

**NAME**

URI::data – URI that contains immediate data

**SYNOPSIS**

```
use URI;

$u = URI->new("data:");
$u->media_type("image/gif");
$u->data(scalar(`cat camel.gif`));
print "$u\n";
open(XV, "|xv -") and print XV $u->data;
```

**DESCRIPTION**

The `URI::data` class supports URI objects belonging to the *data* URI scheme. The *data* URI scheme is specified in RFC 2397. It allows inclusion of small data items as “immediate” data, as if it had been included externally. Examples:

```
data:,Perl%20is%20good
```

```

AAgAAAClYyPqcu9AJyCjtIKc5w5xP14xgeO2t1Y3nWcajmZZdeJcG
Kxrmimms1KMTa1Wg8UROx4MNUq1HrycMjHT9b6xKxaFLM6VRKzI+p
KS9XtXpcbdun6uWVxJXA8pNPkdkkxhxc21LzHFogD2KMoQXa2KMWI
JtnE2KizVUkYJVZZ1nczBxXlFopZBtoJ2diXGdNUymmJdFMAADs=
```

URI objects belonging to the data scheme support the common methods (described in `URI`) and the following two scheme-specific methods:

```
$uri->media_type( [$new_media_type] )
```

Can be used to get or set the media type specified in the URI. If no media type is specified, then the default `"text/plain;charset=US-ASCII"` is returned.

```
$uri->data( [$new_data] )
```

Can be used to get or set the data contained in the URI. The data is passed unescaped (in binary form). The decision about whether to base64 encode the data in the URI is taken automatically, based on the encoding that produces the shorter URI string.

**SEE ALSO**

`URI`

**COPYRIGHT**

Copyright 1995–1998 Gisle Aas.

This library is free software; you can redistribute it and/or modify it under the same terms as Perl itself.