



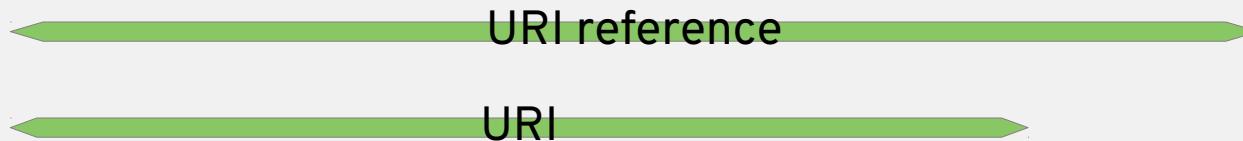
redhat.

LO URI 101

Stephan Bergmann

FOSDEM, February 2018

Anatomy of a URI



scheme:// userinfo @ host : port / path ? query # fragment



percent encoding: %xx

RFC 3986 “Uniform Resource Identifier (URI): Generic Syntax”

Anatomy of an INetURLObject

- Back to the mid '90s...
- Invented its own jargon
 - GetProtocol(), GetParam(), GetMark()
 - partly fixed; even more confusing (setQuery(), setFragment())
- Originally only accepted URLs of a fixed set of schemes
- Always normalizes content
 - if broken web server needs <http://kaputt/foo%21bar>, INetURLObject still gives it <http://kaputt/foo!bar>
- Still insists that file URLs require an authority component
 - <file:///foo/bar>, not <file:/foo/bar>

Anatomy of a URI's payload

- <data:A%42C%20>
 - A -> octet 0x41
 - %42 -> octet 0x42
 - C -> octet 0x43
 - %20 -> octet 0x20
- <file:///foo/bar/baz.txt>
 - corresponds to pathname /foo/bar/baz.txt
- What URL corresponds to pathname /foo/bar/häh.txt
 - <file:///foo/bar/h%C3%A4h.txt>?

StarOffice learns Unicode

- Before:
 - One 8-bit String class
 - URLs written as Strings, payload manipulated as String
- After:
 - Two string classes, 8-bit String and 16-bit UniString
 - URLs written as either, payload manipulated as either
 - But what is UniString GetURLPath("file:///foo/bar/h%C3%A4h.txt")
 - "/foo/bar/häh.txt"
 - "/foo/bar/hÃ¤h.txt"

To cover my ass...

- Every INetURLObject function just got more parameters
 - (Defaulted ones, even where that made little sense)
 - EncodeMechanism:
 - All, WasEncoded, NotCanonical
 - DecodeMechanism:
 - NONE, ToIUri, WithCharset, Unambiguous
 - rtl_TextEncoding eCharset = RTL_TEXTENCODING_UTF8
- INetURLObject("file:///10%2520\$").GetMainURL(WithCharset)
 - "file:///10%20\$" (i.e., "10 \$", not "10%20\$")

Some creative inventions

- <vnd.sun.star.expand:\$FOO/bar/baz>
 - depending on FOO, expands to <file:///some/path/bar/baz>
- <vnd.sun.star.pkg://file:%2F%2F%2Ffoo%2Fbar.odt/path/within/odt>
 - authority encodes <file:///foo/bar.odt>
 - assembling and disassembling such URLs proved error-prone:
 - /foo/bar baz.odt
<file:///foo/bar%20baz.odt>
<vnd.sun.star.pkg://file%2F%2Ffoo%2Fbar%2520baz.odt/...>

UNO to the rescue!

- UNO services to handle such URLs
 - `css::uri::UriReferenceFactory`
 - `XUriReference parse([in] string uriReference);`
 - `css::uri::VndSunStarPkgUrlReferenceFactory`
 - `XUriReference createVndSunStarPkgUrlReference(
[in] XUriReference authority);`
 - Plus some C/C++ stuff in `rtl/uri.hxx`

UNO to the rescue!

- UNO services to handle such URLs
 - `css::uri::UriReferenceFactory`
 - `XUriReference parse([in] string uriReference);`
 - `css::uri::VndSunStarPkgUrlReferenceFactory`
 - `XUriReference createVndSunStarPkgUrlReference(
[in] XUriReference authority);`
 - Plus some C/C++ stuff in `rtl/uri.hxx`
 - *UNO to the rescue?!?*



He said, “How many songs did you write?”
I’d written zero, I lied and said, “Ten.”
“You won’t be young forever
You should have written fifteen.”

—Reed/Cale, Work