

Widget URI Scheme

Widget URI Scheme

To be or not to be? that is the question.

Requirements

Requirements

- hierarchical URI scheme (for DOM).

Requirements

- hierarchical URI scheme (for DOM).
- does not expose the underlying file system (if any).

Requirements

- hierarchical URI scheme (for DOM).
- does not expose the underlying file system (if any).
- must not address resources outside the widget resource via the URI scheme (even if URI scheme allows it).

Requirements

- hierarchical URI scheme (for DOM).
- does not expose the underlying file system (if any).
- must not address resources outside the widget resource via the URI scheme (even if URI scheme allows it).
- would not conflict with browsers.

The Candidates

The Candidates

- file: (file://host/path/to/thing)

The Candidates

- file: (file://host/path/to/thing)
- Content-ID (cid:foo4*foo1@bar.net)

The Candidates

- file: (file://host/path/to/thing)
- Content-ID (cid:foo4*foo1@bar.net)
- http (http://engine/widget/path)

The Candidates

- file: (file://host/path/to/thing)
- Content-ID (cid:foo4*foo1@bar.net)
- http (http://engine/widget/path)
- widget (widget://widget/path/to/thing)

The Candidates

- file: (file://host/path/to/thing)
- Content-ID (cid:foo4*foo1@bar.net)
- http (http://engine/widget/path)
- widget (widget://widget/path/to/thing)
- implementation detail

file: URI Scheme

- PROS

- widely used
- hierarchical

- CONS

- can expose underlying OS*
- no standard
- conflicts (browsers & widgets)

cid: URI Scheme

- PROS

- simplistic

- CONS

- Not specifically hierarchical
- Under-defined
- MIME-based
- obscure

- PROS

- simplistic

- CONS

- Not specifically hierarchical
- Under-defined
- MIME-based
- obscure

http: URI Scheme

- PROS

- well understood
- engine becomes server

- CONS

- engine becomes server
- complex
- overkill
-

widget: URI Scheme

widget: URI Scheme

- PROS
 - designed for requirements
 - does not address file system
 - simple and logical
- CONS
 - new scheme

Implementation detail?

- PROS

- we don't need to specify it
- allows a *de facto* to emerge

- CONS

- irresponsible (can we do better?)
- interoperability problems
- security issues