iOS App Development

3.2 Web Data Access Part II

Agenda

- URL Loading System

    - NSURLRequest, NSURLConnection

- NSMutableURLRequest

    - Composing HTTP POST Request

URL Loading System

- URL Loading System

  - URL Loading System Programming Guide, Apple

  - Supported Protocols: ftp://, http://, https://, file://

  - NSURLResponse, NSURLRequest, NSURLConnection, NSURLDownload
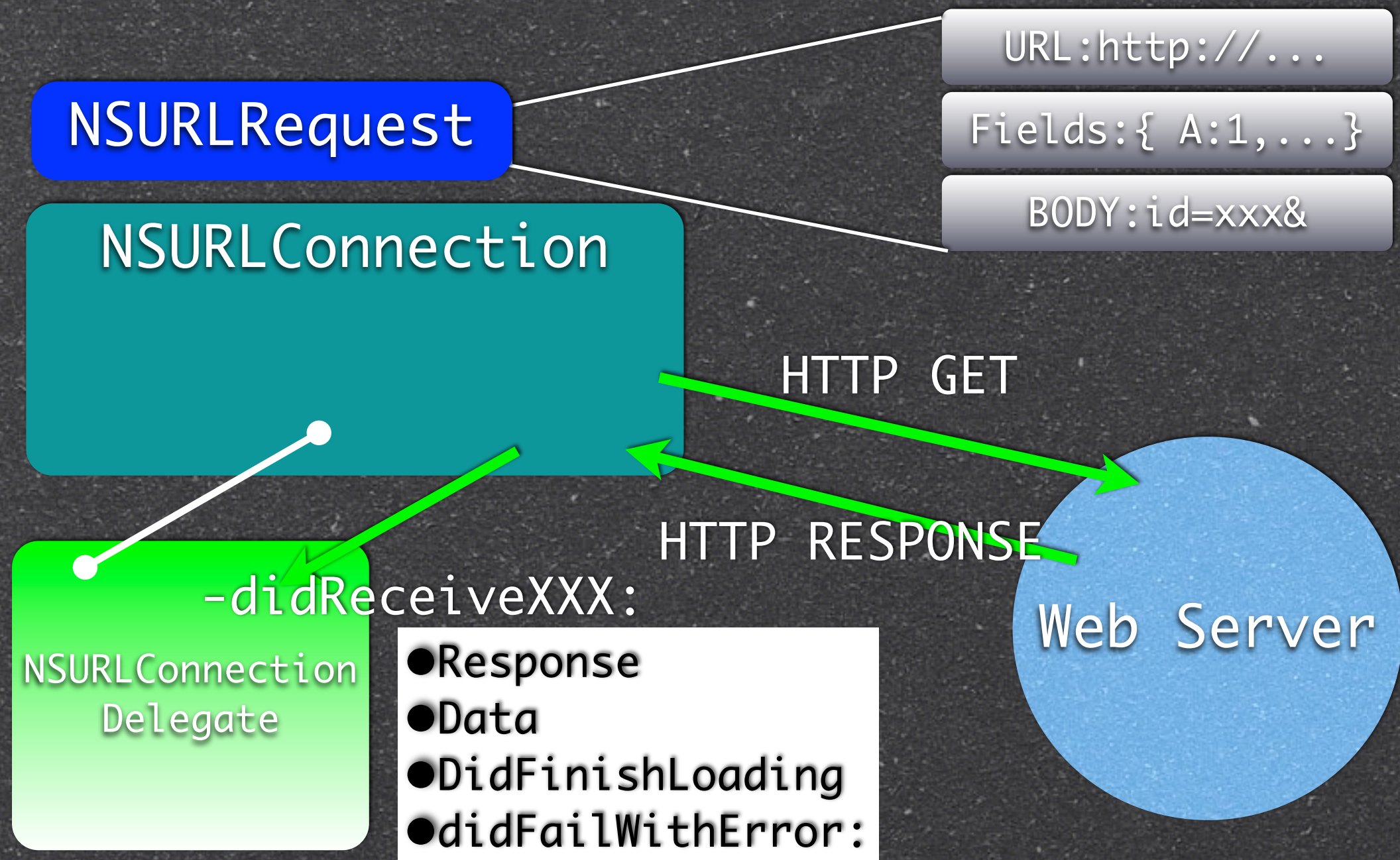
  - Cache Management, Authentication, Cookie, Protocol

URL Loading System

- NSURLRequest, including NSMutableURLRequest represents a "request"

- NSURLConnection provides "connection" for downloading data and/or "response"

- NSURLResponse represents a "response"

URL Loading Practice

NSURLRequest

URL:http://...

Fields:{ A:1,...}

BODY:id=xxx&

NSURLConnection

HTTP GET

HTTP RESPONSE

-didReceiveXXX:

NSURLConnection Delegate

- ●Response
- ●Data
- ●DidFinishLoading
- ●didFailWithError:

Web Server

Simple Loading: HTTP GET

```objc
// Example: Synchronous HTTP GET Request
- (NSArray *) sampleLoadDataList
{
    NSArray *result = nil;
    NSURL *URL = [NSURL URLWithString:@"http://localhost:2403/photos"];
    NSURLRequest *request = [NSURLRequest requestWithURL:URL];
    NSURLResponse *response;
    NSError *error;

    NSData *data = [NSURLConnection sendSynchronousRequest:request
returningResponse:&response error:&error];
    if ( data ) {
        result = [NSJSONSerialization JSONObjectWithData:data options:0
error:&error];
    }
    return result;
}
```

Simple Loading: HTTP GET

```objc
// Example: Asynchronous HTTP GET Request
- (void) sampleLoadDataListDone:(void (^)(NSArray *result)) done
{
    NSURL *URL = [NSURL URLWithString:@"http://localhost:2403/photos"];
    NSURLRequest *request = [NSURLRequest requestWithURL:URL];

    [NSURLConnection sendAsynchronousRequest:request
                                       queue:[NSOperationQueue mainQueue]
        completionHandler:^(NSURLResponse *response, NSData *data, NSError *error) {
            // completion block
            if ( data ) {
                NSArray *result = [NSJSONSerialization JSONObjectWithData:data options:0 error:&error];
                done(result);
            }
    }];
}
```

NSMutableURLRequest

- Mutable version of NSURLRequest

```objc
// Example: Create an HTTP POST request
NSMutableURLRequest *request = [[NSMutableURLRequest alloc] initWithURL:url];
  [request setHTTPMethod:@"POST"];
  [request setValue:@"gzip"
forHTTPHeaderField:@"Accept-Encoding"];
  [request setValue:@"X" forHTTPHeaderField:@"User-Agent"];

NSData *postData = ...; // i.e. XML-RPC payload
  [request setHTTPBody:postData];
  [request setValue:
    [NSString stringWithFormat:@"%d",[postData length]]
            forHTTPHeaderField:@"Content-Length"];
// Not let's send...
```

- and more

References

- URL Loading System Programming Guide, Apple