- Extension to HTTP
- Distributed editing

WebDAV

... allows your users to edit web content easily.

Request methods

HTTP

- OPTIONS
- GET
- HEAD
- POST
- PUT
- DELETE
- TRACE
- CONNECT

Request methods

HTTP

- OPTIONS
- GET
- HEAD
- POST
- PUT
- DELETE
- TRACE
- CONNECT

WebDAV

- PROPFIND
- PROPPATCH
- MKCOL
- COPY
- MOVE
- LOCK
- UNLOCK

Request methods

HTTP

- OPTIONS
- GET
- HEAD
- POST
- PUT
- DELETE
- TRACE
- CONNECT

WebDAV

- PROPFIND
- PROPPATCH
- MKCOL
- COPY
- MOVE
- LOCK
- UNLOCK

Significant methods

Not all methods are significant for implementing a WebDAV server.

Request headers

HTTP

- Accept[-*]
- Authorization
- If-[None-]Match
- If-[Un]Modified-Since
- User-Agent
- ...

Request headers

HTTP

- Accept[-*]
- Authorization
- If-[None-]Match
- If-[Un]Modified-Since
- User-Agent
- ...

WebDAV

- Depth
- Destination
- If
- Overwrite
- Timeout

Request headers

HTTP

- Accept[-*]
- Authorization
- If-[None-]Match
- If-[Un]Modified-Since
- User-Agent
- ...

WebDAV

- Depth
- Destination
- If
- Overwrite
- Timeout

Significant headers

Not all headers are significant for implementing a WebDAV server.

Response headers

HTTP

- Accept-Ranges
- Content-Length
- Content-Type
- ETag
- Location
- Retry-After
- Server
- WWW-Authenticate
- ...

Response headers

HTTP

- Accept-Ranges
- Content-Length
- Content-Type
- ETag
- Location
- Retry-After
- Server
- WWW-Authenticate
- ...

WebDAV

- DAV
- Lock-Token
- Timeout

Response headers

HTTP

- Accept-Ranges
- Content-Length
- Content-Type
- ETag
- Location
- Retry-After
- Server
- WWW-Authenticate
- ...

WebDAV

- DAV
- Lock-Token
- Timeout

Significant headers

Not all methods are significant for implementing a WebDAV server.

Request / response bodies

HTTP

- Request body mostly not significant
- Only PUT method needs body (to be stored)
- Response body usually content to deliver (unspecified)
- Error responses may contain arbitrary content

Request / response bodies

WebDAV

- Bodies are significant
- Many methods require XML bodies

Request / response bodies

WebDAV

- Bodies are significant
- Many methods require XML bodies

Request

- PROPFIND
- PROPPATCH
- COPY (optional)
- MOVE (optional)
- LOCK

Request / response bodies

WebDAV

- Bodies are significant
- Many methods require XML bodies

Request

- PROPFIND
- PROPPATCH
- COPY (optional)
- MOVE (optional)
- LOCK

Response

- PROPFIND
- PROPPATCH
- LOCK
- Potentially others (multi-status)

Properties

- Concept introduced by WebDAV
- Store meta information about content
- Usually not directly visible to the user

Properties

- Concept introduced by WebDAV
- Store meta information about content
- Usually not directly visible to the user

Pre-defined: Live properties

- creationdate
- displayname
- get*
- lockdiscovery
- resourcetype
- source
- supportedlock

Properties

- Concept introduced by WebDAV
- Store meta information about content
- Usually not directly visible to the user

Pre-defined: Live properties

- creationdate
- displayname
- get*
- lockdiscovery
- resourcetype
- source
- supportedlock

Client-specific: Dead properties

- May contain arbitrary data
- Must reside in their own namespace
- XML representation favored

Outline

Development challenges

- Server development in general
- WebDAV RFCs are a BBOM
    - Unstructured
    - Ambiguous
    - Design fails
- Misbehaving clients
    - Ignore the specification
    - Different interpretations of RFCs
    - Proprietary BS
- Exchangeable back ends

Figure: Big Ball Of Mud [FY99]

Outline - Development challenges

COPY / MOVE methods

Errors on COPY

“[...] if an error occurs while copying an internal collection, the server MUST NOT copy any resources identified by members of this collection (i.e., the server must skip this subtree) [...]” [YG99]

COPY / MOVE methods

Errors on COPY

"[...] if an error occurs while copying an internal collection, the server MUST NOT copy any resources identified by members of this collection (i.e., the server must skip this subtree) [...]" [YG99]

MOVE

"[...] is the logical equivalent of a copy (COPY), followed by consistency maintenance processing, followed by a delete of the source,[...]" [YG99]

COPY / MOVE methods

Errors on COPY

“[...] if an error occurs while copying an internal collection, the server MUST NOT copy any resources identified by members of this collection (i.e., the server must skip this subtree) [...]” [YG99]

MOVE

“[...] is the logical equivalent of a copy (COPY), followed by consistency maintenance processing, followed by a delete of the source,[...]” [YG99]

MOVE errors

“[...] after detecting the error, the move operation SHOULD try to finish as much of the original move as possible [...]” [YG99]

The `If` header

The If header

Makes operations conditional.

The If header

The If header

Makes operations conditional.

Apply operation only if ...

- ... all conditions are met
- ... none of the conditions is met

If header EBNF

```
If            = "If" ":" ( 1*No-tag-list | 1*Tagged-list)
No-tag-list   = List
Tagged-list   = Resource 1*List
Resource      = Coded-URL
List          = "(" 1*(["Not"](State-token | "[" entity-tag
    "]")) ")"
State-token   = Coded-URL
Coded-URL     = "<" absoluteURI ">"
```

The If header

The If header

Makes operations conditional.

Can contain

- lock tokens
- entity tags

No-tag list If header

```
If: (<locktoken:a-write-lock-token> ["I am an ETag"]) (["I am
    another ETag"])
```

The `If` header

The If header

Makes operations conditional.

Conditions can apply to

- single resources
- sets of resources
- all affected resources

Negated tagged list `If` header

```
<http://example.com/resource1> (<locktoken:a-write-lock-token
    > [W/"A weak ETag"]) (Not ["strong ETag"])
```

The `If` header

The If header

Makes operations conditional.

- Required own parser implementation
- Parser is about 150 LOC

Locking

- 2 different types of locks
  - Exclusive
  - Shared

Locking

- 2 different types of locks
  - Exclusive
  - Shared
- Lock conditions must be validated before anything else

Locking

- 2 different types of locks
  - Exclusive
  - Shared
- Lock conditions must be validated before anything else
- Timeout refresh on every lock use (successful or not)
  (Fixed in RCF 4918)

Locking

- 2 different types of locks
  - Exclusive
  - Shared
- Lock conditions must be validated before anything else
- Timeout refresh on every lock use (successful or not)
  (Fixed in RCF 4918)
- Not specified how to associate principles with lock tokens

Locking

- 2 different types of locks
  - Exclusive
  - Shared
- Lock conditions must be validated before anything else
- Timeout refresh on every lock use (successful or not)
  (Fixed in RCF 4918)
- Not specified how to associate principles with lock tokens
- Lock-Null-Resources
  (Partly fixed in RFC 4918)
  - Do not behave like real resources
  - Must vanish when the lock is released
  - A collection can be created on them

Outline - Development challenges

Konqueror / Nautilus

- Konqueror (KDE)
  - Does not decode URLs properly
  - Requires Apache like error messages for 404

Konqueror / Nautilus

- Konqueror (KDE)
  - Does not decode URLs properly
  - Requires Apache like error messages for 404
- Nautilus (Gnome)
  - Cannot cope with `charset="..."` info in MIME types

Windows / InternetExplorer

- At least 3 different WebDAV user agents in Windows
  - Loaded depending on how you initialize connection
  - Transparently switched occasionally

Windows / InternetExplorer

- At least 3 different WebDAV user agents in Windows
  - Loaded depending on how you initialize connection
  - Transparently switched occasionally
- Requires custom header `MS-Author-Via` on every response

Windows / InternetExplorer

- At least 3 different WebDAV user agents in Windows
  - Loaded depending on how you initialize connection
  - Transparently switched occasionally
- Requires custom header `MS-Author-Via` on every response
- Requires custom namespaces set on live properties
  - Requires special namespace shortcut to be used (sic!)
  - Requires different shortcuts for `DAV:` namespace

Windows / InternetExplorer II

- Cannot cope with non-significant white spaces in XML bodies

Windows / InternetExplorer II

- Cannot cope with non-significant white spaces in XML bodies
- Requires newline at the end of every XML body

Windows / InternetExplorer II

- Cannot cope with non-significant white spaces in XML bodies
- Requires newline at the end of every XML body
- Occasionally sends invalid PROPFIND requests

Windows / InternetExplorer II

- Cannot cope with non-significant white spaces in XML bodies
- Requires newline at the end of every XML body
- Occasionally sends invalid PROPFIND requests

Figure: WTF? [Nad99]

Outline - Development challenges

Back end flexibility

- Exchangeable back end
  - File system
  - Memory (testing)
  - SQL Database?
  - Subversion?

Back end flexibility

- Exchangeable back end
  - File system
  - Memory (testing)
  - SQL Database?
  - Subversion?
- Independent of client issues

Back end flexibility

- Exchangeable back end
  - File system
  - Memory (testing)
  - SQL Database?
  - Subversion?
- Independent of client issues
- Protocol enhancements back end independent

Back end flexibility

- Exchangeable back end
  - File system
  - Memory (testing)
  - SQL Database?
  - Subversion?
- Independent of client issues
- Protocol enhancements back end independent
- Easy implementation

Outline

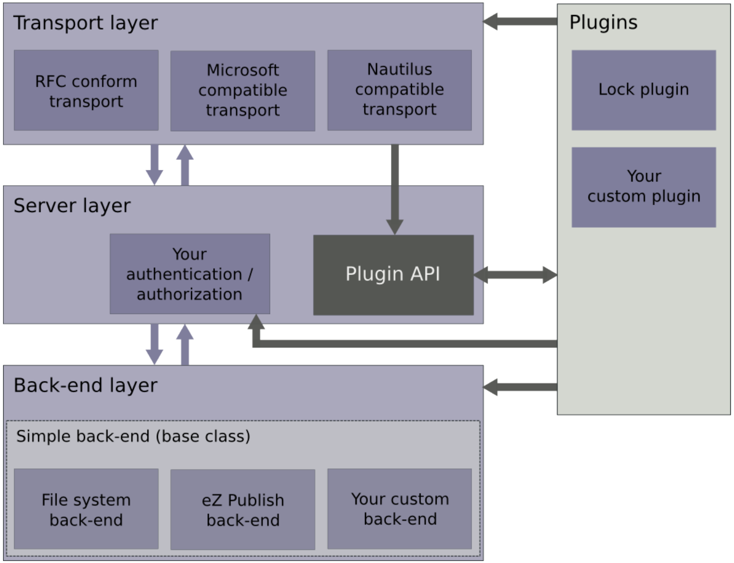
eZ Components

- General purpose component library for PHP 5.1+
- Open source and GPL friendly (New BSD license)
- Vital open source community
- Highly tested
- Extensive docs
- Professional support available

The Webdav component

- General purpose WebDAV server
- Easy integration and customization
- Work around client issues

eZ Webdav architecture



Setup a simple WebDAV server

```php
$server = ezcWebdavServer::getInstance();
$backend = new ezcWebdavFileBackend(
    dirname( __FILE__ ) . '/backend'
);

$server->handle( $backend );
```

WebDAV server with locking

```php
require_once 'custom_lock_auth.php';

$server = ezcWebdavServer::getInstance();

$server->auth = new myCustomLockAuth(
    // Some configuration directory here
    dirname( __FILE__ ) . '/tokens.php'
);

$server->pluginRegistry->registerPlugin(
    new ezcWebdavLockPluginConfiguration()
);

$backend = new ezcWebdavFileBackend(
    // Your WebDAV directory here
    dirname( __FILE__ ) . '/backend'
);

$server->handle( $backend );
```

Outline

Q/A

Thank you for listening!

Q/A

Thank you for listening!

- Are there any questions left?

Q/A

Thank you for listening!

- Are there any questions left?
- Did the session satisfy your needs?

Q/A

Thank you for listening!

- Are there any questions left?
- Did the session satisfy your needs?
- Contact me: Tobias Schlitt <toby@php.net>

Q/A

Thank you for listening!

- Are there any questions left?
- Did the session satisfy your needs?
- Contact me: Tobias Schlitt <toby@php.net>
- Enjoy the rest of the conference!

Outline

- Enrique Dans, *Tim berners-lee*, http://en.wikipedia.org/wiki/File:Tim_Berners-Lee_April_2009.jpg.
- Brian Foote and Joseph Yoder, *Big ball of mud*, http://www.laputan.org/mud/mud.html#BigBallOfMud, 1999.
- Andreas Nadler, *Ich will andere geschenke ...*, http://www.flickr.com/photos/tobi_digital/3145179282/, 1999.
- A. Faizi S. Carter D. Jensen Y. Goland, E. Whitehead, *Http extensions for distributed authoring – webdav*, IETF, Network Working Group, 1999.