texosquery: query OS information from TEX

Nicola L. C. Talbot
dickimaw-books.com

Paulo Cereda

2017-03-31 (v1.3)

Abstract

The texosquery bundle provides the `texosquery.jar` application (and variations `texosquery-jre8.jar` and `texosquery-jre5.jar`) This is a cross-platform Java application to query certain operating system (OS) and locale information. The application is specifically designed for use within TEX's shell escape mechanism, through the `\TeXOSQuery` command provided by the texosquery package (`texosquery.tex` and `texosquery.sty`).

The `\TeXOSQuery` command performs more than a simple piped input as it first changes category codes of various problematic characters and locally defines some short control sequences that are used in the application's result. These commands aren't defined outside of `\TeXOSQuery`, so a direct piped input may cause undefined control sequences. If you really want to use this direct method rather than using `\TeXOSQuery`, then you will need to run `texosquery` in backward compatibility mode 0 or 1 (using `--compatible 1`). The first two versions of `texosquery` didn't use those short commands.

Important Note: You will need TEX's shell escape enabled, and you will also need the Java Runtime Environment (JRE) installed.

There are three variations of the `texosquery` application provided:

- `texosquery.jar`: requires at least Java 7, has medium locale support, obeys `openin_any` but has additional restrictions imposed for security reasons (no listings outside the current working directory path); Note that Java 7 has reached its end of life and is now deprecated.
- `texosquery-jre8.jar`: requires at least Java 8, has best locale support, obeys `openin_any` but has additional restrictions imposed for security reasons (no listings outside the current working directory path);
- `texosquery-jre5.jar`: requires at least Java 5, has poor locale support (language scripts not recognised), doesn't have the walk action, obeys `openin_any` but doesn't have the extra restrictions of the Java 7 and 8 versions for the listing functions. Note that Java 5 and 6 are deprecated. Old deprecated versions are considered a security risk.

The default is `texosquery.jar`. Throughout this document `texosquery` is used to reference the application, regardless which of these three jar files you've chosen to use. See section 1.1 for further details.

The aim of the original version of `texosquery` was to provide a way of accessing the operating system's locale information. Version 1.3 of the tracklang package provides `\TrackLangQueryEnv` which uses `kpsewhich` to query the appropriate locale environment variable (such as `$LANG` or `LC_ALL`). Unfortunately this doesn't work under Windows as the locale information there is stored in the registry. The Lua `os.setlocale(nil)` function can simply return `C` or `POSIX`, which isn't helpful from tracklang's point of view. Although Java has its drawbacks, it's one of the most ubiquitous platform-independent methods to obtain this information. Since it seemed overkill to write a Java application that simply returned the locale, I decided to add a few extra functions that might be of use, but accessing locale information was, and still is, the primary purpose of this application.

Although the POSIX environment variables, such as `$LC_ALL`, are easy to read with `kpsewhich`, these days the IETF BCP 47 language tag is the more appropriate way of identifying a locale, so version 1.2 has added the `--bcp47` function to support this. The tracklang package has similarly added `\TrackLanguageTag{`⟨*IETF tag*⟩`}`.

The locale package occasionally referenced in this document is still under development at the time of writing. The `--numeric`, `--locale-data`, `--date-time` and `--time-zones` options are designed to interface with the locale package, so although texosquery and locale will be distributed separately, version 1.2 of `texosquery` is being developed alongside version 1.0 of the locale package. The aim of the locale package is to use both tracklang and texosquery to automatically set up the document language. For example, in the following LaTeX document

```
\documentclass{article}
\usepackage{locale}
\begin{document}
Language: \CurrentLocaleLanguageNativeName.
Region: \CurrentLocaleRegionNativeName.
Today: \CurrentLocaleDate. (Compare with \today.)
Time: \CurrentLocaleTime.
Currency Symbol: \CurrentLocaleCurrency
Integer:
\texosqueryfmtnumber{\CurrentLocaleIntegerPattern}{123456}{0}{0}
Decimal:
\texosqueryfmtnumber{\CurrentLocaleDecimalPattern}{123456}{78}{0}
Percentage:
\texosqueryfmtnumber{\CurrentLocalePercentPattern}{0}{65}{0}
Currency:
\texosqueryfmtnumber{\CurrentLocaleCurrencyPattern}{1234567}{0}{0}
\end{document}
```

the locale package will automatically:

- load the textcomp package for currency symbols (package option `symbols=fontawesome` will use fontawesome instead);
- if XeLaTeX or LuaLaTeX:

  - load fontspec (unless option `fontspec=false` is used);
  - load polyglossia and use `\setmainlanguage` with options that can be determined from the language tag (use package option `support=babel` to use babel regardless of the LaTeX format);

  otherwise:

  - load inputenc (default file encoding obtained from `texosquery`'s `--codeset-lcs` action);
  - load fontenc (font encoding obtained using tracklang to query the language script);
  - load babel with the appropriate language label (use `support=none` to prevent this);

- load datetime2 with the `useregional=text` option (use `datetime2=false` to prevent this).

The generic `locale.tex` code doesn't load the above packages, but can still obtain information about the locale:

```
\input locale

Language: \CurrentLocaleLanguageNativeName.
Region: \CurrentLocaleRegionNativeName.
Today: \CurrentLocaleDate. (Compare with \today.)
Time: \CurrentLocaleTime.
Currency Symbol: \CurrentLocaleCurrency
Integer:
\texosqueryfmtnumber{\CurrentLocaleIntegerPattern}{123456}{0}{0}
Decimal:
\texosqueryfmtnumber{\CurrentLocaleDecimalPattern}{123456}{78}{0}
Percentage:
\texosqueryfmtnumber{\CurrentLocalePercentPattern}{0}{65}{0}
Currency:
\texosqueryfmtnumber{\CurrentLocaleCurrencyPattern}{1234567}{0}{0}
\bye
```

So that's the reasoning behind the new v1.2 actions. Hopefully the new locale package will be uploaded to CTAN shortly after the new version of texosquery.

Contents

1 texosquery.jar: the Java application

The texosquery Java command line application looks up certain system information that may be of use in TEX documents. This information can be obtained using native commands, but the Java application allows an OS-independent approach with results that can easily be captured by TEX's shell-escape without having to strip formatting information. It also uses control sequence markup to indicate whether characters should be interpreted literally (such as in file names) or if they should obey their current category code (such as punctuation occurring in textual information) or if they should be interpreted in some other way (such as pattern markup). This markup is expanded by \TeXOSQuery when it performs the piped shell escape.

> **Important Note:** texosquery provides read-only actions, and I don't intend adding any actions that modify system settings or files.

Since the application is designed to work with TEX (through \TeXOSQuery defined in texosquery.tex) each action (indicated by a command line switch) will display the result on a single line. For multiple results, each line is grouped. A blank line (or empty group) will be displayed if the information isn't available or is prohibited. A forward slash (\fslh) is always used as a directory divider, regardless of the operating system, so the result can be used, for example, in \input or \includegraphics.

For example, I have a 64-bit Linux operating system installed on my computer, so I could use uname in a bash terminal:

```
uname -o -r
```

which (for me) produces:

```
4.1.13-100.fc21.x86_64 GNU/Linux
```

I could also run texosquery directly from the bash terminal:

```
texosquery -o -r
```

which produces the rather more cryptic:

```
{Linux}
{4\fdot 1\fdot 13\fhyn 100\fdot fc21\fdot x86\fusc 64}
```

However texosquery isn't intended for this direct use. It's intended for use with \TeXOSQuery provided by texosquery.tex. Here's a plain TEX document:

```
\input texosquery
\TeXOSQuery{\result}{-o -r}
\def\parseresult#1#2{OS Name: {\tt #1}. OS Version: {\tt #2}.}
\ifx\result\empty
 Query failed!
\else
 \expandafter\parseresult\result
\fi
\bye
```

The markup commands, such as `\fusc`, are now converted to literal characters with category code 12 ("other"), so the underscore isn't a problem. This document is now also platform independent (as long as `texosquery` and a recent version of the JRE are installed). Unlike `uname`, `texosquery` also obeys the order of the command line switches, which makes it easier to define the helper command (`\parseresult` in the above) that processes the result.

1.1 Installation and Setup

Installation is best done through your TEX package manager. However if for some reason you need to install this package manually the instructions are below. If you install through your package manager, Windows users will probably find that the `.jar` files have been converted to `.exe` (with the `.bat` files omitted) and Unix-like users may find that the bash scripts are missing the `.sh` extension (these are actually symbolic links to the distributed `.sh` files). See section 1.2 to test that the package has been successfully installed.

Even if you use your TEX distribution's package manager to install this package, you may still need to edit the `texosquery.cfg` file (see step 3 below). If you don't have write permission for this file you can copy it to ⟨*TEXMFHOME*⟩`/tex/generic/texosquery/` (that directory path may need to be created if it doesn't already exist). You can find the correct value of ⟨*TEXMFHOME*⟩ using

```
kpsewhich -var-value=TEXMFHOME
```

You can find where the package manager has put `texosquery.cfg` using

```
kpsewhich texosquery.cfg
```

This bundle contains the following files:

- `texosquery.dtx`

  The DTX file contains the source code for this document, and also the files:

  - `texosquery.tex` (generic TEX code)
  - `texosquery.sty` (LATEX package wrapper)
  - `texosquery.cfg` (configuration file)

  The bash scripts (which will need the extensions removed):

  - `texosquery-jre8.sh`
  - `texosquery.sh`
  - `texosquery-jre5.sh`

  Windows batch files (which will need the extensions changed to `.bat`)

  - `texosquery-jre8.batch`
  - `texosquery.batch`
  - `texosquery-jre5.batch`

- `texosquery.ins` The driver file used to extract all the above files contained in `texosquery.dtx`.
- The three different versions of the `texosquery` application: `texosquery-jre8.jar`, `texosquery.jar` and `texosquery-jre5.jar`. The source code for these is contained in the `java` sub-directory.
- `texosquery.pdf` This PDF document.
- `README.md` The README file in markdown format.
- `CHANGES` Lists major changes for each version.

To install manually (⟨*TEXMF*⟩ indicates the TEXMF directory):

1. Run

   ```
   tex texosquery.ins
   ```

   to extract the `.tex`, `.sty`, `.cfg`, `.sh` and `.batch` files.

   Windows Change the extension of the `.batch` files to `.bat` (TEX on Windows prohibits the creation of `.bat` files). Move the `.bat` files to somewhere on your system's path. (You may omit the `.bat` files you don't need.) The `.sh` files may be deleted.

   Unix-like Make the `.sh` files executable:

   ```
   chmod u+x texosquery*.sh
   ```

   Move the `.sh` files to somewhere on your path *without* the `.sh` extension. (If the `.sh` extension is retained, you will have to edit the `texosquery.cfg` file to include it.) For example (if `~/bin` is included in `$PATH`):

   ```
   mv texosquery-jre8.sh ~/bin/texosquery-jre8
   ```

   (You may omit the `.sh` files you don't need.) The `.batch` files may be deleted.

2. Move `texosquery.tex` to ⟨*TEXMF*⟩`/tex/generic/texosquery/`
3. Edit `texosquery.cfg` so that `\TeXOSInvokerName` is defined to the application of your choice. For example, if you have Java 8 installed:

   ```
   \def\TeXOSInvokerName{texosquery-jre8}
   ```

   Or if you only have Java 5 or 6 installed:

   ```
   \def\TeXOSInvokerName{texosquery-jre5}
   ```

   You can find out your Java version by running the following in your command prompt or terminal:

   ```
   java -version
   ```

If the version number starts with `1.8` then you have Java 8 installed, if it starts with `1.7` then you have Java 7, etc.

4. Move `texosquery.cfg` to ⟨*TEXMF*⟩`/tex/generic/texosquery/`
5. Move `texosquery.sty` to ⟨*TEXMF*⟩`/tex/latex/texosquery/`
6. Move the `.jar` files to ⟨*TEXMF*⟩`/scripts/texosquery/`

1.2 Installation Test

To test the installation:

1. In the command prompt or terminal do:

   ```
   texosquery -b
   ```

   (Replace `texosquery` with the command that matches the value of `\TeXOSInvokerName` in the `texosquery.cfg` file described in section 1.1.) The above command should display the system's default locale. For me, this simply displays the line:

   ```
   en-GB
   ```

   If you get an `Unknown option '-b'` error, then your OS is picking up an old version of `texosquery`. Check the version number with the `-v` switch.

   ```
   texosquery -v
   ```

   If you get a "command not found" or "bad command or file name" error, then recheck the installation steps in section 1.1 and make sure that the executable file has been placed on your system's path.

   If this test is successful, try the next step.

2. Create the following plain TEX document called `test.tex`:

   ```
   \input texosquery
   \TeXOSQuery{\result}{-b}\result
   \bye
   ```

   and compile using:

   ```
   pdftex --shell-escape test
   ```

   Alternatively, create the follow LATEX document called `test.tex`:

   ```
   \documentclass{article}
   \usepackage{texosquery}
   \begin{document}
   \TeXOSQuery{\result}{-b}\result
   \end{document}
   ```

and compile using:

```
pdflatex --shell-escape test
```

In both cases, the resulting PDF file `test.pdf` should show the default locale. If not check the transcript `test.log` which should include something like `(|texosquery -b)` or `(|texosquery-jre8 -b)` etc. If it simply has the line:

```
TeXOSQuery: texosquery -b
```

(or similar) then the dry run mode was on, which means the shell escape wasn't used. Check that the `--shell-escape` switch was used when calling `pdftex` or `pdflatex`.

If you are using `texosquery` or `texosquery-jre8` and your chosen application has been added to the restricted list, make sure that the line

```
\TeXOSQueryAllowRestricted
```

hasn't been commented out in the `texosquery.cfg` file and try the above example documents in restricted mode. (Note that `texosquery-jre5` should not be added to the restricted list as Java 5 and 6 are now deprecated and considered security risks, so the above command should be commented out in the `texosquery.cfg` file if `\TeXOSInvokerName` has been set to `texosquery-jre5`.)

1.3 Accessing file information

If an input file name is required (for example, with the `--pdfdate` argument described below) then the file may be in the current working directory, relative to the current directory (with forward slash `/` as the directory divider), an absolute path (again with forward slash) or on TEX's path (in which case, `kpsewhich` is used to locate it). As from version 1.2, `texosquery` honours the `openin_any` attribute set in the `texmf.cnf` configuration file. This value is fetched using

```
kpsewhich -var-value=openin_any
```

(You can find the configuration files using `kpsewhich -a texmf.cnf`) For example, suppose the file `/tmp/.test` exists. If the `openin_any` attribute is set to "`a`" (any file), then (assuming the operating system allows read-access to that file) the `texosquery` file-reading operations will be permitted. For example

```
texosquery --pdfdate /tmp/.test
```

will return the file modification date in PDF date-time format. However, if `openin_any` is set to "`r`" (restricted), the read access will be denied because the file is considered hidden so an empty result is returned. Similarly, if `openin_any` is set to "`p`" (paranoid), the read access will be denied again because the file is hidden but also because the file has an absolute path that isn't under `$TEXMFOUTPUT` (assuming that environment variable hasn't been set to `/tmp`).

1.4 MikTeX

Apparently MikTeX doesn't support the `openin_any` variable, so if this is unset, `texosquery.jar` and `texosquery-jre5.jar` will fallback on "a", but the stricter `texosquery-jre8.jar` will fallback on "p".

Note that MiKTeX disables piped input by default for security reasons. Since `\TeXOSQuery` relies on piped input, you'll need to enable it with `--enable-pipes` when you run TEX.

1.5 Return Values

The return values may include literal text where special characters need to have their category code changed to 12 (for example, file names) but the return values may also include TEX code that needs to be processed by TEX, either during the shell escape or deferred for later (such as date-time or numeric patterns). This means that the result from the shell escape can't be automatically detokenized.

Therefore, as from version 1.2, the return values include short control sequences that are locally defined by `\TeXOSQuery` and so are only valid within that command's scope. For example, `\fcln` expands to a colon (`:`) with category code 12 whereas `\tcln` expands to a colon according to its current meaning. Note that this has changed from earlier versions which simply returned the actual characters, which may or may not have had the category code set to 12 at the start of `\TeXOSQuery`. To reproduce the original behaviour, use the compatibility mode (`--compatible`) with the level set to 0 or 1. For the full list of shortcut commands, see the definition of `\@texosquery@enableshortcs`.

The output produced by the `texosquery` application will be returned using the system's default input encoding. (For example, UTF-8.) You will need to ensure that your TEX document uses the same encoding if you want to typeset any of the results that may contain non-ASCII characters. You can determine the default encoding with `texosquery -C`, which is formatted to match the options used by the inputenc package. (For example, `uft8` for UTF-8.)

To test the file encoding rerun the plain TEX or LATEX test file in section 1.2 with `-N` instead of `-b`. Most currency symbols are outside the ASCII set, so this should return a non-ASCII character. If you happen to have $ as your currency, then try `en-GB` or `en-IE` which have £ and €, respectively. LATEX users may need to load inputenc and fontenc. XELATEX and LUALATEX users may need to load fontspec.

You can change the default encoding by invoking the Java Virtual Machine with the option `-Dfile.encoding=`⟨*codeset*⟩. For example, bash users can modify the `texosquery` (or `texosquery.sh`) script to set the encoding to UTF-8 as follows:

```
#!/bin/sh

jarpath=`kpsewhich --progname=texosquery --format=texmfscripts texosquery.jar`
java -Dfile.encoding=UTF-8 -jar "$jarpath" "$@"
```

Similarly for the corresponding `.bat` file for Windows users.

If the script file can't be modified or you have only `.exe` instead of `.jar` files then you can set the `JAVA_TOOL_OPTIONS` environment variable. For example:

```
declare -x JAVA_TOOL_OPTIONS=-Dfile.encoding=UTF-8
```

Note that, unlike the edit to the bash or `.bat` file, this will affect all your Java applications.

1.6 Locales

The options that have a locale identifier as an argument need the identifier formatted as a *regular* IETF BCP 47 language tag that uses hyphens as separators. POSIX style locales (with underscores replaced by hyphens, for example `fr-BE.utf8@euro`) are only used as a return value in the `--locale` and `--locale-lcs` options.

Not all locales are supported by Java. For example, Irish is supported but Scottish and Welsh aren't supported by the JRE. The Unicode Consortium's Common Locale Data Repository (CLDR) can be accessed with Java 8, but the CLDR isn't enabled by default. It can be turned on using the system property `java.locale.providers`, which may provide additional support. For example, although Welsh isn't supported by the JRE, it is supported with the CLDR, so both Java 8 and the CLDR locale provider are required for that language. The proposed Java 9 should have the CLDR enabled by default.

The bash script `texosquery-jre8` (or `texosquery-jre8.sh`) automatically sets `java.locale.providers` to `CLDR,JRE`. Alternatively, the `JAVA_TOOL_OPTIONS` environment variable can be set to

```
-Djava.locale.providers=CLDR,JRE
```

which will enable it for all installed Java applications. If you need to set multiple options, these can be combined in the value of `JAVA_TOOL_OPTIONS`. For example

```
-Djava.locale.providers=CLDR,JRE -Dfile.encoding=UTF-8
```

You may find that the results are different depending on the data provider. For example with `java.locale.providers` set to `JRE,CLDR` then

```
texosquery -D en-GB
```

displays the long date in the form "06 November 2016" and the medium date in the form "06-Nov-2016", but with the ordering reversed to `CLDR,JRE` (so that the CLDR is queried first) then the long date is now in the form "6 November 2016" and the medium date is in the form "6 Nov 2016".

Note that `texosquery` can only access locale information provided by Java. For example, Java currently doesn't provide any methods to access telephone codes.

1.7 Command line invocation

The syntax for the command line invocation of `texosquery` is:

`texosquery [`⟨*options*⟩`]` ⟨*action*⟩ `...`

The syntax for `texosquery-jre8` is exactly the same except for the application name:

`texosquery-jre8 [`⟨*options*⟩`]` ⟨*action*⟩ `...`

Similarly for `texosquery-jre5`. (Bash users may need the `.sh` extension if it wasn't removed from the script name during the installation setup.) Available actions are described below. At least one action is required.

Available options (must come before actions):

`-h` **or** `--help` **or** `-help` Displays help message and exits.

`-v` **or** `--version` **or** `-version` Displays version information and exits.

`--nodebug` No debugging information. Only command line syntax errors are written to STDERR. (Default.)

`--debug [`$\langle n \rangle$`]` **or** `-debug [`$\langle n \rangle$`]` Set the debugging level, where $\langle n \rangle$ is a non-negative integer. If $\langle n \rangle$ is omitted, 3 is assumed. If $\langle n \rangle$ is 0, then debugging information is suppressed (equivalent to `--nodebug`). If $\langle n \rangle \geq 1$, error messages are written to STDERR. If $\langle n \rangle \geq 2$, any exceptions encountered will additionally write the stack trace to STDERR. If $\langle n \rangle \geq 3$ non-error informational messages are included.

`--compatible` $\langle n \rangle$ **or** `-compat` $\langle n \rangle$ Set the compatibility mode. The argument should be either a non-negative integer (0 for version 1.0, 1 for version 1.1, etc) or the keyword `latest` to indicate the latest version (default). Note that the compatibility mode only affects the available actions and the display style of the result, and does not change security features. For example, the check for the `openin_any` setting was only introduced to version 1.2, but this is still checked even if the compatibility mode is set to 0 or 1.

If multiple actions are given, they will be processed in the order specified in the command line invocation. Each result will be displayed on a separate line. As from v1.1, if there are multiple actions, each result will be grouped. This makes it easier to process the results in TEX. For example:

```
texosquery -l
```

This just produces (for me):

```
en\fhyn GB\fdot utf8
```

(which expands to `en-GB.utf8` when used with `\TeXOSQuery`) whereas

```
texosquery -l -o
```

produces:

```
{en\fhyn GB\fdot utf8}
{Linux}
```

To reproduce the v1.0 display use `--compatible 0`. (This will also explicitly use the punctuation characters rather than replacing them with the control sequence markup, such as `\fhyn` or `\fdot`.)

Note that unavailable information will produce an empty group. For example (assuming `nofile` doesn't exist or doesn't have read access):

```
texosquery -l -d nofile
```

produces:

```
{en\fhyn GB\fdot utf8}
{}
```

whereas

```
texosquery -d nofile
```

just displays an empty line.

If you're puzzled as to why an empty line has been returned, try rerunning the command with `--debug` for further information. Available actions are listed below.

Action `-b` or `--bcp47`

(New to version 1.2.) This action displays the BCP 47 language tag. For example, my locale is `en-GB` (English in the United Kingdom), so

```
texosquery -b
```

Simply returns:

```
en\fhyn GB
```

(which expands to `en-GB` with `\TeXOSQuery`) whereas a user whose default locale is set to Swiss German with the new orthography would get:

```
de\fhyn CH\fhyn 1996
```

(which expands to `de-CH-1996` with `\TeXOSQuery`).

Action `-L` or `--locale`

This action displays the locale information in the POSIX form

⟨*lang*⟩`-`⟨*region*⟩`.`⟨*codeset*⟩`@`⟨*modifier*⟩

where ⟨*lang*⟩ is the ISO code for the language (e.g. `en`), ⟨*region*⟩ is the ISO code for the region (e.g. `GB`), ⟨*codeset*⟩ is the default code set (e.g. `UTF-8`) and ⟨*modifier*⟩ is the modifier. Elements may be omitted if unavailable. For example, `en-GB.UTF-8` has the ⟨*modifier*⟩ omitted, and `en` has all but the language omitted. As above, the punctuation characters will actually be returned using the control sequences `\fhyn` (hyphen), `\fdot` (dot) and `\fatc` (at).

Action `-l` or `--locale-lcs`

This action is similar to `--locale`, but the codeset is converted to lower case and any hyphens are stripped. For example, if `--locale` returns `en-GB.UTF-8`, then `--locale-lcs` would return `en-GB.utf8`. As above, the punctuation characters will actually be returned using the control sequences `\fhyn` (hyphen), `\fdot` (dot) and `\fatc` (at).

Action `-C` **or** `--codeset-lcs`

(New to version 1.2.) This action returns just the codeset converted to lower case with hyphens stripped. For example, my default file encoding is UTF-8, so

```
texosquery -C
```

returns

```
utf8
```

This action is used by the locale package to determine the option to use when it needs to automatically load the inputenc package.

Action `-o` **or** `--osname`

This action displays the operating system name. For example, for me this produces:

```
Linux
```

Action `-r` **or** `--osversion`

This action displays the operating system version. For example, for me this produces:

```
4\fdot 1\fdot 13\fhyn 100\fdot fc21\fdot x86\fusc 64
```

(which expands to `4.1.13-100.fc21.x86_64` when used with `\TeXOSQuery`).

Action `-a` **or** `--osarch`

This action displays the operating system architecture. For example, for me this produces:

```
amd64
```

Action `-M` **or** `--date-time`

(New to version 1.2.) This action displays all the current date time data in a format suitable for use in `\texosqueryfmtdatetime`. (See section 3.1.4.)

Action `-Z [`⟨***locale***⟩`]` **or** `--time-zones [`⟨***locale***⟩`]`

(New to version 1.2.) This action displays all of the time zone mappings for the given locale (or the default if ⟨*locale*⟩ is omitted) in the format

`{{`⟨id_1⟩`}{`⟨*short name*⟩`}{`⟨*long name*⟩`}{`⟨*dst short name*⟩`}{`⟨*dst long name*⟩`}}...{{`⟨id_n⟩`}{`⟨*short name*⟩`}{`⟨*long name*⟩`}{`⟨*dst short name*⟩`}{`⟨*dst long name*⟩`}}`

The ⟨*id*⟩ is the unique label used by Java to identify the time zone (such as `Europe/London`) as used in the time zone information returned by `-M` (`--date-time`).

Action `-n` **or** `--pdfnow`

This action displays the current date and time in PDF format. For example

```
\pdfd \fcln 20160704131006\fpls 01\fapo 00\fapo
```

This uses the shorthand tags `\pdfd`, `\fcln`, `\fpls` and `\fapo` that are locally redefined by `\TeXOSQuery` to produce a `D`, a colon (`:`), a plus sign (`+`) and an apostrophe (`'`) with the category code set to 12 to make it consistent with `\pdfcreationdate`. This also allows for situations where the punctuation characters have been made active (for example, through babel).

Some, but not all, TEX formats provide `\pdfcreationdate`, which is more efficient than using the shell escape, but this can be used as a fallback method for those that don't (for example, XETEX).

Note that versions 1.0 and 1.1 didn't use `\pdfd` etc but simply used the actual characters. For example:

```
D:20160704131006+01'00'
```

If you want to reproduce this format, use `--compatible` with the level set to 0 or 1.

Action `-d` ⟨***file***⟩ **or** `--pdfdate` ⟨***file***⟩

This action displays the last modified time stamp of the given file in PDF format or a blank line if the file doesn't exist or the file permissions prohibit this action. Again some, but not all, TEX formats provide `\pdffilemoddate{`⟨*file*⟩`}`, which is more efficient than using the shell escape.

As with `--pdfnow` this now uses `\pdfd` etc which are converted by `\TeXOSQuery` to characters with the category code set to 12.

This action obeys the `openin_any` setting, so if access to ⟨*file*⟩ is forbidden by this setting, the result will be empty.

Action `-s` ⟨***file***⟩ **or** `--filesize` ⟨***file***⟩

This action displays the size in bytes of the given file or an empty string if the file doesn't exist or the file permissions prohibit this action. Some, but not all, TEX formats provide `\pdffilesize{`⟨*file*⟩`}`, which is more efficient than using the shell escape.

This action obeys the `openin_any` setting, so if access to ⟨*file*⟩ is forbidden by this setting, the result will be empty.

Action `-c` **or** `--cwd`

This action displays the current working directory. This obeys the `openin_any` setting, so this action will return an empty string if this file information is forbidden by that setting.

Action `-m` **or** `--userhome`

This action displays the user's home directory. This obeys the `openin_any` setting, so this action will return an empty string if this file information is forbidden by that setting.

Action `-t` **or** `--tmpdir`

This action displays the temporary directory. This obeys the `openin_any` setting, so this action will return an empty string if this file information is forbidden by that setting.

Action `-i` ⟨***sep***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]` **or** `--list` ⟨***sep***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]`

This action lists all files in the given directory with the output on a single line using ⟨*sep*⟩ as the separator between entries. Note that the list doesn't include the full path, just the file names.

> **Important Note:** As from v1.2, new restrictions have been placed on the value of ⟨*dir*⟩ for security reasons. For all three applications, `texosquery-jre8`, `texosquery` and `texosquery-jre5`, the `openin_any` setting is checked. If read access to ⟨*dir*⟩ is forbidden by the `openin_any` setting, then this action returns an empty string. *Additionally*, regardless of `openin_any`, the more restrictive applications, `texosquery-jre8` and `texosquery`, prohibit a value of ⟨*dir*⟩ that's outside the current working directory path (e.g. `..`) or that has no parent directory (e.g. `/`). Both `texosquery-jre8` and `texosquery` check the *canonical path* of ⟨*dir*⟩, so if ⟨*dir*⟩ is a symbolic link, the target path is checked.
>
> This is a security feature to prevent any malicious code that might try to recursively list the contents of the entire filing system, which would hog resources, or that might try to discover files outside the current working directory. An exception is made for `texosquery-jre5` since that application is already considered insecure (due to Java 5 and 6 being deprecated), so if you really need ⟨*dir*⟩ as, say `..` (the parent directory) or `/` (the root directory), you can use `texosquery-jre5` (by redefining `\TeXOSInvokerName` before using `\TeXOSQuery`) although this isn't recommended. It will still obey the `openin_any` setting, so the listing still won't work with `texosquery-jre5` if the `openin_any` setting is set to `p` (paranoid).

As from version 1.2, there is now an optional argument ⟨*sort*⟩, which indicates how the returned list should be sorted. If omitted `default` is assumed. Available values of ⟨*sort*⟩:

`default` Use the default order. This is typically in alphabetical order, but depends on the operating system or JRE.

`date-ascending` Order by file modified date from oldest to newest. This option has synonyms `date` and `date-asc`.

`date-descending` Order by file modified date from newest to oldest. You may use the shorter `date-des` value instead.

`size-ascending` Order by file size from smallest to largest. This option has synonyms `size` and `size-asc`.

`size-descending` Order by file size from largest to smallest. You may use the shorter `size-des` value instead.

`name-ascending` Order by file name (case-sensitive) alphabetically. This option has synonyms `name` and `name-asc`.

`name-descending` Order by file name (case-sensitive) in reverse alphabetic order. You may use the shorter `name-des` value instead.

`iname-ascending` Order by file name (case-insensitive) alphabetically. This option has synonyms `iname` and `iname-asc`.

`iname-descending` Order by file name (case-insensitive) in reverse alphabetic order. You may use the shorter `iname-des` value instead.

`ext-ascending` Order by file extension (case-sensitive) alphabetically. If files have the same extension, they are ordered by name. This option has synonyms `ext` and `ext-asc`.

`ext-descending` Order by file extension (case-sensitive) in reverse alphabetic order. If files have the same extension, they are ordered by name (reverse alphabetic order). You may use the shorter `ext-des` value instead.

This action obeys the `openin_any` setting for all the listed files as well as for the directory ⟨*dir*⟩, so if access to a file in the directory is forbidden, the file will be omitted from the list. (This action is equivalent to the following with ⟨*regex*⟩ set to `.*` to match all files.)

If you want to excluded hidden dot files (where they aren't automatically excluded by `openin_any`), use the `--filterlist` action described below with ⟨*regex*⟩ set to `[^\.].*`. (Remember that you'll need to use `\string` when using the shell escape, as noted below.)

> **Important Note:** Unlike most of the return values the ⟨*sep*⟩ part here isn't escaped, so take care if ⟨*sep*⟩ contains any commands. For example, if you want to use `\\` as the separator, you'll need to use `\string\noexpand\string\\` in the ⟨*sep*⟩ part within `\TeXOSQuery`.

For example:

```
\TeXOSQueryFileList{\result}{\string\noexpand\string\\}{.}
```

calls (through the shell escape):

```
texosquery -i '\noexpand\\' '.'
```

(the two `\string` commands have detokenized their arguments) so `texosquery` uses `\noexpand\\` as the separator in the returned list, but this list is expanded as it's read in. However `\noexpand` prevents the `\\` from being expanded, so the separator becomes just `\\` which may be (re)defined before the resulting list is processed.

Note that `\TeXOSQueryFileList` automatically adds the single quotes around the arguments. If `\TeXOSQuery` is used explicitly, these quotes would need to be added as appropriate.

Action `-id` ⟨***sep***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]` **or** `--list-dir` ⟨***sep***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]`

This action is like `--list` but only includes sub-directories of ⟨*dir*⟩. The caveats and security notes for `--list` also apply here.

Action `-ir` ⟨***sep***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]` **or** `--list-regular` ⟨***sep***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]`

This action is like `--list` but only includes regular files. The caveats and security notes for `--list` also apply here.

Action `-f` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]` **or** `--filterlist` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]`

This action is like `--list` but only lists those files whose name matches the regular expression given in ⟨*regex*⟩. Note that since this uses Java's `String.matches` method this tests for a *complete* match on the file name (not including directory path). For example, if ⟨*regexp*⟩ is `foo.*`, it will only match files whose name starts with `foo` (for example, `foobar` will match but `barfoo` won't). Use `.*foo.*` to match all files that contain `foo` in the name (so `foobar` and `barfoo` will both match).

> **Important Note:** You can't have an empty regular expression. You can use the regular expression `.*` to match all files (which is what `--list` does).

As from version 1.2, this action now has an optional argument ⟨*sort*⟩, which indicates how to sort the returned list. The available values for ⟨*sort*⟩ are the same as for `--list`, described above.

The caveats and security notes for `--list` also apply here.

Action `-fd` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]` **or** `--filterlist-dir` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]`

This action is like `--filterlist` but only includes sub-directories of ⟨*dir*⟩.

The caveats and security notes for `--list` also apply here.

Action `-fr` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]` **or** `--filterlist-regular` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]`

This action is like `--filterlist` but only includes regular files.

The caveats and security notes for `--list` also apply here.

Action `-w` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]` **or** `--walk` ⟨***sep***⟩ ⟨***regex***⟩ ⟨***dir***⟩ `[`⟨***sort***⟩`]`

(New to version 1.2. Not available with `texosquery-jre5`.)

This action starts from the directory ⟨*dir*⟩ *which must be on the current working directory's path* and returns a list separated by ⟨*sep*⟩ of all the regular files whose basename matches the regular expression ⟨*regex*⟩ (as for the filtered file listings described above), recursively descending sub-directories. Any files or sub-directories that are hidden, unreadable or symbolic links are skipped. The list is sorted according to ⟨*sort*⟩, which is as for the file listing actions described above. Note that ⟨*dir*⟩ is first converted to its canonical

path, so if ⟨*dir*⟩ is a symbolic link, the security check will test if the *target* path is on the current working directory path.

As with the above file listings, the separator ⟨*sep*⟩ isn't escaped so take care if ⟨*sep*⟩ contains any commands. The resulting list will consist of paths relative to ⟨*dir*⟩.

> **Important Note:** This action requires at least Java 7 so it's not available with `texosquery-jre5.jar`.

Action `-u` ⟨***file***⟩ **or** `--uri` ⟨***file***⟩

This action displays the URI of the given file or an empty string if the file doesn't exist or if the file permissions or the `openin_any` setting prohibit read access.

Action `-p` ⟨***file***⟩ **or** `--path` ⟨***file***⟩

This action displays the canonical path of the given file or an empty string if the file doesn't exist or if the file permissions or the `openin_any` setting prohibit this action.

Action `-e` ⟨***file***⟩ **or** `--dirname` ⟨***file***⟩

(New to v1.1.) This action displays the canonical path of the given file's parent (that is, the directory containing ⟨*file*⟩) or an empty string if the file doesn't exist or if the file permissions or the `openin_any` setting prohibits this action. Note that this is different to the Unix-like `dirname` command, which will return a relative path if ⟨*file*⟩ isn't an absolute path.

Action `-N [`⟨***language tag***⟩`]` **or** `--numeric [`⟨***language tag***⟩`]`

(New to v1.2.) This action displays:

`{`⟨*locale tag*⟩`}{`⟨*group sep*⟩`}{`⟨*decimal sep*⟩`}{`⟨*exp sep*⟩`}{`⟨*use group*⟩`}{`⟨*currency code*⟩`}{`⟨*regional currency code*⟩`}{`⟨*currency sym*⟩`}{`⟨*TEX currency*⟩`}{`⟨*currency sep*⟩`}`

for the ⟨*language tag*⟩ given in the optional argument. If omitted, the default locale is assumed. The returned values are:

- ⟨*tag*⟩ the language tag.
- ⟨*group sep*⟩ the numeric group separator.
- ⟨*decimal sep*⟩ the decimal separator.
- ⟨*exp sep*⟩ the exponent separator.
- ⟨*use group*⟩ `1` if the locale uses number grouping otherwise `0`.
- ⟨*currency code*⟩ the ISO 4217 currency code.
- ⟨*regional currency code*⟩ either the ISO 4217 currency code or an unofficial code. The only unofficial codes returned are: GGP (Guernsey pound), JEP (Jersey pound), IMP (Isle of Man pound), KID (Kiribati dollar) and TVD (Tuvaluan dollar).

- ⟨*currency sym*⟩ the currency symbol. (This may sometimes be the same as ⟨*currency code*⟩.) Non-ASCII characters will be marked up with `\twrp` (see below).
- ⟨*TEX currency*⟩ the currency symbol using TEX code provided by texosquery. This is obtained by substituting known Unicode currency symbols occurring in ⟨*currency sym*⟩ with `\texosquerycurrency{`⟨*xxx*⟩`}`, which expands to the control sequence given by the name `texosquerycurrency`⟨*xxx*⟩. These commands are defined in `texosquery.tex`. Since there are no generic TEX commands available for all these symbols (except $), these commands will need to be redefined as appropriate but are provided in the event that there's no UTF-8 support. There is a limited check for some known currency commands, such as `\texteuro` or `\euro`, but if an appropriate currency command can't be found, the `\texosquerycurrency`⟨*xxx*⟩ commands will be defined to simply the currency label (usually the same as the ⟨*xxx*⟩ part).
- ⟨*currency sep*⟩ the currency decimal separator.

The language tag should conform to IETF BCP 47. See http://docs.oracle.com/javase/8/docs/api/java/util/Locale.html for further details. If you are using `texosquery-jre5`, only the language, region and variant elements will be recognised since the language tag support was introduced in Java 7.

For example:

```
texosquery -N en-GB
```

produces

```
{en-GB}{,}{.}{E}{1}{GBP}{GBP}{\twrp{£}}{\texosquerycurrency{pound}}{.}
```

The `\twrp` command is used by `texosquery` to markup a non-ASCII character. This command is one of the shorthands only defined within `\TeXOSQuery`. In this case it's a shortcut for the command `\texosquerynonasciiwrap`. By default this just does its argument, but it may be redefined to perform some other action such as converting from one encoding to another.

In most cases the ⟨*regional currency code*⟩ will be the same as ⟨*currency*⟩ code. A few non-ambiguous unofficial codes are known by `texosquery` and may be used if the country code is recognised. For example,

```
texosquery -N en-IM
```

produces

```
{en-IM}{,}{.}{E}{1}{GBP}{IMP}{M\twrp{£}}{M\texosquerycurrency{pound}}{.}
```

If Java doesn't support the given locale, the currency code will appear as `XXX` with the symbol ¤ (generic currency sign).

> **Important Note:** This option and the following (`--locale-data`) are best used with XETEX or LuaTEX to deal with the non-ASCII characters. Make sure the file encoding used by Java matches the TEX file.

(See section 1.6 for the difference in locale providers.)

Action `-D [`⟨***language tag***⟩`]` **or** `--locale-data [`⟨***language tag***⟩`]`

(New to v1.2.) This action provides more extensive information than `--numeric`. The result has nested groups to assist parsing. Again the ⟨*language tag*⟩ may be omitted. For example,

```
texosquery --locale-data
```

For the default locale or

```
texosquery --locale-data en-GB
```

for the locale identified by `en-GB`. As with all the other actions, the result is written to STDOUT on a single line. Its overall length and the use of the shortcut commands used by `texosquery` to markup certain elements mean that it's not particularly human-readable, but it's designed to be easy for TEX to interpret. The information is returned in the following format:

`{`⟨*locale block*⟩`}{`⟨*current date block*⟩`}{`⟨*date pattern block*⟩`}{`⟨*current time block*⟩`}{`⟨*time pattern block*⟩`}{`⟨*current date time block*⟩`}{`⟨*date time pattern block*⟩`}{`⟨*days of the week block*⟩`}{`⟨*abbreviated dates of the week block*⟩`}{`⟨*month names block*⟩`}{`⟨*abbreviated month names block*⟩`}{`⟨*standalone days of the week block*⟩`}{`⟨*abbreviated standalone days of the week block*⟩`}{`⟨*standalone month names block*⟩`}{`⟨*abbreviated standalone month names block*⟩`}{`⟨*numeric block*⟩`}{`⟨*numeric patterns block*⟩`}`

There may seem to be some repetition here with the month and week day names, but with `texosquery-jre8`, the second set are the standalone version (for example, for a column header). In some languages, these may be different from the names used in the date format. Since this is new to Java 8, it's not supported in `texosquery.jar` or `texosquery-jre5.jar` and they simply reproduce the non-standalone names.

The information supplied with this option is quite complex, but it's used by the locale package to set up all the required information for each locale used in the document. Any non-ASCII characters are marked up with `\twrp`, which is locally defined by `\TeXOSQuery` to expand to `\texosquerynonasciiwrap`. This may be redefined to deal with the characters if necessary. For example, if the character needs to be converted from one encoding to another.

The blocks are:

⟨*locale block*⟩

The locale information in the form:

`{`⟨*tag*⟩`}{`⟨*language name*⟩`}{`⟨*locale language name*⟩`}{`⟨*region name*⟩`}{`⟨*locale region name*⟩`}{`⟨*variant name*⟩`}{`⟨*locale variant name*⟩`}`

The ⟨*tag*⟩ is the language tag (the same format as `--bcp47`).

The ⟨*language name*⟩ is the language name in the operating system's default locale.

The ⟨*locale language name*⟩ is the language name in the locale's language.

For example, my locale is `en-GB`, so if I use

```
texosquery -D en-GB
```

then both ⟨*language name*⟩ and ⟨*locale language name*⟩ will be `English`, but if I use:

```
texosquery -D fr-GB
```

then ⟨*language name*⟩ will be `French` and the ⟨*locale language name*⟩ will be `fran\twrp{ç}ais` (note the non-ASCII character has been marked up). The locale tag `fr-GB` indicates that I'm writing in French but I'm in the United Kingdom (so the currency should be GBP).

The ⟨*region name*⟩ is the region's name in the operating system's default language.

The ⟨*locale region name*⟩ is the region's name in the locale's language.

So for me with `-D en-GB` I get `United\tspc Kingdom` for both ⟨*region name*⟩ and ⟨*locale region name*⟩. This illustrates another of the shorthand commands that `texosquery` uses that's only locally defined within `\TeXOSQuery`. In this case, `\tspc` just expands to a space. This is used to avoid accidentally discarding any intentional spaces that might follow a command name or any intentional consecutive spaces.

If, however, I use `-D fr-GB` I still get `United\tspc Kingdom` in ⟨*region name*⟩, but ⟨*locale region name*⟩ is now `Royaume-Uni`.

The ⟨*variant name*⟩ is the language's variant. For example, with `de-CH-1996` (Swiss German using the new orthography), the variant is 1996. There's no variant in `en-GB` so this value is empty for me.

The ⟨*locale variant name*⟩ is the variant in the locale's language. In the case of `de-CH-1996` this is still 1996.

⟨*current date block*⟩

This returns the current date in four different formats and also an integer that indicates the first day of the week in the given locale:

`{`⟨*full date*⟩`}{`⟨*long date*⟩`}{`⟨*medium date*⟩`}{`⟨*short date*⟩`}{`⟨*first day*⟩`}`

The actual date formats depend on the locale. For example, with `en-GB` the ⟨*full date*⟩ is (assuming today is 2016-11-08):

```
Tuesday,\tspc 8\tspc November\tspc 2016
```

(Tuesday, 8 November 2016). The ⟨*long date*⟩ is

```
08\tspc November\tspc 2016
```

(08 November 2016). The ⟨*medium date*⟩ is

```
08\thyn Nov\thyn 2016
```

(08-Nov-2016). The ⟨*short date*⟩ is

```
08\tslh 11\tslh 16
```

(08/11/2016). Note that there's a difference between using the CLDR locale data and the JRE data. If I'm using the Java 7 compatible `texosquery.jar` which only uses JRE locale data, then I get the above results, but if I use `texosquery-jre8.sh` which sets `java.locale.providers` to `CLDR,JRE` then I get a slightly different result. The ⟨*long date*⟩ is

```
8\tspc November\tspc 2016
```

(8 November 2016) and the ⟨*medium date*⟩ is

```
8\tspc Nov\tspc 2016
```

(8 Nov 2016). The ⟨*medium date*⟩ may be numeric or may be an abbreviated form of ⟨*long date*⟩, depending on the language and the locale provider. Some languages aren't supported by JRE but are supported by CLDR. (Some aren't supported by either, but there's a chance that those languages will eventually be added to the CLDR.) For example, if I use `-D cy-GB` with the JRE as the locale provider I just get the `en-GB` dates, but if I use the CLDR provider I get Welsh dates.

Note that the proposed Java 9 should automatically use the CLDR as the locale provider, which is being increasingly adopted by applications as a common data repository.

The first day of the week index is zero-based starting with Monday. This is done in order to be compatible with pgfcalendar. For example, with `-D en-GB` ⟨*first day*⟩ is 0 (Monday), but with `pt-BR` ⟨*first day*⟩ is 6 (Sunday). The locale package provides a way of converting the index to Monday=1 or Sunday=1 indexing.

⟨*date pattern block*⟩

The pattern used to format the full date, long date, medium date and short date. This is in the form:

`{`⟨*full pattern*⟩`}{`⟨*long pattern*⟩`}{`⟨*medium pattern*⟩`}{`⟨*short pattern*⟩`}`

Each pattern uses shorthand mark-up that's only locally defined within `\TeXOSQuery`. These short commands are expanded to longer commands provided by `texosquery.tex` to avoid name clashing with other packages. When used directly in the document text, these expand to reproduce the pattern.

For example, with `-D en-GB` I get the following pattern for the short date:

```
\patdtf{2}{d}\tslh \patdtf{2}{M}\tslh \patdtf{2}{y}
```

When parsed by `\TeXOSQuery`, this is internally converted to

```
\texosquerydtf{2}{d}/\texosquerydtf{2}{M}/\texosquerydtf{2}{y}
```

But default this simply expands to `dd/MM/yy` but may be used in the first argument of `\texosqueryfmtdatetime`. See sections 3.1.3 and 3.1.4 for further details.

⟨*current time block*⟩

The current time provided in various formats suitable to the given locale:

{⟨*full time*⟩}{⟨*long time*⟩}{⟨*medium time*⟩}{⟨*short time*⟩}

As with the current date, the actual format depends on the locale and the locale provider. For example, with `en-GB` I get:

```
{15:59:41\tspc o\csq clock\tspc GMT}{15:59:41\tspc GMT}{15:59:41}{15:59}
```

with the JRE. If I switch to CLDR (`texosquery-jre8.sh`) I get:

```
{16:00:51\tspc Greenwich\tspc Mean\tspc Time}{16:00:51\tspc GMT}{16:00:51}{16:00}
```

⟨*time pattern block*⟩

The pattern used to format the full time, long time, medium time and short time.

{⟨*full time pattern*⟩}{⟨*long time pattern*⟩}{⟨*medium time pattern*⟩}{⟨*short time pattern*⟩}

Again, when used with `\TeXOSQuery`, the short commands, such as `\patdtf`, are internally converted. They're not defined outside that scope.

⟨*current date time block*⟩

The current date and time provided in various formats suitable to the given locale:

{⟨*full date time*⟩}{⟨*long date time*⟩}{⟨*medium date time*⟩}{⟨*short date time*⟩}

This may simply be the date and time from above separated by a space.

⟨*date time pattern block*⟩

The pattern used to format the full date time, long date time, medium date time and short date time. This may simply be the date and time patterns from above separated by a space.

{⟨*full date time pattern*⟩}{⟨*long date time pattern*⟩}{⟨*medium date time pattern*⟩}{⟨*short date time pattern*⟩}

⟨*days of the week block*⟩

The week day names (starting with Monday for consistency with pgfcalendar) in the locale's language. Non-ASCII characters are marked up with `\twrp`.

{⟨*Monday*⟩}{⟨*Tuesday*⟩}{⟨*Wednesday*⟩}{⟨*Thursday*⟩}{⟨*Friday*⟩}{⟨*Saturday*⟩}{⟨*Sunday*⟩}

⟨*abbreviated days of the week block*⟩

As above, but abbreviated.

⟨*month names block*⟩

The month names.

⟨*abbreviated month names block*⟩

The abbreviated month names.

⟨*standalone days of the week block*⟩

The week day names when used in a standalone context (for example, a column header). This may be the same as the earlier ⟨*days of the week block*⟩ (and will be the same for `texosquery.jar` and `texosquery-jre5.jar`). The standalone support was introduced to Java 8.

⟨*abbreviated standalone days of the week block*⟩

As above, but abbreviated.

⟨*standalone month names block*⟩

The month names when used in a standalone context (for example, a column header). This may be the same as the earlier ⟨*month names block*⟩ (and will be the same for `texosquery.jar` and `texosquery-jre5.jar`). The standalone support was introduced to Java 8.

⟨*abbreviated standalone month names block*⟩

As above but abbreviated.

⟨*numeric block*⟩

The numeric data similar to `--numeric` but it's missing the ⟨*tag*⟩ (which is provided in the earlier ⟨*locale block*⟩) and there are two extra items:

{⟨*group sep*⟩}{⟨*decimal sep*⟩}{⟨*exp sep*⟩}{⟨*use group*⟩}{⟨*currency code*⟩}{⟨*regional currency code*⟩}{⟨*currency sym*⟩}{⟨*currency tex*⟩}{⟨*currency sep*⟩} {⟨*percent sym*⟩}{⟨*per mill sym*⟩}

See above for the elements that are also provided in `--numeric`. The additional elements are ⟨*percent sym*⟩ and ⟨*per mill sym*⟩, which are the percent and per-mill symbols, respectively. The percent symbol % has its category code changed to 12 by `\TeXOSQuery`. As with other non-ASCII characters, the per-mill symbol will be marked up with `\twrp`.

⟨*numeric patterns block*⟩

The patterns used to format decimals, integers, currency and percentages.

`{`⟨*decimal pattern*⟩`}{`⟨*integer pattern*⟩`}{`⟨*currency pattern*⟩`}{`⟨*percentages pattern*⟩`}`

As with the date and time patterns, when used with `\TeXOSQuery`, the short commands, such as `\patdgt`, are internally converted. They're not defined outside that scope.

If a pattern is used directly in the text, it will expand to the original pattern padded to ten digits. (Eleven digit integers are outside TEX's maximum number range.)

Any of these numeric patterns may be used in the first argument of the low-level user command `\texosqueryfmtnumber` described in section 3.1.5. This command uses the following macros:

`\texosquerypatfmtcurrencysign`

The currency sign (defaults to `\$`). For example, when parsing the previous ⟨*numeric block*⟩, this command can be redefined to the `{`⟨*currency sym*⟩`}` or `{`⟨*currency tex*⟩`}` elements.

`\texosquerypatfmtgroupsep`

The group separator (defaults to `,`). For example, when parsing the previous ⟨*numeric block*⟩, this command can be redefined to the `{`⟨*group sep*⟩`}` element.

`\texosquerypatfmtdecsep`

The decimal separator (defaults to `.`). For example, when parsing the previous ⟨*numeric block*⟩, this command can be redefined to the `{`⟨*dec sep*⟩`}` element.

`\texosquerypatfmtcurdecsep`

The monetary decimal separator (defaults to `.`). For example, when parsing the previous ⟨*numeric block*⟩, this command can be redefined to the `{`⟨*currency sep*⟩`}` element.

`\texosquerypatfmtexp`

The exponent sign (defaults to `E`). For example, when parsing the previous ⟨*numeric block*⟩, this command can be redefined to the `{`⟨*exp sep*⟩`}` element.

`\texosquerypatfmtpercentsign`

The percent symbol. For example, when parsing the previous ⟨*numeric block*⟩, this command can be redefined to the `{`⟨*percent sym*⟩`}` element.

`\texosquerypatfmtpermillsign`

The per-mill symbol. For example, when parsing the previous ⟨*numeric block*⟩, this command can be redefined to the `{`⟨*per-mill sym*⟩`}` element.

`\texosquerypatfmticurrencysign`

The international currency sign. This is defined as ¤ by default, unless the command `\textcurrency` has been defined, in which case that's used instead. If the UTF-8 character ¤ is available and `\textcurrency` hasn't been defined before texosquery was input, then you will need to redefine this command as appropriate.

`\texosquerypatfmtminus`

The minus sign.

`\texosquerypatfmtplus`

The plus sign.

2 texosquery.tex: generic TEX code

You can run `texosquery` directly from TEX's shell escape. For example:

```
\input|"texosquery --locale"
```

However, `texosquery` uses markup commands in some of the results which need to be defined first. The file `texosquery.tex` provides generic TEX code to do this for you and stores the result in a control sequence.

Plain TEX users can input this file through the usual `\input` method:

```
\input texosquery
```

LATEX users may also simply input this file:

```
\input{texosquery}
```

but may prefer the standard package approach:

```
\usepackage{texosquery}
```

Important Note: The commands described below are all fragile.

`\TeXOSQuery`

The basic command to run `texosquery` and capture its output in a control sequence is:

`\TeXOSQuery{`⟨*cs*⟩`}{`⟨*args*⟩`}`

where ⟨*cs*⟩ is the control sequence in which to store the result and ⟨*args*⟩ are the command line arguments to pass to `texosquery`. This first locally changes the category code of some problematic characters and defines the short markup commands that `texosquery` uses to identify characters that need to be interpreted literally (for example, in file names). These commands will automatically be expanded by `\TeXOSQuery` when the result is input. For example

```
texosquery -n
```

produces

```
\pdfd \fcln 20161129221559\fpls 00\fapo 00\fapo
```

but when used with

```
\TeXOSQuery{\result}{texosquery -n}
```

the `\result` command will be set to

```
D:20161129221559+00'00'
```

where the characters `D` `:` `+` and `'` all have category code 12 (other).

If the command failed, ⟨*cs*⟩ will be set to empty. It's best to always test for success after using `\TeXOSQuery` (or one of the shortcut commands described below). For example:

```
\TeXOSQuery{\result}{-b}
\ifx\result\empty
 Failed!
\else
 Result: \result.
\fi
```

Failure can occur because the dry run mode was on, or it can occur if the query was denied (for example, forbidden file access), or if there's a syntax error in the system call. As from version 1.2, `texosquery` checks the `openin_any` setting, which may forbid read access. Java's security manager or the filing system may also forbid read access.

To determine the cause of the error, first inspect the log file to check if the shell escape was used. In the above example, if the shell escape was permitted, then the log file should include

```
(|texosquery -b)
```

Copy and paste the system call (`texosquery -b` in the above case) into your command prompt or terminal and insert the `--debug` switch at the start of the argument list. For example:

```
texosquery --debug -b
```

This should help determine whether it's a syntax error or a query forbidden by the operating system.

If multiple queries are required, it's more efficient to perform them all in one go. For example:

```
\TeXOSQuery{\result}{-l -n -o}

\def\parseresult#1#2#3{%
  Locale: #1. Now: #2. OS: #3.%
}

\ifx\result\empty
  Query failed.
\else
  \expandafter\parseresult\result
\fi
```

(Make sure you have at least v1.1 for this to work correctly.)

> **Important Note:** Take care of characters that have a special meaning to your shell. For example, bash interprets # as a comment. For example, if you have a file called `image#1.png`, then you can't simply do
>
> ```
> \TeXOSQuery{\result}{-p image#1.png}
> ```
>
> since bash will pass this as
>
> ```
> texosquery.sh -p image
> ```
>
> (The `#1.png` part is treated as a comment.) Nor can you do
>
> ```
> \TeXOSQuery{\result}{-p image\#1.png}
> ```
>
> as TEX will replace the `\#` with # when passing the command invocation to the shell. The only way to deal with this situation is to do
>
> ```
> \TeXOSQuery{\result}{-p image\string\#1.png}
> ```
>
> to protect the # character from both TEX and the shell.

`\ifTeXOSQueryDryRun` Dry run mode is determined by the conditional

```
\ifTeXOSQueryDryRun
```

If true, the shell escape won't be used and the requested command invocation will be printed in the transcript file prefixed with

```
TeXOSQuery:
```

(the control sequence ⟨*cs*⟩ will be set to empty).

> **Important Note:** Remember that a query can still fail even if the dry run mode is off.

Note that if you switch off the dry run mode when the shell escape setting forbids the execution of `texosquery`, then you'll get the rather annoying error:

```
runpopen command not allowed: texosquery

! I can't find file `"|texosquery -b"'.
\TeXOSQueryInvoker ...TeXOSInvokerName \space #1"

\TeXOSQuery ...noexpand #1{\TeXOSQueryInvoker {#2}
                                                    }}\x \fi
1.5 \TeXOSQuery{\result}{-b}
                                ^^M
(Press Enter to retry, or Control-D to exit)
```

By default, the dry run mode is only switched off if the unrestricted shell escape mode is on (detected through `\shellescape` or `\pdfshellescape`).

If `texosquery` is added to the restricted list, you can add

```
\TeXOSQueryAllowRestricted
```

to the `texosquery.cfg` file. (This command can't be used outside of that file.)

If you get the above error, then:

- make sure you don't have `\TeXOSQueryAllowRestricted` in your `texosquery.cfg` file;
- make sure you run TEX with the shell escape enabled;
- check the definition of `\TeXOSInvokerName`;
- try using the application directly from the command prompt or terminal. For example, in the above message, the bit between `` `"| `` and `"'` (that is, `texosquery-jre8 -b`) shows the attempted system call. Copy and paste it directly into your operating system's command prompt or terminal and to check the application has been installed correctly.

`\TeXOSInvokerName`

The `\TeXOSQuery` command uses `\TeXOSInvokerName` to reference the application name. This defaults to `texosquery` but needs to be redefined to reflect the particular system call that's required. For example, `texosquery.sh` (Unix-like with Java 7) or `texosquery-jre8` (Windows with Java 8). This redefinition can be done in the configuration file `texosquery.cfg` for a system-wide setting. See section 1.1 for further details.

> **Important Note:** Some of the shortcut commands listed below require extra arguments after the relevant switch. These are automatically enclosed in single-quotes to protect any spaces. If the argument actually contains any single-quote characters, make sure you use `\string\'` to prevent interference.
>
> Since a file name reference may need to be obtained from `\jobname`, which sometimes includes double-quotes, the first double-quote pair found

is stripped in file name arguments. Any other double-quotes will need to be protected in the same manner as single-quotes (but this shouldn't be an issue if you use a safe file naming scheme). All paths should use a forward slash for the directory divider.

2.1 Locale

\TeXOSQueryLocale The locale (`-l` or `--locale-lcs`) information can be obtained using:

`\TeXOSQueryLocale{`⟨*cs*⟩`}`

Note that this uses the lower case codeset form, which has a better chance of matching the encoding names used by the inputenc package. If you want the unprocessed codeset name, you can do:

`\TeXOSQuery{`⟨*cs*⟩`}{-L}`

If you just want the codeset in the same form as `--locale-lcs` you can do:

`\TeXOSQuery{`⟨*cs*⟩`}{-C}`

\TeXOSQueryLangTag The IETF BCP 47 language tag (`-b` or `--bcp47`) can be obtained using:

`\TeXOSQueryLangTag{`⟨*cs*⟩`}`

\TeXOSQueryNumeric The numeric separators and currency symbols (`-N` or `--numeric`) can be obtained using

`\TeXOSQueryNumeric{`⟨*cs*⟩`}{`⟨*locale*⟩`}`

The ⟨*locale*⟩ should be a valid language tag or may be empty for the system's default locale. Similarly for the command below.

\TeXOSQueryLocaleData All the locale data (`-D` or `--locale-data`) can be obtained using

`\TeXOSQueryLocaleData{`⟨*cs*⟩`}{`⟨*locale*⟩`}`

2.2 Operating System Information

\TeXOSQueryName The OS name (`-o` or `--osname`) can be obtained using:

`\TeXOSQueryName{`⟨*cs*⟩`}`

\TeXOSQueryVersion The OS version (`-r` or `--osversion`) can be obtained using:

`\TeXOSQueryVersion{`⟨*cs*⟩`}`

\TeXOSQueryArch The OS architecture (`-a` or `--osarch`) can be obtained using:

`\TeXOSQueryArch{`⟨*cs*⟩`}`

2.3 Dates and Times

\TeXOSQueryDateTime

The current date and time information (`-M` or `--date-time`) can be obtained using:

`\TeXOSQueryDateTime{`⟨*cs*⟩`}`

Example usage:

```
\texosquerydefpattern{\pattern}{\%2d/\%2M/\%4y \%2H:\%2m:\%2s}

\TeXOSQueryDateTime{\datetimedata}

\ifx\datetimedata\empty
 Query Failed!
\else
 \expandafter\texosqueryfmtdatetime\expandafter\pattern\datetimedata
\fi
```

Note that commands such as `\texosqueryfmtpatMMM` will need to be defined to produce textual elements. See sections 3.1.3 and 3.1.4 for further details.

\TeXOSQueryTimeZones

The time zone mappings (`-Z` or `--time-zones`) can be obtained using:

`\TeXOSQueryTimeZones{`⟨*cs*⟩`}{`⟨*locale*⟩`}`

Leave ⟨*locale*⟩ empty if the default locale is required.

\TeXOSQueryNow

The current date-time stamp in PDF format (`-n` or `--pdfnow`) can be obtained using:

`\TeXOSQueryNow{`⟨*cs*⟩`}`

This is provided for the benefit of users who don't have `\pdfcreationdate` defined by their TEX format (for example, XETEX).

\TeXOSQueryFileDate

The modification date-time stamp in PDF format for a file (`-d` or `--pdfdate`) can be obtained using:

`\TeXOSQueryFileDate{`⟨*cs*⟩`}{`⟨*filename*⟩`}`

where ⟨*filename*⟩ is the name of the file. This is provided for the benefit of users who don't have `\pdffilemoddate` defined by their TEX format.

2.4 File Operations

\TeXOSQueryCwd

The current working directory (`-c` or `--cwd`) can be obtained using:

`\TeXOSQueryCwd{`⟨*cs*⟩`}`

\TeXOSQueryHome

The home directory (`-m` or `--userhome`) can be obtained using:

`\TeXOSQueryHome{`⟨*cs*⟩`}`

\TeXOSQueryTmpDir

The temporary directory (`-t` or `--tmpdir`) can be obtained using:

`\TeXOSQueryTmpDir{`⟨*cs*⟩`}`

\TeXOSQueryFileSize The size in bytes of a file (`-s` or `--filesize`) can be obtained using:

`\TeXOSQueryFileSize{`⟨*cs*⟩`}{`⟨*filename*⟩`}`

where ⟨*filename*⟩ is the name of the file. This is provided for the benefit of users who don't have `\pdffilesize` defined by their TEX format.

\TeXOSQueryFileURI The URI of a file (`-u` or `--uri`) can be obtained using:

`\TeXOSQueryFileURI{`⟨*cs*⟩`}{`⟨*filename*⟩`}`

where ⟨*filename*⟩ is the name of the file. (Any percent symbols `%` contained in the URI will have their category code set to 12.)

\TeXOSQueryFilePath The canonical path of a file (`-p` or `--path`) can be obtained using:

`\TeXOSQueryFilePath{`⟨*cs*⟩`}{`⟨*filename*⟩`}`

where ⟨*filename*⟩ is the name of the file.

\TeXOSQueryDirName The canonical path of a file's parent (`-e` or `--dirname`) can be obtained using:

`\TeXOSQueryDirName{`⟨*cs*⟩`}{`⟨*filename*⟩`}`

where ⟨*filename*⟩ is the name of the file.

\TeXOSQueryFileList The list of files in a given directory (`-i` or `--list`) can be obtained using:

`\TeXOSQueryFileList{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*dir*⟩`}`

where ⟨*sep*⟩ is the separator and ⟨*dir*⟩ is the directory name. For example:

```
\TeXOSQueryFileList{\result}{,}{.}
```

will store a comma-separated list of all the files contained in the current directory in the control sequence `\result`.

\TeXOSQueryRegularFileList To omit directories (`-ir` or `--list-regular`):

`\TeXOSQueryRegularFileList{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*dir*⟩`}`

\TeXOSQuerySubDirFileList To omit regular files (`-id` or `--list-dir`):

`\TeXOSQuerySubDirList{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*dir*⟩`}`

\TeXOSQueryFilterFileList A filtered list of files in a given directory (`-f` or `--filterlist`) can be obtained using:

`\TeXOSQueryFilterFileList{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*regex*⟩`}{`⟨*dir*⟩`}`

where ⟨*regex*⟩ is a regular expression. *Take care of any backslashes in the regular expression!* For example, to list only those files that have an extension:

```
\TeXOSQueryFilterFileList{\result}{,}{.+\string\..*}{.}
```

Note the use of `\string\.` to ensure that `\.` isn't interpreted as a command. Another example, list only `.png` and `.jpg` files in the directory called `images`:

```
\TeXOSQueryFilterFileList{\result}{,}{.+\string\.(jpg|png)}{images}
```

Important Note: Unlike most of the return values the ⟨*sep*⟩ part here isn't escaped, so take care if ⟨*sep*⟩ contains any commands. For example, if you want to use \\ as the separator, you'll need to use `\string\noexpand\string\\` in the ⟨*sep*⟩ part.

```
\TeXOSQueryFilterFileList{\result}{\string\noexpand\string\\}{.*\string\.tex}{.}
```

If you want the list sorted, you can use the following which set the optional ⟨*sort*⟩ argument.

`\TeXOSQueryFileListDateAsc` Order by last modified date starting with the oldest (`date-ascending`):

```
\TeXOSQueryFileListDateAsc{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

`\TeXOSQueryRegularFileListDateAsc` or the regular files only list:

```
\TeXOSQueryRegularFileListDateAsc{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

`\TeXOSQuerySubDirListDateAsc` or the sub-directories only list:

```
\TeXOSQuerySubDirListDateAsc{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

`\TeXOSQueryFilterFileListDateAsc` or for the filtered list:

```
\TeXOSQueryFilterFileListDateAsc{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

`\TeXOSQueryFilterRegularFileListDateAsc` or the filtered regular files only list:

```
\TeXOSQueryFilterRegularFileListDateAsc{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

`\TeXOSQueryFilterSubDirListDateAsc` or the filtered sub-directories only list:

```
\TeXOSQueryFilterSubDirListDateAsc{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

`\TeXOSQueryFileListDateDes` Order by last modified date starting with the newest (`date-descending`):

```
\TeXOSQueryFileListDateDes{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

`\TeXOSQueryRegularFileListDateDes` or the regular files only list:

```
\TeXOSQueryRegularFileListDateDes{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

`\TeXOSQuerySubDirListDateDes` or the sub-directories only list:

```
\TeXOSQuerySubDirListDateDes{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

`\TeXOSQueryFilterFileListDateDes` or for the filtered list:

```
\TeXOSQueryFilterFileListDateDes{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

`\TeXOSQueryFilterRegularFileListDateDes` or the filtered regular files only list:

```
\TeXOSQueryFilterRegularFileListDateDes{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

`\TeXOSQueryFilterSubDirListDateDes` or the filtered sub-directories only list:

\TeXOSQueryFilterRegularFileListDateDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFileListSizeAsc Order by file size starting with the smallest (`size-ascending`):

\TeXOSQueryFileListSizeAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryRegularFileListSizeAsc or the regular files only list:

\TeXOSQueryRegularFileListSizeAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQuerySubDirListSizeAsc or the sub-directories only list:

\TeXOSQuerySubDirListSizeAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterFileListSizeAsc or for the filtered list:

\TeXOSQueryFilterFileListSizeAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterRegularFileListSizeAsc or the filtered regular files only list:

\TeXOSQueryFilterRegularFileListSizeAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterSubDirListSizeAsc or the filtered sub-directories only list:

\TeXOSQueryFilterSubDirListSizeAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFileListSizeDes Order by file size starting with the largest (`size-descending`):

\TeXOSQueryFileListSizeDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryRegularFileListSizeDes or the regular files only list:

\TeXOSQueryRegularFileListSizeDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQuerySubDirListSizeDes or the sub-directories only list:

\TeXOSQuerySubDirListSizeDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterFileListSizeDes or for the filtered list:

\TeXOSQueryFilterFileListSizeDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterRegularFileListSizeDes or the filtered regular files only list:

\TeXOSQueryFilterRegularFileListSizeDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterSubDirListSizeDes or the filtered sub-directories only list:

\TeXOSQueryFilterSubDirListSizeDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFileListNameAsc Order by file name in alphabetical order (`name-ascending`):

\TeXOSQueryFileListNameAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryRegularFileListNameAsc or the regular files only list:

\TeXOSQueryRegularFileListNameAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQuerySubDirListNameAsc or the sub-directories only list:

\TeXOSQuerySubDirListNameAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterFileListNameAsc or for the filtered list:

\TeXOSQueryFilterFileListNameAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterRegularFileListNameAsc or the filtered regular files only list:

\TeXOSQueryFilterRegularFileListNameAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterSubDirListNameAsc or the filtered sub-directories only list:

\TeXOSQueryFilterSubDirListNameAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFileListNameDes Order by file name in reverse alphabetical order (`name-descending`):

\TeXOSQueryFileListNameDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryRegularFileListNameDes or the regular files only list:

\TeXOSQueryRegularFileListNameDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQuerySubDirListNameDes or the sub-directories only list:

\TeXOSQuerySubDirListNameDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterFileListNameDes or for the filtered list:

\TeXOSQueryFilterFileListNameDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterRegularFileListNameDes or the filtered regular files only list:

\TeXOSQueryFilterRegularFileListNameDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterSubDirListNameDes or the filtered sub-directories only list:

\TeXOSQueryFilterSubDirListNameDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFileListNameIgnoreCaseAsc Order by file name in case-insensitive alphabetical order (`iname-ascending`):

\TeXOSQueryFileListNameIgnoreCaseAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryRegularFileListNameIgnoreCaseAsc or the regular files only list:

\TeXOSQueryRegularFileListNameIgnoreCaseAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQuerySubDirListNameIgnoreCaseAsc or the sub-directories only list:

\TeXOSQuerySubDirListNameIgnoreCaseAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterFileListNameIgnoreCaseAsc or for the filtered list:

\TeXOSQueryFilterFileListNameIgnoreCaseAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterRegularFileListNameIgnoreCaseAsc or the filtered regular files only list:

\TeXOSQueryFilterRegularFileListNameIgnoreCaseAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterSubDirListNameIgnoreCaseAsc or the filtered sub-directories only list:

\TeXOSQueryFilterSubDirListNameIgnoreCaseAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

Order by file name in reverse case-insensitive alphabetical order (`iname-descending`):

\TeXOSQueryFileListNameIgnoreCaseDes

\TeXOSQueryFileListNameIgnoreCaseDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryRegularFileListNameIgnoreCaseDes or the regular files only list:

\TeXOSQueryRegularFileListNameIgnoreCaseDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQuerySubDirListNameIgnoreCaseDes or the sub-directories only list:

\TeXOSQuerySubDirListNameIgnoreCaseDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterFileListNameIgnoreCaseDes or for the filtered list:

\TeXOSQueryFilterFileListNameIgnoreCaseDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterRegularFileListNameIgnoreCaseDes or the filtered regular files only list:

\TeXOSQueryFilterRegularFileListNameIgnoreCaseDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterSubDirListNameIgnoreCaseDes or the filtered sub-directories only list:

\TeXOSQueryFilterSubDirListNameIgnoreCaseDes{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFileListExtAsc Order by file extension in alphabetical order (`ext-ascending`):

\TeXOSQueryFileListExtAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryRegularFileListExtAsc or the regular files only list:

\TeXOSQueryRegularFileListExtAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQuerySubDirListExtAsc or the sub-directories only list:

\TeXOSQuerySubDirListExtAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterFileListExtAsc or for the filtered list:

\TeXOSQueryFilterFileListExtAsc{⟨*cs*⟩}{⟨*sep*⟩}{⟨*regex*⟩}{⟨*dir*⟩}

\TeXOSQueryFilterRegularFileListExtAsc or the filtered regular files only list:

```
\TeXOSQueryFilterRegularFileListExtAsc{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryFilterSubDirListExtAsc or the filtered sub-directories only list:

```
\TeXOSQueryFilterSubDirListExtAsc{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryFileListExtDes Order by file extension in reverse alphabetical order (`ext-descending`):

```
\TeXOSQueryFileListExtDes{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

\TeXOSQueryRegularFileListExtDes or the regular files only list:

```
\TeXOSQueryRegularFileListExtDes{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

\TeXOSQuerySubDirListExtDes or the sub-directories only list:

```
\TeXOSQuerySubDirListExtDes{⟨cs⟩}{⟨sep⟩}{⟨dir⟩}
```

\TeXOSQueryFilterFileListExtDes or for the filtered list:

```
\TeXOSQueryFilterFileListExtDes{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryFilterRegularFileListExtDes or the filtered regular files only list:

```
\TeXOSQueryFilterRegularFileListExtDes{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryFilterSubDirListExtDes or the filtered sub-directories only list:

```
\TeXOSQueryFilterSubDirListExtDes{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryWalk A recursive filtered list of regular files starting from a given directory on the current working path (`-w` or `--alk`) can be obtained using:

```
\TeXOSQueryWalk{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

where ⟨*regex*⟩ is as for the filtered listings described above.

\TeXOSQueryWalkDateAsc To sort according to last modified date:

```
\TeXOSQueryWalkDateAsc{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryWalkDateDes or in reverse order:

```
\TeXOSQueryWalkDateDes{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryWalkSizeAsc To sort according to file size:

```
\TeXOSQueryWalkSizeAsc{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryWalkSizeDes or in reverse order:

```
\TeXOSQueryWalkSizeDes{⟨cs⟩}{⟨sep⟩}{⟨regex⟩}{⟨dir⟩}
```

\TeXOSQueryWalkNameAsc To sort according to path name (case-sensitive):

`\TeXOSQueryWalkNameAsc{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*regex*⟩`}{`⟨*dir*⟩`}`

`\TeXOSQueryWalkNameDes` or in reverse order:

`\TeXOSQueryWalkNameDes{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*regex*⟩`}{`⟨*dir*⟩`}`

`\TeXOSQueryWalkNameIgnoreCaseAsc` To sort according to path name (case-insensitive):

`\TeXOSQueryWalkNameIgnoreCaseAsc{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*regex*⟩`}{`⟨*dir*⟩`}`

`\TeXOSQueryWalkNameIgnoreCaseDes` in reverse order:

`\TeXOSQueryWalkNameIgnoreCaseDes{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*regex*⟩`}{`⟨*dir*⟩`}`

`\TeXOSQueryWalkExtAsc` To sort according to file extension:

`\TeXOSQueryWalkExtAsc{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*regex*⟩`}{`⟨*dir*⟩`}`

`\TeXOSQueryWalkExtDes` or in reverse order:

`\TeXOSQueryWalkExtDes{`⟨*cs*⟩`}{`⟨*sep*⟩`}{`⟨*regex*⟩`}{`⟨*dir*⟩`}`

3 The Code

3.1 Generic TEX Code

Change category code of `@` if necessary.

```
\ifnum\catcode`\@=11\relax
  \def\@texosquery@restore@at{}%
\else
  \expandafter\edef\csname @texosquery@restore@at\endcsname{%
    \noexpand\catcode`\noexpand\@=\number\catcode`\@\relax
  }%
 \catcode`\@=11\relax
\fi
```

Check if already loaded.

```
\ifx\TeXOSQuery\undefined \else
  \@texosquery@restore@at
  \expandafter\endinput
\fi
```

Version info.

```
\expandafter\def\csname ver@texosquery.tex\endcsname{2017/03/31 v1.3 (NLCT)}
```

`\@texosquery@warn` Generate warning message. Use tracklang's warning if available (so that the warnings can be disabled for both packages at the same time).

```
\ifx\@tracklang@pkgwarn\undefined
  \ifx\PackageWarning\undefined
    \def\@texosquery@warn#1{%
      {%
```

```
        \newlinechar=‘\^^J
        \def\MessageBreak{^^J}%
        \message{^^Jtexosquery Warning: #1 on line \the\inputlineno.^^J}%
      }%
    }
  \else
    \def\@texosquery@warn#1{%
      \PackageWarning{texosquery}{#1}%
    }
  \fi
\else
  \def\@texosquery@warn#1{%
    \@tracklang@pkgwarn{texosquery}{#1}%
  }
\fi
```

`\@texosquery@err`

```
\ifx\PackageError\undefined
  \def\@texosquery@err#1#2{%
    \errhelp{#2}%
    \errmessage{texosquery: #1}}
\else
  \def\@texosquery@err#1#2{\PackageError{texosquery}{#1}{#2}}
\fi
```

`\@texosquery@ifundef` This is defined in the same way as tracklang's `\@tracklang@ifundef`. (Can't assume tracklang has been loaded.)

```
\long\def\@texosquery@ifundef#1#2#3{%
  \ifcsname#1\endcsname
    \expandafter\ifx\csname #1\endcsname\relax
      #2%
    \else
      #3%
    \fi
  \else
    \expandafter\ifx\csname #1\endcsname\relax
      #2%
    \else
      #3%
    \fi
  \fi
}
\ifx\ifcsname\undefined
  \long\def\@texosquery@ifundef#1#2#3{%
    \expandafter\ifx\csname #1\endcsname\relax
      #2%
    \else
      #3%
    \fi
  }
```

```
\fi
```

`\TeXOSInvokerName` The name of the `texosquery` application.

```
\def\TeXOSInvokerName{texosquery}
```

`\TeXOSQueryInvoker` If we're using LaTeX, we'll need to use `\@@input` rather than `\input`.

```
\ifx\@@input\undefined
  \def\TeXOSQueryInvoker#1{\input|"\TeXOSInvokerName\space#1" }
\else
  \def\TeXOSQueryInvoker#1{\@@input|"\TeXOSInvokerName\space#1" }
\fi
```

`\ifTeXOSQueryDryRun` Provide a dry-run mode.

```
\newif\ifTeXOSQueryDryRun \TeXOSQueryDryRuntrue
```

`\TeXOSQueryAllowRestricted` The default behaviour only switches off the dry-run mode if the shell escape is unrestricted. The configuration file may override this with `\TeXOSQueryAllowRestricted`, which will allow the dry run mode to be switched off if restricted mode is detected.

```
\def\TeXOSQueryAllowRestricted{%
 \def\@texosquery@allowrestricted##1##2{##1}%
}
```

`\TeXOSQueryDenyRestricted` Switch it off.

```
\def\TeXOSQueryDenyRestricted{%
 \def\@texosquery@allowrestricted##1##2{##2}%
}
```

`\@texosquery@allowrestricted` Initialise to prevent shell escape in restricted mode.

```
\def\@texosquery@allowrestricted#1#2{#2}%
```

Load the configuration file if it exists.

```
\openin0=texosquery.cfg \ifeof0\relax \else
  \closein0\relax
  \begingroup
    \newlinechar=`\^^J
    \message{^^JTeXOSQuery: reading configuration file^^J}%
  \endgroup
  \input texosquery.cfg
\fi
```

Disable cfg-only commands:

```
\def\TeXOSQueryAllowRestricted{%
  \@texosquery@warn{\string\TeXOSQueryAllowRestricted\space
   ignored (only allowed in texosquery.cfg)}%
}
\def\TeXOSQueryDenyRestricted{%
  \@texosquery@warn{\string\TeXOSQueryDenyRestricted\space
   ignored (only allowed in texosquery.cfg)}%
}
```

If shell escape is unrestricted, automatically switch off dry-run mode, unless the cfg file has allowed it.

```
\ifx\shellescape\undefined
  \ifx\pdfshellescape\undefined
  \else
    \ifnum\pdfshellescape=1\relax
      \TeXOSQueryDryRunfalse
    \else
      \@texosquery@allowrestricted
      {%
        \ifnum\pdfshellescape=2\relax
          \TeXOSQueryDryRunfalse
        \fi
      }
      {}
    \fi
  \fi
\else
  \ifnum\shellescape=1\relax
    \TeXOSQueryDryRunfalse
  \else
    \@texosquery@allowrestricted
    {%
      \ifnum\shellescape=2\relax
        \TeXOSQueryDryRunfalse
      \fi
    }
    {}
  \fi
\fi
```

`\@texosquery@edef` Need to provide some protection (if available) against non-ASCII characters that have been made active by inputenc when reading in the results of the shell escape. This command may be defined before loading texosquery, otherwise it's set to `\protected@edef`, if defined, or `\edef`.

```
\ifx\@texosquery@edef\undefined
  \ifx\protected@edef\undefined
    \let\@texosquery@edef\edef
  \else
    \let\@texosquery@edef\protected@edef
  \fi
\fi
```

Provide some utility commands. (Can't use `\@gobble` etc, as we may not be using LaTeX.)

`\@texosquery@gobble`

```
\def\@texosquery@gobble#1{}
```

\@texosquery@firstofone

```
130 \def\@texosquery@firstofone#1{#1}
```

The results obtained from texosquery may be file names for use in commands like \input or \includegraphics or they may be text that needs typesetting (such as month names) or they may be date-time patterns or numeric patterns or they may be PDF date-time strings, which may need to have the category code of the initial "D" set to 12 for parsing commands that include this character in the argument syntax.

This means that we need to take special characters into account, but the way they are dealt with vary according to context. For example, # needs to have the category code set to 12 if it's part of a file name. If an image file is called, say, test_imagefile#.png then the following doesn't work:

```
\includegraphics{test\_imagefile\#}
```

It needs to be

```
\includegraphics{test\string_imagefile\string#}
```

or

```
\includegraphics{\detokenize{test_imagefile#}}
```

The first two versions of texosquery try to deal with this by simply changing the category code of _ to 12 and getting texosquery to replace all instances of # with \#. This hash substitution doesn't work with the above image example so version 1.2 introduced a new command that texosquery could use instead of \# that expands to \string#. This now solves the problem for file names that are obtained through texosquery, but texosquery doesn't solely return file names. It also returns text that needs typesetting and it also returns numeric patterns, which in their raw form include # as a digit identifier.

This means that we can't simply detokenize the result from texosquery. Instead texosquery replaces problematic characters with control sequences *according to context*. For example, \texosqueryhash is used in a file name context, \texosquerytexthash is used in a textual context and \texosquerypatdigitnozero in a numeric pattern context.

These long control sequence names clutter the results when testing the application directly in a terminal, so the Java code uses short forms that are locally defined by \TeXOSQuery to expand to the longer forms.

\texosquerynonasciiwrap

Allow a way to deal with non-ASCII characters returned by texosquery. \TeXOSQuery locally defines \twrp to this command. By default this just does its argument but may be redefined. For example, if the document uses a different file encoding to Java, then this command might need to be redefined to perform the appropriate conversion.

```
131 \def\texosquerynonasciiwrap#1{#1}
```

\texosquerynonasciidetokwrap

We also need to allow for non-ASCII characters in file names. In this case the argument needs detokenizing. With eTEX, we can simply use \detokenize but we need to allow for plain non-extended TEX, so check for the existence of \detokenize first.

```
132 \ifx\detokenize\undefined
```

This won't work for characters consisting of multiple octets, but if users want UTF-8 support then they really need eTEX at the very least (but ideally XETEX or LuaTEX).

```
  \def\texosquerynonasciidetokwrap#1{\string#1}
\else
  \def\texosquerynonasciidetokwrap#1{\detokenize{#1}}
\fi
```

Now define commands used in `\TeXOSQuery` for various escaped characters. The literal versions are for file names. The textual versions are for use within the document text. For completeness, all the ASCII punctuation characters have both a literal and textual version. This helps to protect against babel shorthands etc.

`\texosquerybackslash` Literal backslash.

```
\edef\texosquerybackslash{\expandafter\@texosquery@gobble\string\\}
```

`\texosquerytextbackslash` Textual backslash.

```
\ifx\textbackslash\undefined
  \def\texosquerytextbackslash{\texosquerybackslash}
\else
  \def\texosquerytextbackslash{\noexpand\textbackslash}
\fi
```

`\texosqueryleftbrace` Literal left brace.

```
\edef\texosqueryleftbrace{\expandafter\@texosquery@gobble\string\{}
```

`\texosquerytextleftbrace` Textual left brace.

```
\def\texosquerytextleftbrace{\{}
```

`\texosqueryrightbrace` Literal right brace.

```
\edef\texosqueryrightbrace{\expandafter\@texosquery@gobble\string\}}
```

`\texosquerytextrightbrace` Textual right brace.

```
\def\texosquerytextrightbrace{\}}
```

`\texosqueryhash` Literal hash.

```
\edef\texosqueryhash{\expandafter\@texosquery@gobble\string\#}
```

`\texosquerytexthash` Textual hash.

```
\def\texosquerytexthash{\#}
```

`\texosqueryunderscore` Literal underscore.

```
\edef\texosqueryunderscore{\expandafter\@texosquery@gobble\string\_}
```

`\texosquerytextunderscore` Textual underscore.

```
\def\texosquerytextunderscore{\_}
```

`\texosquerybacktick` Literal grave.

```
\edef\texosquerybacktick{\string`}
```

`\texosquerytextbacktick` Textual open quote.

```
\def\texosquerytextbacktick{‘}
```

`\texosqueryclosequote` Literal apostrophe.

```
\edef\texosqueryclosequote{\string’}
```

`\texosquerytextclosequote` Textual apostrophe / single closing quote.

```
\def\texosquerytextclosequote{’}
```

`\texosquerydoublequote` Literal double-quote.

```
\edef\texosquerydoublequote{\string"}
```

`\texosquerytextdoublequote` Textual double-quote.

```
\def\texosquerytextdoublequote{"}
```

`\texosquerycolon` Literal colon.

```
\edef\texosquerycolon{\string:}
```

`\texosquerytextcolon` Textual colon.

```
\def\texosquerytextcolon{:}
```

`\texosquerysemicolon` Literal semi-colon.

```
\edef\texosquerysemicolon{\string;}
```

`\texosquerytextsemicolon` Textual semi-colon.

```
\def\texosquerytextsemicolon{;}
```

`\texosqueryequals` Literal equals.

```
\edef\texosqueryequals{\string=}
```

`\texosquerytextequals` Textual equals.

```
\def\texosquerytextequals{=}
```

`\texosqueryslash` Literal slash.

```
\edef\texosqueryslash{\string/}
```

`\texosquerytextslash` Textual slash.

```
\def\texosquerytextslash{/}
```

`\texosqueryhyphen` Literal hyphen.

```
\edef\texosqueryhyphen{\string-}
```

`\texosquerytexthyphen` Textual hyphen.

```
\def\texosquerytexthyphen{-}
```

`\texosqueryplus` Literal plus.

```
\edef\texosqueryplus{\string+}
```

`\texosquerytextplus` Textual plus.

```
\def\texosquerytextplus{+}
```

`\texosqueryperiod` Literal period.

```
\edef\texosqueryperiod{\string.}
```

`\texosquerytextperiod` Textual period.

```
\def\texosquerytextperiod{.}
```

`\texosquerycomma` Literal comma.

```
\edef\texosquerycomma{\string,}
```

`\texosquerytextcomma` Textual comma.

```
\def\texosquerytextcomma{,}
```

`\texosqueryopenparen` Literal open bracket.

```
\edef\texosqueryopenparen{\string(}
```

`\texosquerytextopenparen` Textual open bracket.

```
\def\texosquerytextopenparen{(}
```

`\texosquerycloseparen` Literal close bracket.

```
\edef\texosquerycloseparen{\string)}
```

`\texosquerytextcloseparen` Textual close bracket.

```
\def\texosquerytextcloseparen{)}
```

`\texosqueryopensq` Literal open square bracket.

```
\edef\texosqueryopensq{\string[}
```

`\texosquerytextopensq` Textual open square bracket.

```
\def\texosquerytextopensq{[}
```

`\texosqueryclosesq` Literal close square bracket.

```
\edef\texosqueryclosesq{\string]}
```

`\texosquerytextclosesq` Textual close square bracket.

```
\def\texosquerytextclosesq{]}
```

`\texosqueryasterisk` Literal asterisk.

```
\edef\texosqueryasterisk{\string*}
```

`\texosquerytextasterisk` Textual asterisk.

```
\def\texosquerytextasterisk{*}
```

`\texosqueryatchar` Literal at character.

```
\edef\texosqueryatchar{\string @}
```

\texosquerytextatchar Textual at character.

```
\def\texosquerytextatchar{@}
```

\texosquerybar Literal bar.

```
\edef\texosquerybar{\string|}
```

\texosquerytextbar Textual bar.

```
\ifx\undefined\textbar
  \def\texosquerytextbar{|}
\else
  \def\texosquerytextbar{\ifmmode|\else\textbar\fi}
\fi
```

\texosquerylessthan Literal less than.

```
\edef\texosquerylessthan{\string<}
```

\texosquerytextlessthan Textual less than.

```
\ifx\undefined\textless
  \def\texosquerytextlessthan{<}
\else
  \def\texosquerytextlessthan{\ifmmode<\else\textless\fi}
\fi
```

\texosquerygreaterthan Literal greater than.

```
\edef\texosquerygreaterthan{\string>}
```

\texosquerytextgreaterthan Textual greater than.

```
\ifx\undefined\textgreater
  \def\texosquerytextgreaterthan{>}
\else
  \def\texosquerytextgreaterthan{\ifmmode<\else\textgreater\fi}
\fi
```

\texosquerytilde Literal tilde.

```
\edef\texosquerytilde{\string~}
```

\texosquerytexttilde Textual tilde.

```
\ifx\textasciitilde\undefined
  \def\texosquerytexttilde{\string~}
\else
  \def\texosquerytexttilde{\textasciitilde}
\fi
```

\texosquerycircum Literal circumflex.

```
\edef\texosquerycircum{\string^}
```

`\texosquerytextcircum` Textual circumflex.

```
\ifx\textasciicircum\undefined
  \def\texosquerytextcircum{\string^}
\else
  \def\texosquerytextcircum{\textasciicircum}
\fi
```

`\texosqueryampersand` Literal ampersand.

```
\edef\texosqueryampersand{\string&}
```

`\texosquerytextampersand` Textual ampersand.

```
\def\texosquerytextampersand{\&}
```

`\texosquerydollar` Literal dollar. (This could just be defined as `\string$`, but that plays havoc with the syntax highlighting!)

```
\edef\texosquerydollar{\expandafter\@texosquery@gobble\string\$}
```

`\texosquerytextdollar` Textual dollar.

```
\def\texosquerytextdollar{\$}
```

`\texosquerypercent` Literal percent.

```
\edef\texosquerypercent{\expandafter\@texosquery@gobble\string\%}
```

`\texosquerytextpercent` Textual percent.

```
\def\texosquerytextpercent{\%}
```

`\texosqueryexclam` Literal exclamation.

```
\edef\texosqueryexclam{\string!}
```

`\texosquerytextexclam` Textual exclamation.

```
\def\texosquerytextexclam{!}
```

`\texosqueryquestion` Literal question mark.

```
\edef\texosqueryquestion{\string?}
```

`\texosquerytextquestion` Textual question mark.

```
\def\texosquerytextquestion{?}
```

`\texosqueryliteralspace` Literal space.

```
\edef\texosqueryliteralspace{\expandafter\string\space}
```

`\texosquerytextspace` Textual space. (Don't allow it to expand while it's being fetched from `\texosquery` just in case it disappears.)

```
\def\texosquerytextspace{\noexpand\space}
```

`\@texosquery@D` The `D` identifier in PDF date-time formats need to have category code 12. This is only used by methods that return results in the form:

`D:`⟨*YYYY*⟩⟨*MM*⟩{⟨*DD*⟩}⟨*HH*⟩⟨*mm*⟩⟨*ss*⟩⟨*TZh*⟩'⟨*TZm*⟩'

```
\edef\@texosquery@D{\string D}
```

`\@texosquery@enableshortcs` Enable shortcut commands.

```
\def\@texosquery@enableshortcs{%
```

Thesre are for the date-time and numeric patterns.

```
    \def\patdtf{\noexpand\texosquerydtf}%
    \def\patpmnumfmt{\noexpand\texosquerypatplusminus}%
    \def\patnumfmt{\noexpand\texosquerypatnum}%
    \def\patsinumfmt{\noexpand\texosquerypatsinum}%
    \def\patdecfmt{\noexpand\texosquerypatdec}%
    \def\patpcur{\noexpand\texosquerypatprefixcurrency}%
    \def\patpicur{\noexpand\texosquerypatprefixicurrency}%
    \def\patscur{\noexpand\texosquerypatsuffixcurrency}%
    \def\patsicur{\noexpand\texosquerypatsuffixicurrency}%
    \def\patstr{\noexpand\texosquerypatstr}%
    \def\patapo{\noexpand\texosquerypatquote}%
    \def\patdgt{\noexpand\texosquerypatdigit}%
    \def\patdgtnz{\noexpand\texosquerypatdigitnozero}%
    \def\patmsg{\noexpand\texosquerypatminus}%
    \def\patngp{\noexpand\texosquerypatgroupsep}%
    \def\patppct{\noexpand\texosquerypatprefixpercent}%
    \def\patspct{\noexpand\texosquerypatsuffixpercent}%
    \def\patppml{\noexpand\texosquerypatprefixpermill}%
    \def\patspml{\noexpand\texosquerypatsuffixpermill}%
```

Hook to adjust the processing of non-ASCII characters.

```
    \def\twrp{\texosquerynonasciiwrap}%
    \def\fwrp{\texosquerynonasciidetokwrap}%
```

Locally redefine some more commands that may occur in `texosquery`'s return value (via the `escapeSpChars` method in `TeXOSQuery.java`). The `t` prefix indicates textual commands and the `f` prefix indicates literal characters, for example, in file names.

```
    \let\fbks\texosquerybackslash
    \let\tbks\texosquerytextbackslash
    \let\flbr\texosqueryleftbrace
    \let\tlbr\texosquerytextleftbrace
    \let\frbr\texosqueryrightbrace
    \let\trbr\texosquerytextrightbrace
    \let\fhsh\texosqueryhash
    \let\thsh\texosquerytexthash
    \let\fusc\texosqueryunderscore
    \let\tusc\texosquerytextunderscore
    \let\fgrv\texosquerybacktick
    \let\tgrv\texosquerytextbacktick
    \let\fapo\texosqueryclosequote
    \let\tapo\texosquerytextclosequote
    \let\fdqt\texosquerydoublequote
    \let\tdqt\texosquerytextdoublequote
    \let\fspc\texosqueryliteralspace
    \let\tspc\texosquerytextspace
    \let\fcln\texosquerycolon
    \let\tcln\texosquerytextcolon
```

```
    \let\fscl\texosquerysemicolon
    \let\tscl\texosquerytextsemicolon
    \let\feql\texosqueryequals
    \let\teql\texosquerytextequals
    \let\fhyn\texosqueryhyphen
    \let\thyn\texosquerytexthyphen
    \let\fpls\texosqueryplus
    \let\tpls\texosquerytextplus
    \let\ftld\texosquerytilde
    \let\ttld\texosquerytexttilde
    \let\fcir\texosquerycircum
    \let\tcir\texosquerytextcircum
    \let\famp\texosqueryampersand
    \let\tamp\texosquerytextampersand
    \let\fslh\texosqueryslash
    \let\tslh\texosquerytextslash
    \let\fpct\texosquerypercent
    \let\tpct\texosquerytextpercent
    \let\fexc\texosqueryexclam
    \let\texc\texosquerytextexclam
    \let\fque\texosqueryquestion
    \let\tque\texosquerytextquestion
    \let\fles\texosquerylessthan
    \let\tles\texosquerytextlessthan
    \let\fgre\texosquerygreaterthan
    \let\tgre\texosquerytextgreaterthan
    \let\fdol\texosquerydollar
    \let\tdol\texosquerytextdollar
    \let\fdot\texosqueryperiod
    \let\tdot\texosquerytextperiod
    \let\fcom\texosquerycomma
    \let\tcom\texosquerytextcomma
    \let\fopb\texosqueryopenparen
    \let\topb\texosquerytextopenparen
    \let\fclb\texosquerycloseparen
    \let\tclb\texosquerytextcloseparen
    \let\fosb\texosqueryopensq
    \let\tosb\texosquerytextopensq
    \let\fcsb\texosqueryclosesq
    \let\tcsb\texosquerytextclosesq
    \let\fast\texosqueryasterisk
    \let\tast\texosquerytextasterisk
    \let\fatc\texosqueryatchar
    \let\tatc\texosquerytextatchar
    \let\pdfd\@texosquery@D
}
```

`\TeXOSQuery` Use `texosquery` with the option given in the second argument and store the result in control sequence given in the first argument.

```
\def\TeXOSQuery#1#2{%
```

```
\ifTeXOSQueryDryRun
  \begingroup
    \newlinechar=‘\^^J
    \message{^^JTeXOSQuery: \TeXOSInvokerName\space#2^^J}%
  \endgroup
  \def#1{}%
\else
  \begingroup
  \endlinechar=-1\relax
```

Locally redefine short commands used by `texosquery`

```
    \@texosquery@enableshortcs
```

Change the category code of some potentially awkward characters. (This should no longer be an issue with the new commands that are now used in the returned text, but `texosquery` might be run with the backward compatibility mode on, so this is still needed just in case.)

```
    \catcode‘\-=12\relax
    \catcode‘\_=12\relax
    \catcode‘\^=12\relax
    \catcode‘\~=12\relax
    \catcode‘\$=12\relax
    \catcode‘\&=12\relax
    \catcode‘\.=12\relax
    \catcode‘\/=12\relax
    \catcode‘\:=12\relax
    \catcode‘\"=12\relax
    \catcode‘\’=12\relax
    \catcode‘\;=12\relax
    \catcode‘\%=12\relax
    \everyeof{\noexpand}\relax
    \@texosquery@edef\x{\endgroup\def\noexpand#1{\TeXOSQueryInvoker{#2}}}\x
  \fi
}
```

3.1.1 Currency

The ⟨*TEX currency*⟩ element of `--numeric` and `--locale-data` identifies the currency symbol using

`\texosquerycurrency{`⟨*label*⟩`}`

which simply expands to the appropriate command.

`\texosquerycurrency`

```
\def\texosquerycurrency#1{%
 \expandafter\noexpand\csname texosquerycurrency#1\endcsname
}
```

Provide the currency commands that may be returned `texosquery` (on expansion of `\texosquerycurrency`). Most of these will need redefining as there's no appropriate generic code to use as a default. The fontawesome package has the most support for currency symbols, so these are checked first.

\texosquerycurrencydollar

```
\ifx\faDollar\undefined
  \def\texosquerycurrencydollar{\$}
\else
  \def\texosquerycurrencydollar{\faDollar}
\fi
```

\texosquerycurrencycent

```
\ifx\textcent\undefined
  \def\texosquerycurrencycent{cent}
\else
  \def\texosquerycurrencycent{\textcent}
\fi
```

\texosquerycurrencypound

```
\ifx\faGbp\undefined
  \ifx\pounds\undefined
    \def\texosquerycurrencypound{pound}
  \else
    \def\texosquerycurrencypound{\pounds}
  \fi
\else
  \def\texosquerycurrencypound{\faGbp}
\fi
```

\texosquerycurrencysign

```
\ifx\textcurrency\undefined
  \def\texosquerycurrencysign{currency-sign}
\else
  \def\texosquerycurrencysign{\textcurrency}
\fi
```

\texosquerycurrencyyen

```
\ifx\faYen\undefined
  \ifx\textyen\undefined
    \def\texosquerycurrencyyen{yen}
  \else
    \def\texosquerycurrencyyen{\textyen}
  \fi
\else
  \def\texosquerycurrencyyen{\faYen}
\fi
```

\texosquerycurrencyecu

```
\def\texosquerycurrencyecu{ecu}
```

\texosquerycurrencycolon

```
\def\texosquerycurrencycolon{colon}
```

```
\texosquerycurrencycruzeiro
                              \def\texosquerycurrencycruzeiro{cruzeiro}
   \texosquerycurrencyfranc
                              \def\texosquerycurrencyfranc{franc}
    \texosquerycurrencylira
                              \ifx\textlira\undefined
                                \def\texosquerycurrencylira{lira}
                              \else
                                \def\texosquerycurrencylira{\textlira}
                              \fi
    \texosquerycurrencymill
                              \def\texosquerycurrencymill{mill}
   \texosquerycurrencynaira
                              \ifx\textnaira\undefined
                                \def\texosquerycurrencynaira{naira}
                              \else
                                \def\texosquerycurrencynaira{\textnaira}
                              \fi
  \texosquerycurrencypeseta
                              \def\texosquerycurrencypeseta{peseta}
   \texosquerycurrencyrupee
                              \ifx\faRupee\undefined
                                \def\texosquerycurrencyrupee{rupee}
                              \else
                                \def\texosquerycurrencyrupee{\faRupee}
                              \fi
     \texosquerycurrencywon
                              \ifx\faWon\undefined
                                \ifx\textwon\undefined
                                  \def\texosquerycurrencywon{won}
                                \else
                                  \def\texosquerycurrencywon{\textwon}
                                \fi
                              \else
                                \def\texosquerycurrencywon{\faWon}
                              \fi
\texosquerycurrencynewsheqel
                              \ifx\faSheqel\undefined
                                \def\texosquerycurrencynewsheqel{newsheqel}
                              \else
                                \def\texosquerycurrencynewsheqel{\faSheqel}
                              \fi
```

```
\texosquerycurrencydong
\ifx\textdong\undefined
  \def\texosquerycurrencydong{dong}
\else
  \def\texosquerycurrencydong{\textdong}
\fi

\texosquerycurrencyeuro
\ifx\faEuro\undefined
  \ifx\texteuro\undefined
    \ifx\euro\undefined
      \def\texosquerycurrencyeuro{euro}
    \else
      \def\texosquerycurrencyeuro{\euro}
    \fi
  \else
    \def\texosquerycurrencyeuro{\texteuro}
  \fi
\else
  \def\texosquerycurrencyeuro{\faEuro}
\fi

\texosquerycurrencykip
\def\texosquerycurrencykip{kip}

\texosquerycurrencytugrik
\def\texosquerycurrencytugrik{tugrik}

\texosquerycurrencydrachma
\def\texosquerycurrencydrachma{drachma}

\texosquerycurrencygermanpenny
\def\texosquerycurrencygermanpenny{german-penny}

\texosquerycurrencypeso
\ifx\textpeso\undefined
  \def\texosquerycurrencypeso{peso}
\else
  \def\texosquerycurrencypeso{\textpeso}
\fi

\texosquerycurrencyguarani
\ifx\textguarani\undefined
  \def\texosquerycurrencyguarani{guarani}
\else
  \def\texosquerycurrencyguarani{\textguarani}
\fi
```

```
    \texosquerycurrencyaustral
                                447 \def\texosquerycurrencyaustral{austral}
    \texosquerycurrencyhryvnia
                                448 \def\texosquerycurrencyhryvnia{hryvnia}
       \texosquerycurrencycedi
                                449 \ifx\textcolonmonetary\undefined
                                450   \def\texosquerycurrencycedi{cedi}
                                451 \else
                                452   \def\texosquerycurrencycedi{\textcolonmonetary}
                                453 \fi
texosquerycurrencylivretournois
                                454 \def\texosquerycurrencylivretournois{livre-tournois}
   \texosquerycurrencyspesmilo
                                455 \def\texosquerycurrencyspesmilo{spesmilo}
      \texosquerycurrencytenge
                                456 \def\texosquerycurrencytenge{tenge}
      \texosquerycurrencyrupee
                                457 \def\texosquerycurrencyrupee{rupee}
\texosquerycurrencyturkishlira
                                458 \ifx\faTurkishLira\undefined
                                459   \def\texosquerycurrencyturkishlira{turkish-lira}
                                460 \else
                                461   \def\texosquerycurrencyturkishlira{\faTurkishLira}
                                462 \fi
 \texosquerycurrencynordicmark
                                463 \def\texosquerycurrencynordicmark{nordic-mark}
      \texosquerycurrencymanat
                                464 \def\texosquerycurrencymanat{manat}
      \texosquerycurrencyruble
                                465 \ifx\faRuble\undefined
                                466   \def\texosquerycurrencyruble{ruble}
                                467 \else
                                468   \def\texosquerycurrencyruble{\faRuble}
                                469 \fi
```

3.1.2 Shortcut Commands

Now for some convenient shortcuts so the user doesn't have to remember the command line options. `\string` is used in case the hyphen character has been made active.

`\TeXOSQueryLocale` Query the locale and store the result in the control sequence provided in the argument.

```
\def\TeXOSQueryLocale#1{\TeXOSQuery{#1}{\string-l}}
```

`\TeXOSQueryLangTag` Query the language tag and store the result in the control sequence provided in the argument.

```
\def\TeXOSQueryLangTag#1{\TeXOSQuery{#1}{\string-b}}
```

`\TeXOSQueryNumeric` Query the numeric settings for the locale given in the second argument and store the result in the control sequence provided in the argument. Leave the second argument empty for the default locale.

```
\def\TeXOSQueryNumeric#1#2{\TeXOSQuery{#1}{\string-N #2}}
```

`\TeXOSQueryLocaleData` Query the data for the locale given in the second argument and store the result in the control sequence provided in the argument. Leave the second argument empty for the default locale.

```
\def\TeXOSQueryLocaleData#1#2{\TeXOSQuery{#1}{\string-D #2}}
```

`\TeXOSQueryCwd` Query the current working directory.

```
\def\TeXOSQueryCwd#1{\TeXOSQuery{#1}{\string-c}}
```

`\TeXOSQueryHome` Query the user's home directory.

```
\def\TeXOSQueryHome#1{\TeXOSQuery{#1}{\string-m}}
```

`\TeXOSQueryTmpDir` Query the temporary directory.

```
\def\TeXOSQueryTmpDir#1{\TeXOSQuery{#1}{\string-t}}
```

`\TeXOSQueryVersion` Query the operating system version.

```
\def\TeXOSQueryVersion#1{\TeXOSQuery{#1}{\string-r}}
```

`\TeXOSQueryArch` Query the operating system architecture.

```
\def\TeXOSQueryArch#1{\TeXOSQuery{#1}{\string-a}}
```

`\TeXOSQueryName` Query the operating system name.

```
\def\TeXOSQueryName#1{\TeXOSQuery{#1}{\string-o}}
```

`\TeXOSQueryDateTime` Query the current date and time.

```
\def\TeXOSQueryDateTime#1{%
  \TeXOSQuery{#1}{\string-M}%
}
```

`\TeXOSQueryTimeZones` Query the current time zone mappings. Leave the second argument empty for the default locale.

```
\def\TeXOSQueryTimeZones#1#2{%
  \TeXOSQuery{#1}{\string-Z #2}%
}
```

\TeXOSQueryNow Query the current time stamp.

```
\def\TeXOSQueryNow#1{%
```

The D needs category code 12 just in case texosquery is running in a backward compatibility mode that doesn't use \pdfd.

```
  \edef\@texosquery@restore@D{%
    \noexpand\catcode`\noexpand\D=\the\catcode`\D\relax}%
  \catcode`\D=12\relax
  \TeXOSQuery{#1}{\string-n}%
  \@texosquery@restore@D
}
```

If the file name is supplied using \jobname it may have double-quotes which will interfere with things.

\texosquerystripquotes

```
\def\texosquerystripquotes#1{%
  \@texosquery@stripquotes#1\@mid@texosquery@stripquotes
    "\relax"\relax\@end@texosquery@stripquotes
}
\def\@texosquery@stripquotes#1"#2"{%
  \@@texosquery@stripquotes#1#2%
}
\def\@@texosquery@stripquotes#1\@mid@texosquery@stripquotes#2\@end@texosquery@stripquotes{%
  #1%
}
```

\TeXOSQueryFileDate Query the time stamp of the file given in the second argument.

```
\def\TeXOSQueryFileDate#1#2{%
```

The D needs category code 12 just in case texosquery is running in a backward compatibility mode that doesn't use \pdfd.

```
  \edef\@texosquery@restore@D{%
    \noexpand\catcode`\noexpand\D=\the\catcode`\D\relax}%
  \catcode`\D=12\relax
  \TeXOSQuery{#1}{\string-d \string'\texosquerystripquotes{#2}\string'}%
  \@texosquery@restore@D
}
```

\TeXOSQueryFileSize Query the size of the file given in the second argument.

```
\def\TeXOSQueryFileSize#1#2{\TeXOSQuery{#1}{\string-s
 \string'\texosquerystripquotes{#2}\string'}}
```

\@texosquery@filelist

```
\def\@texosquery@filelist#1#2#3#4#5{\TeXOSQuery{#1}{%
 \string#2 \string'#3\string'
 \string'\texosquerystripquotes{#4}\string' #5}}
```

\TeXOSQueryFileList List all files in the directory given in the third argument, separated by the second argument.

```
\def\TeXOSQueryFileList#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{}%
}
```

\TeXOSQueryFileListDateAsc As above, but sort by date.

```
\def\TeXOSQueryFileListDateAsc#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{date}%
}
```

\TeXOSQueryFileListDateDes As above, but sort by date in descending order.

```
\def\TeXOSQueryFileListDateDes#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{date\string-des}%
}
```

\TeXOSQueryFileListSizeAsc As above, but sort by size.

```
\def\TeXOSQueryFileListSizeAsc#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{size}%
}
```

\TeXOSQueryFileListSizeDes As above, but sort by size in descending order.

```
\def\TeXOSQueryFileListSizeDes#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{size\string-des}%
}
```

\TeXOSQueryFileListNameAsc As above, but sort by name.

```
\def\TeXOSQueryFileListNameAsc#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{name}%
}
```

\TeXOSQueryFileListNameDes As above, but sort by name in descending order.

```
\def\TeXOSQueryFileListNameDes#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{name\string-des}%
}
```

SQueryFileListNameIgnoreCaseAsc As above, but sort by case-insensitive name.

```
\def\TeXOSQueryFileListNameIgnoreCaseAsc#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{iname}%
}
```

SQueryFileListNameIgnoreCaseDes As above, but sort by case-insensitive name in descending order.

```
\def\TeXOSQueryFileListNameIgnoreCaseDes#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{iname\string-des}%
}
```

\TeXOSQueryFileListExtAsc As above, but sort by extension.

```
\def\TeXOSQueryFileListExtAsc#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{ext}%
}
```

\TeXOSQueryFileListExtDes As above, but sort by extension in descending order.

```
\def\TeXOSQueryFileListExtDes#1#2#3{%
 \@texosquery@filelist{#1}{-i}{#2}{#3}{ext\string-des}%
}
```

\TeXOSQueryRegularFileList List all regular files.

```
\def\TeXOSQueryRegularFileList#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{}%
}
```

\TeXOSQuerySubDirList List all sub-directories.

```
\def\TeXOSQuerySubDirList#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{}%
}
```

eXOSQueryRegularFileListDateAsc List all regular files sorted by date.

```
\def\TeXOSQueryRegularFileListDateAsc#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{date}%
}
```

\TeXOSQuerySubDirListDateAsc List all sub-directories sorted by date.

```
\def\TeXOSQuerySubDirListDateAsc#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{date}%
}
```

eXOSQueryRegularFileListDateDes List all regular files sorted by date in descending order.

```
\def\TeXOSQueryRegularFileListDateDes#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{date\string-des}%
}
```

\TeXOSQuerySubDirListDateDes List all sub-directories sorted by date in descending order.

```
\def\TeXOSQuerySubDirListDateDes#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{date\string-des}%
}
```

eXOSQueryRegularFileListSizeAsc List all regular files sorted by size.

```
\def\TeXOSQueryRegularFileListSizeAsc#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{size}%
}
```

\TeXOSQuerySubDirListSizeAsc List all sub-directories sorted by size.

```
\def\TeXOSQuerySubDirListSizeAsc#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{size}%
}
```

eXOSQueryRegularFileListSizeDes List all regular files sorted by size in descending order.

```
\def\TeXOSQueryRegularFileListSizeDes#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{size\string-des}%
}
```

\TeXOSQuerySubDirListSizeDes List all sub-directories sorted by size in descending order.

```
\def\TeXOSQuerySubDirListSizeDes#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{size\string-des}%
}
```

eXOSQueryRegularFileListNameAsc List all regular files sorted by file name.

```
\def\TeXOSQueryRegularFileListNameAsc#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{name}%
}
```

\TeXOSQuerySubDirListNameAsc List all sub-directories sorted by file name.

```
\def\TeXOSQuerySubDirListNameAsc#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{name}%
}
```

eXOSQueryRegularFileListNameDes List all regular files sorted by file name in descending order.

```
\def\TeXOSQueryRegularFileListNameDes#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{name\string-des}%
}
```

\TeXOSQuerySubDirListNameDes List all sub-directories sorted by name in descending order.

```
\def\TeXOSQuerySubDirListNameDes#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{name\string-des}%
}
```

egularFileListNameIgnoreCaseAsc List all regular files sorted by file case-insensitive name.

```
\def\TeXOSQueryRegularFileListNameIgnoreCaseAsc#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{iname}%
}
```

uerySubDirListNameIgnoreCaseAsc List all sub-directories sorted by file case-insensitive name.

```
\def\TeXOSQuerySubDirListNameIgnoreCaseAsc#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{iname}%
}
```

egularFileListNameIgnoreCaseDes List all regular files sorted by file case-insensitive name in descending order.

```
\def\TeXOSQueryRegularFileListNameIgnoreCaseDes#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{iname\string-des}%
}
```

uerySubDirListNameIgnoreCaseDes List all sub-directories sorted by case-insensitive name in descending order.

```
\def\TeXOSQuerySubDirListNameIgnoreCaseDes#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{iname\string-des}%
}
```

TeXOSQueryRegularFileListExtAsc List all regular files sorted by file extension.

```
\def\TeXOSQueryRegularFileListExtAsc#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{ext}%
}
```

\TeXOSQuerySubDirListExtAsc List all sub-directories sorted by file extension.

```
\def\TeXOSQuerySubDirListExtAsc#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{ext}%
}
```

TeXOSQueryRegularFileListExtDes List all regular files sorted by file extension in descending order.

```
\def\TeXOSQueryRegularFileListExtDes#1#2#3{%
 \@texosquery@filelist{#1}{-ir}{#2}{#3}{ext\string-des}%
}
```

\TeXOSQuerySubDirListExtDes List all sub-directories sorted by extension in descending order.

```
\def\TeXOSQuerySubDirListExtDes#1#2#3{%
 \@texosquery@filelist{#1}{-id}{#2}{#3}{ext\string-des}%
}
```

\@texosquery@filterfilelist

```
\def\@texosquery@filterfilelist#1#2#3#4#5#6{%
 \TeXOSQuery{#1}%
 {%
   \string#2 \string'#3\string' \string'#4\string'
   \string'\texosquerystripquotes{#5}\string' #6%
 }%
}
```

\TeXOSQueryFilterFileList Filtered list files in the directory given in the fourth argument, separated by the second argument. The third argument is the regular expression used to filter the list. *Take care of backslashes in the regular expression!*

```
\def\TeXOSQueryFilterFileList#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{}%
}
```

TeXOSQueryFilterFileListDateAsc As above, but sort by date.

```
\def\TeXOSQueryFilterFileListDateAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{date}%
}
```

TeXOSQueryFilterFileListDateDes As above, but sort by date in descending order.

```
\def\TeXOSQueryFilterFileListDateDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{date\string-des}%
}
```

TeXOSQueryFilterFileListSizeAsc As above, but sort by size.

```
\def\TeXOSQueryFilterFileListSizeAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{size}%
}
```

TeXOSQueryFilterFileListSizeDes As above, but sort by size in descending order.

```
\def\TeXOSQueryFilterFileListSizeDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{size\string-des}%
}
```

TeXOSQueryFilterFileListNameAsc As above, but sort by file name.

```
\def\TeXOSQueryFilterFileListNameAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{name}%
}
```

TeXOSQueryFilterFileListNameDes As above, but sort by name in descending order.

```
\def\TeXOSQueryFilterFileListNameDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{name\string-des}%
}
```

FilterFileListNameIgnoreCaseAsc As above, but sort by file name (case-insensitive).

```
\def\TeXOSQueryFilterFileListNameIgnoreCaseAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{iname}%
}
```

FilterFileListNameIgnoreCaseDes As above, but sort by name in descending order (case-insensitive).

```
\def\TeXOSQueryFilterFileListNameIgnoreCaseDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{iname\string-des}%
}
```

\TeXOSQueryFilterFileListExtAsc As above, but sort by file extension.

```
\def\TeXOSQueryFilterFileListExtAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{ext}%
}
```

\TeXOSQueryFilterFileListExtDes As above, but sort by extension in descending order.

```
\def\TeXOSQueryFilterFileListExtDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-f}{#2}{#3}{#4}{ext\string-des}%
}
```

TeXOSQueryFilterRegularFileList Filtered list or regular files.

```
\def\TeXOSQueryFilterRegularFileList#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{}%
}
```

\TeXOSQueryFilterSubDirList Filtered list of sub-directories.

```
\def\TeXOSQueryFilterSubDirList#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{}%
}
```

XOSQueryFilterSubDirListDateAsc Filtered sort of sub-directories by file date.

```
\def\TeXOSQueryFilterSubDirListDateAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{date}%
}
```

eryFilterRegularFileListDateAsc Filtered sort of regular files by file date.

```
\def\TeXOSQueryFilterRegularFileListDateAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{date}%
}
```

XOSQueryFilterSubDirListDateDes Filtered sort of sub-directories by file date in descending order.

```
\def\TeXOSQueryFilterSubDirListDateDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{date\string-des}%
}
```

eryFilterRegularFileListDateDes Filtered sort of regular files by file date in descending order.

```
\def\TeXOSQueryFilterRegularFileListDateDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{date\string-des}%
}
```

XOSQueryFilterSubDirListSizeAsc Filtered sort of sub-directories by file size.

```
\def\TeXOSQueryFilterSubDirListSizeAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{size}%
}
```

eryFilterRegularFileListSizeAsc Filtered sort of regular files by file size.

```
\def\TeXOSQueryFilterRegularFileListSizeAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{size}%
}
```

XOSQueryFilterSubDirListSizeDes Filtered sort of sub-directories by file size in descending order.

```
\def\TeXOSQueryFilterSubDirListSizeDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{size\string-des}%
}
```

eryFilterRegularFileListSizeDes Filtered sort of regular files by file size in descending order.

```
\def\TeXOSQueryFilterRegularFileListSizeDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{size\string-des}%
}
```

XOSQueryFilterSubDirListNameAsc Filtered sort of sub-directories by file name.

```
\def\TeXOSQueryFilterSubDirListNameAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{name}%
}
```

eryFilterRegularFileListNameAsc Filtered sort of regular files by file name.

```
\def\TeXOSQueryFilterRegularFileListNameAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{name}%
}
```

XOSQueryFilterSubDirListNameDes Filtered sort of sub-directories by file name in descending order.

```
\def\TeXOSQueryFilterSubDirListNameDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{name\string-des}%
}
```

eryFilterRegularFileListNameDes Filtered sort of regular files by file name in descending order.

```
\def\TeXOSQueryFilterRegularFileListNameDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{name\string-des}%
}
```

lterSubDirListNameIgnoreCaseAsc Filtered sort of sub-directories by case-insensitive file name.

```
\def\TeXOSQueryFilterSubDirListNameIgnoreCaseAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{iname}%
}
```

egularFileListNameIgnoreCaseAsc Filtered sort of regular files by case-insensitive file name.

```
\def\TeXOSQueryFilterRegularFileListNameIgnoreCaseAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{iname}%
}
```

lterSubDirListNameIgnoreCaseDes Filtered sort of sub-directories by case-insensitive file name in descending order.

```
\def\TeXOSQueryFilterSubDirListNameIgnoreCaseDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{iname\string-des}%
}
```

egularFileListNameIgnoreCaseDes Filtered sort of regular files by case-insensitive file name in descending order.

```
\def\TeXOSQueryFilterRegularFileListNameIgnoreCaseDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{iname\string-des}%
}
```

eXOSQueryFilterSubDirListExtAsc Filtered sort of sub-directories by file extension. (Added for completeness as directories don't tend to have extensions.)

```
\def\TeXOSQueryFilterSubDirListExtAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{ext}%
}
```

ueryFilterRegularFileListExtAsc Filtered sort of regular files by file extension.

```
\def\TeXOSQueryFilterRegularFileListExtAsc#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{ext}%
}
```

eXOSQueryFilterSubDirListExtDes Filtered sort of sub-directories by file extension in descending order.

```
\def\TeXOSQueryFilterSubDirListExtDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fd}{#2}{#3}{#4}{ext\string-des}%
}
```

ueryFilterRegularFileListExtDes Filtered sort of regular files by file extension in descending order.

```
\def\TeXOSQueryFilterRegularFileListExtDes#1#2#3#4{%
 \@texosquery@filterfilelist{#1}{-fr}{#2}{#3}{#4}{ext\string-des}%
}
```

\@texosquery@walk

```
\def\@texosquery@walk#1#2#3#4#5{%
 \TeXOSQuery{#1}%
 {%
   \string-w \string'#2\string' \string'#3\string'
   \string'\texosquerystripquotes{#4}\string' #5%
 }%
}
```

`\TeXOSQueryWalk` Recursive filtered listing of regular files.

```
\def\TeXOSQueryWalk#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{}%
}
```

`\TeXOSQueryWalkDateAsc` As above, but sort by date.

```
\def\TeXOSQueryWalkDateAsc#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{date}%
}
```

`\TeXOSQueryWalkDateDes` As above, but sort by date in descending order.

```
\def\TeXOSQueryWalkDateDes#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{date\string-des}%
}
```

`\TeXOSQueryWalkSizeAsc` As above, but sort by file size.

```
\def\TeXOSQueryWalkSizeAsc#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{size}%
}
```

`\TeXOSQueryWalkSizeDes` As above, but sort by file size in descending order.

```
\def\TeXOSQueryWalkSizeDes#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{size\string-des}%
}
```

`\TeXOSQueryWalkNameAsc` As above, but sort by file name.

```
\def\TeXOSQueryWalkNameAsc#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{name}%
}
```

`\TeXOSQueryWalkNameDes` As above, but sort by file name in descending order.

```
\def\TeXOSQueryWalkNameDes#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{name\string-des}%
}
```

`TeXOSQueryWalkNameIgnoreCaseAsc` As above, but sort by file name (case-insensitive).

```
\def\TeXOSQueryWalkNameIgnoreCaseAsc#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{iname}%
}
```

`TeXOSQueryWalkNameIgnoreCaseDes` As above, but sort by file name (case-insensitive) in descending order.

```
\def\TeXOSQueryWalkNameIgnoreCaseDes#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{iname\string-des}%
}
```

`\TeXOSQueryWalkExtAsc` As above, but sort by file extension.

```
\def\TeXOSQueryWalkExtAsc#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{ext}%
}
```

\TeXOSQueryWalkExtDes As above, but sort by file extension in descending order.

```
\def\TeXOSQueryWalkExtDes#1#2#3#4{%
 \@texosquery@walk{#1}{#2}{#3}{#4}{ext\string-des}%
}
```

\TeXOSQueryFileURI Get the URI of the file given in the second argument.

```
\def\TeXOSQueryFileURI#1#2{\TeXOSQuery{#1}{\string-u
 \string'\texosquerystripquotes{#2}\string'}}
```

\TeXOSQueryFilePath Get the canonical path of the file given in the second argument.

```
\def\TeXOSQueryFilePath#1#2{\TeXOSQuery{#1}{\string-p
 \string'\texosquerystripquotes{#2}\string'}}
```

\TeXOSQueryDirName Get the canonical path of the directory containing the file given in the second argument.

```
\def\TeXOSQueryDirName#1#2{\TeXOSQuery{#1}{\string-e
 \string'\texosquerystripquotes{#2}\string'}}
```

3.1.3 Pattern Formats

There are two basic types of patterns: date/time or numeric. A pattern is stored in a control sequence using custom markup that's easier for TEX to parse than it would be to parse strings in the form `YYYY-MM` or `#,##0`. This internal pattern format can be obtained through capturing the output of `texosquery`'s `-D` action, but patterns can also be constructed using

`\texosquerydefpattern{`⟨*cs*⟩`}{`⟨*pattern specs*⟩`}`

The pattern is stored in ⟨*cs*⟩. The ⟨*pattern specs*⟩ depend on whether a date-time or numeric pattern is required. For a date-time pattern, each date/time element is identified using

`\texosquerydtf{`⟨*n*⟩`}{`⟨*identifier*⟩`}`

where ⟨*identifier*⟩ identifies the element type (such as `M` for month or `s` for seconds) and ⟨*n*⟩ indicates how the element should be formatted, where ⟨*n*⟩ is an integer from 1 to 4. For example if ⟨*n*⟩ is 2 and ⟨*identifier*⟩ is `M`, then this indicates the `MM` format, which produces a two-digit number.

Since it's rather cumbersome to keep typing `\texosquerydtf` and it can make for rather hard to read code, `\texosquerydefpattern` locally redefines `\%` to expand to `\texosquerydtf`. This means that if you do, for example:

```
\texosquerydefpattern{\pattern}{\%2d/\%2M/\%4y}
```

then `\pattern` is defined to

```
\texosquerydtf 2d/\texosquerydtf 2M/\texosquerydtf 4y
```

When simply used within the document, this just expands to the pattern format. For example:

```
Pattern: \pattern.
```

will display "Pattern: dd/MM/yyyy" in the PDF. However, when used with \texosqueryfmtdatetime, the definition of \texosquerydtf changes to reproduce the required date/time element.

For example:

```
\texosquerydefpattern{\pattern}{\%2d/\%2M/\%4y \%2H:\%2m:\%2s \%2Z}

Pattern: \pattern.

\TeXOSQueryDateTime{\datetimedata}

\ifx\datetimedata\empty
 Query Failed!
\else
 \expandafter\texosqueryfmtdatetime\expandafter\pattern\datetimedata
\fi
```

The numeric patterns are rather more complicated. The ⟨*pattern specs*⟩ now needs to use the following formats:

\texosquerypatnum{⟨*+ve/-ve numeric pattern*⟩}

This is a numeric pattern applied to a number regardless of whether the number is positive or negative. (If negative, the minus sign is automatically inserted.) This is rather a long and cumbersome command to type, so \texosquerydefpattern locally defines \numfmt to expand to it.

\texosquerypatplusminus{⟨*+ve numeric pattern*⟩}{⟨*-ve numeric pattern*⟩}

This provides a pattern ⟨*+ve numeric pattern*⟩ to use if the number is positive and a patter ⟨*-ve numeric pattern*⟩ to use if the number is negative. Again \texosquerydefpattern locally defines a shortcut, \pmnumfmt, to expand to this command.

\texosquerypatsinum{⟨*decimal pattern*⟩}{⟨*mantissa pattern*⟩}

This provides a pattern to use for SI numbers where ⟨*decimal pattern*⟩ is a pattern for the decimal number part (before the exponent symbol) and ⟨*mantissa*⟩ is the pattern for the integer part in the mantissa (after the exponent symbol). The locally defined shortcut is \sinumfmt. The ⟨*decimal pattern*⟩ will typically be in the form:

\texosquerypatdec{⟨*integer pattern*⟩}{⟨*fraction pattern*⟩}

This indicates a decimal pattern where the ⟨*integer pattern*⟩ is applied to the part before the decimal separator and ⟨*fraction pattern*⟩ is applied to the part after the separator. The shortcut is \decfmt.

\texosquerypatprefixcurrency{⟨*decimal pattern*⟩}{⟨*text*⟩}

This indicates a currency pattern with a prefixed currency symbol where ⟨*text*⟩ is inserted before the currency symbol. The shortcut is \pcur. The ⟨*decimal pattern*⟩ will typically use \texosquerypatdec{⟨*int*⟩}⟨*frac*⟩. (Similarly for the following.)

`\texosquerypatprefixicurrency{`⟨*decimal pattern*⟩`}{`⟨*text*⟩`}`

This indicates an international currency pattern with a prefixed international currency symbol where ⟨*text*⟩ is inserted before the symbol. The shortcut is `\picur`.

`\texosquerypatsuffixcurrency{`⟨*decimal pattern*⟩`}{`⟨*text*⟩`}`

This indicates a currency pattern with a suffixed international currency symbol where ⟨*text*⟩ is inserted after the currency symbol. The shortcut is `\scur`.

`\texosquerypatsuffixicurrency{`⟨*decimal pattern*⟩`}{`⟨*text*⟩`}`

This indicates an international currency pattern with a suffixed international currency symbol where ⟨*text*⟩ is inserted after the symbol. The shortcut is `\sicur`.

`\texosquerypatprefixpercent{`⟨*decimal pattern*⟩`}{`⟨*text*⟩`}`

This indicates a percentage pattern with a prefixed percent symbol where ⟨*text*⟩ is inserted before the symbol. The shortcut is `\ppct`.

`\texosquerypatsuffixpercent{`⟨*decimal pattern*⟩`}{`⟨*text*⟩`}`

This indicates a percentage pattern with a suffixed percent symbol where ⟨*text*⟩ is inserted after the symbol. The shortcut is `\spct`.

`\texosquerypatprefixpermill{`⟨*decimal pattern*⟩`}{`⟨*text*⟩`}`

This indicates a per-mill pattern with a prefixed per-mill symbol where ⟨*text*⟩ is inserted before the symbol. The shortcut is `\ppml`.

`\texosquerypatsuffixpermill{`⟨*decimal pattern*⟩`}{`⟨*text*⟩`}`

This indicates a per-mill pattern with a suffixed per-mill symbol where ⟨*text*⟩ is inserted after the symbol. The shortcut is `\spml`.

> **Important Note:** The integer parts ⟨*integer part*⟩, ⟨*fraction part*⟩ and ⟨*mantissa*⟩ must have *exactly* ten digit identifiers. (TEX can't reach 11 digit numbers.)

There are two types of digit identifiers:

`\texosquerypatdigit`

This indicates a digit that must be displayed, even if it's not significant (for example a leading zero). The shortcut command is `\0` (backslash zero).

`\texosquerypatdigitnozero`

This indicates a digit that should only be displayed if it's significant. (For example, if it's a leading zero, it's not shown.) The shortcut command is `\#` (backslash hash).

The number group separator can be inserted using

```
\texosquerypatgroupsep
```

The shortcut command is `\,` (backslash comma).

The sign can be inserted using

```
\texosquerypatminus
```

This ensures the sign is displayed even if the number is positive. The shortcut command is `\-` (backslash hyphen).

Here's an example of a decimal pattern:

```
\texosquerydefpattern{\numpattern}{%
 \numfmt{\decfmt{\#\,\#\#\#\,\#\#\#\,\#\#\0}{\0\#\#\#\#\#\#\#\#\#}}}
```

The pattern can be applied to a number using `\texosqueryfmtnumber`:

```
\texosqueryfmtnumber{\numpattern}{123}{4567}{2}
```

which produces: 12,345.67 (the group and decimal separators can be redefined as appropriate).

Here's an example of a scientific number:

```
\texosquerydefpattern{\sinumpattern}{%
 \sinumfmt
 {\decfmt{\#\,\#\#\#\,\#\#\#\,\#\#\0}{\0\#\#\#\#\#\#\#\#\#}}%
 {\-\#\#\#\#\#\#\#\#\0\0}%
}
```

The pattern can be applied to a number:

```
\texosqueryfmtnumber{\sinumpattern}{1}{234567}{3}
```

which produces: 1.234567E+03

Here's an integer pattern:

```
\texosquerydefpattern{\intpattern}{%
 \patnumfmt{\#\,\#\#\#\,\#\#\#\,\#\#\0}}
```

The pattern applied to a number:

```
\texosqueryfmtnumber{\intpattern}{123}{4567}{2}
```

which produces: 12,345 (the fractional part has been omitted).

Here's a currency pattern that applies a different format for positive and negative numbers:

```
\texosquerydefpattern{\curpattern}{%
 \pmnumfmt
 {\pcur{\decfmt{\#\,\#\#\#\,\#\#\#\,\#\#\0}{\0\0\#\#\#\#\#\#\#\#}}{}}%
 {\pcur{\decfmt{\#\,\#\#\#\,\#\#\#\,\#\#\0}{\0\0\#\#\#\#\#\#\#\#}}{\-}}}
```

This uses the ⟨*text*⟩ part of `\pcur` to insert the sign before the currency symbol (but only for negative values).

```
\texosqueryfmtnumber{\curpattern}{-1234567}{0}{0}
```

This produces: −$12,345,678.00 (again the symbol and separators can be redefined as appropriate).

Here's an example of a percentage pattern:

```
\texosquerydefpattern{\pcpattern}{%
 \numfmt{\spct{\#\,\#\#\#\,\#\#\#\,\#\#\0}{}}}
```

The pattern can similarly be applied to a number using `\texosqueryfmtnumber`.

`\texosquerydtf` Date/time format placeholder. The second argument is the placeholder character and the first argument is the number of occurrences of that character in the placeholder. The default definition just converts it back to pattern format used by Java's SimpleDateFormat class. The pattern interprets ≥ 4 as a single case, so this will only produce a maximum of four characters.

```
\def\texosquerydtf#1#2{%
  \ifcase#1
  \or
   #2%
  \or
   #2#2%
  \or
   #2#2#2%
  \else
   #2#2#2#2%
  \fi
}
```

The following commands are used to display the pattern in the document text to reproduce the pattern string recognised by Java. This is provided for debugging to check the pattern. In most cases the pattern will be applied to a number rather than simply displayed.

`\texosquerypatstr` Quoted string contained in number format.

```
\def\texosquerypatstr#1{'#1'}
```

`\texosquerypatquote` Literal quote contained in number format.

```
\def\texosquerypatquote{''}
```

Number format place holders.

`\texosquerypatplusminus`

```
\def\texosquerypatplusminus#1#2{#1;#2}
```

`\texosquerypatnum`

```
\def\texosquerypatnum#1{#1}
```

`\texosquerypatsinum`

```
\def\texosquerypatsinum#1#2{#1E#2}
```

`\texosquerypatdec`

```
\def\texosquerypatdec#1#2{#1.#2}
```

\texosquerypatprefixcurrency First argument is a number, the second is optional text before the currency symbol. This will require UTF-8 support otherwise it will need redefining as appropriate. (Similarly for the other currency commands and for the per-mill commands.)

```
\def\texosquerypatprefixcurrency#1#2{#2¤#1}
```

\texosquerypatprefixicurrency As above but use international currency symbol.

```
\def\texosquerypatprefixicurrency#1#2{#2¤¤#1}
```

\texosquerypatsuffixcurrency First argument is a number, the second is optional text after the currency symbol.

```
\def\texosquerypatsuffixcurrency#1#2{#1¤#2}
```

\texosquerypatsuffixicurrency As above but use international currency symbol.

```
\def\texosquerypatsuffixicurrency#1#2{#1¤¤#2}
```

\texosquerypatdigit

```
\def\texosquerypatdigit{0}
```

\texosquerypatdigitnozero

```
\def\texosquerypatdigitnozero{\#}
```

\texosquerypatminus

```
\def\texosquerypatminus{-}
```

\texosquerypatgroupsep

```
\def\texosquerypatgroupsep{,}
```

\texosquerypatprefixpercent The first argument is the value, the second argument is optional text before the percent symbol.

```
\def\texosquerypatprefixpercent#1#2{#2\%#1}
```

\texosquerypatsuffixpercent The first argument is the value, the second argument is optional text after the percent symbol.

```
\def\texosquerypatsuffixpercent#1#2{#1\%#2}
```

\texosquerypatprefixpermill The first argument is the value, the second argument is optional text before the per-mill symbol.

```
\def\texosquerypatprefixpermill#1#2{#2‰#1}
```

\texosquerypatsuffixpermill The first argument is the value, the second argument is optional text after the per-mill symbol.

```
\def\texosquerypatsuffixpermill#1#2{#1‰#2}
```

\@texosquery@pattern@shortcuts Provide much shorter cuts for the convenience of directly defining patterns with \texosquerydefpattern.

```
\def\@texosquery@pattern@shortcuts{%
    \def\%{\noexpand\texosquerydtf}%
    \def\0{\noexpand\texosquerypatdigit}%
```

```
    \def\#{\noexpand\texosquerypatdigitnozero}%
    \def\-{\noexpand\texosquerypatminus}%
    \def\,{\noexpand\texosquerypatgroupsep}%
    \def\numfmt{\noexpand\texosquerypatnum}%
    \def\pmnumfmt{\noexpand\texosquerypatplusminus}%
    \def\sinumfmt{\noexpand\texosquerypatsinum}%
    \def\decfmt{\noexpand\texosquerypatdec}%
    \def\pcur{\noexpand\texosquerypatprefixcurrency}%
    \def\picur{\noexpand\texosquerypatprefixicurrency}%
    \def\scur{\noexpand\texosquerypatsuffixcurrency}%
    \def\sicur{\noexpand\texosquerypatsuffixicurrency}%
    \def\ppct{\noexpand\texosquerypatprefixpercent}%
    \def\spct{\noexpand\texosquerypatsuffixpercent}%
    \def\ppml{\noexpand\texosquerypatprefixpermill}%
    \def\spml{\noexpand\texosquerypatsuffixpermill}%
}
```

`\texosquerydefpattern` Define a new pattern using the shortcut markup. The first argument is the name of the control sequence in which to store the pattern provided in the second argument. Be careful of any fragile commands within the second argument. They will need protecting!

```
\def\texosquerydefpattern#1#2{%
 \begingroup
  \@texosquery@pattern@shortcuts
  \@texosquery@edef\x{\endgroup\def\noexpand#1{#2}}\x
}
```

3.1.4 Applying Date-Time Patterns

In order to apply date-time patterns, we need all the information about the date or time we're trying to format.

1. Era needed by the `G` designator. Java identifies the era by an integer (0 = BC and 1 = AD).
2. Era text (e.g. AD) can be supplied by a macro.
3. Year needed by the `y` designator.
4. Week year needed by the `Y` designator.
5. Month in year needed by the `M` or `L` designators.
6. Month name needed by the `M` or `L` designators. This can be provided as macros that convert the month number to the name. Four macros are needed: short, full, standalone short and standalone full.
7. Week in year needed by the `w` designator.
8. Week in month needed by the `W` designator.
9. Day in year needed by the `D` designator.

10. Day in month needed by the `d` designator.
11. Day of week in month needed by the `F` designator.
12. Day name in week needed by the `E` designator. This can be provided as a macros that accepts the day of week number. The full form is needed for 4 letter patterns otherwise a short form.
13. Day number of week (1 = Monday, 7 = Sunday) needed by the `u` designator. This means that the above day of week name macros needs to use Monday=1 base indexing. This means that `\pgfcalendarweekdayname` can't be used directly.
14. AM/PM identifier needed by the `a` designator. Assume 0 = AM and 1 = PM to match Java.
15. AM/PM text can be provided by a macro.
16. Hour of the day (0-23) needed by the `H` designator.
17. Hour in day (1-24) needed by the `k` designator.
18. Hour in am/pm (0-11) needed by the `K` designator.
19. Hour in am/pm (1-12) needed by the `h` designator.
20. Minute in hour needed by the `m` designator.
21. Second in minute needed by the `s` designator.
22. Millisecond needed by the `S` designator.
23. Time zone needed by the `z`, `Z` and `X` designators. This will require macros for converting the time zone to each of those formats.

Supply a general utility command that has enough arguments to pass all the above information. A higher level user command can then be provided that determines all the arguments to provide an easier interface.

The arguments need to be the pattern followed by `{`⟨*era id*⟩`}{`⟨*year*⟩`}{`⟨*week year*⟩`}{`⟨*month*⟩`}{`⟨*week in year*⟩`}{`⟨*week in month*⟩`}{`⟨*day in year*⟩`}{`⟨*day in month*⟩`}{`⟨*day of week in month*⟩`}{`⟨*day number of week*⟩`}{`⟨*am/pm id*⟩`}{`⟨*hour of day (H)*⟩`}{`⟨*hour in day (k)*⟩`}{`⟨*hour in am/pm (K)*⟩`}{`⟨*hour in am/pm (h)*⟩`}{`⟨*minute in hour*⟩`}{`⟨*second in minute*⟩`}{`⟨*millisecond*⟩`}{`⟨*time zone*⟩`}` The arguments must all be integers except for the time zone which must be in the form `{`⟨*TZh*⟩`}{`⟨*TZm*⟩`}{`⟨*id*⟩`}{`⟨*dst flag*⟩`}`. These are the time zone hour and min offsets, time zone ID and daylight saving flag (1 if daylight saving in effect otherwise 0). These arguments can all be obtained using the `--date-time` / `-M` action.

We'll need some helper macros to get around the nine argument maximum limit.

`\texosqueryfmtdatetime` The general utility command to format a pattern. The first argument is the pattern. After that are the date-time data arguments.

```
\def\texosqueryfmtdatetime#1{%
  \def\@texosquery@fmt@dt@pattern{#1}%
  \@texosquery@fmt@getera
}
```

All the remaining arguments except for the time zone must be integers. These are padded using `\@texosquery@paddigits`.

`\@texosquery@fmt@getera`

```
\def\@texosquery@fmt@getera#1{%
  \edef\@texosquery@fmt@G{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getyear
}
```

`\@texosquery@fmt@getyear`

```
\def\@texosquery@fmt@getyear#1{%
  \edef\@texosquery@fmt@y{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getweekyear
}
```

`\@texosquery@fmt@getweekyear`

```
\def\@texosquery@fmt@getweekyear#1{%
  \edef\@texosquery@fmt@Y{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getmonth
}
```

`\@texosquery@fmt@getmonth`

```
\def\@texosquery@fmt@getmonth#1{%
  \edef\@texosquery@fmt@M{\@texosquery@paddigits{#1}}%
  \let\@texosquery@fmt@L\@texosquery@fmt@M
  \@texosquery@fmt@getweekinyear
}
```

`\@texosquery@fmt@getweekinyear`

```
\def\@texosquery@fmt@getweekinyear#1{%
  \edef\@texosquery@fmt@w{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getweekinmonth
}
```

`\@texosquery@fmt@getweekinmonth`

```
\def\@texosquery@fmt@getweekinmonth#1{%
  \edef\@texosquery@fmt@W{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getdayinyear
}
```

`\@texosquery@fmt@getdayinyear`

```
\def\@texosquery@fmt@getdayinyear#1{%
  \edef\@texosquery@fmt@D{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getdayinmonth
}
```

`\@texosquery@fmt@getdayinmonth`

```
\def\@texosquery@fmt@getdayinmonth#1{%
  \edef\@texosquery@fmt@d{\@texosquery@paddigits{#1}}%
```

```
    \@texosquery@fmt@getdayofweekinmonth
}

osquery@fmt@getdayofweekinmonth

\def\@texosquery@fmt@getdayofweekinmonth#1{%
  \edef\@texosquery@fmt@F{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getdaynumberofweek
}

xosquery@fmt@getdaynumberofweek

\def\@texosquery@fmt@getdaynumberofweek#1{%
  \edef\@texosquery@fmt@u{\@texosquery@paddigits{#1}}%
  \let\@texosquery@fmt@E\@texosquery@fmt@u
  \@texosquery@fmt@getampm
}

      \@texosquery@fmt@getampm

\def\@texosquery@fmt@getampm#1{%
  \edef\@texosquery@fmt@a{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@gethourindayH
}

 \@texosquery@fmt@gethourindayH

\def\@texosquery@fmt@gethourindayH#1{%
  \edef\@texosquery@fmt@H{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@gethourindayk
}

 \@texosquery@fmt@gethourindayk

\def\@texosquery@fmt@gethourindayk#1{%
  \edef\@texosquery@fmt@k{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@gethourinampmK
}

\@texosquery@fmt@gethourinampmK

\def\@texosquery@fmt@gethourinampmK#1{%
  \edef\@texosquery@fmt@K{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@gethourinampmh
}

\@texosquery@fmt@gethourinampmh

\def\@texosquery@fmt@gethourinampmh#1{%
  \edef\@texosquery@fmt@h{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getminute
}

     \@texosquery@fmt@getminute

\def\@texosquery@fmt@getminute#1{%
  \edef\@texosquery@fmt@m{\@texosquery@paddigits{#1}}%
```

```
  \@texosquery@fmt@getsecond
}
```

`\@texosquery@fmt@getsecond`

```
\def\@texosquery@fmt@getsecond#1{%
  \edef\@texosquery@fmt@s{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@getmillisecond
}
```

`\@texosquery@fmt@getmillisecond`

```
\def\@texosquery@fmt@getmillisecond#1{%
  \edef\@texosquery@fmt@S{\@texosquery@paddigits{#1}}%
  \@texosquery@fmt@gettimezone
}
```

`\@texosquery@fmt@gettimezone`

```
\def\@texosquery@fmt@gettimezone#1{%
  \def\@texosquery@fmt@Z{#1}%
  \def\@texosquery@fmt@z{#1}%
  \def\@texosquery@fmt@X{#1}%
```

All data now supplied. Temporarily redefine pattern markup and process the pattern.

```
  \begingroup
    \@texosquery@setup@dtpattern
    \@texosquery@fmt@dt@pattern
  \endgroup
}
```

`\@texosquery@setup@dtpattern`

```
\def\@texosquery@setup@dtpattern{%
  \let\texosquerydtf\@texosquery@fmt@dtf
  \let\texosquerypatstr\texosquerypatfmtstr
  \let\texosquerypatquote\texosquerypatfmtquote
}
```

`\@texosquery@paddigits@pos` Pad positive number to 10 digits. TEX can't reach 11 digits, so this is the maximum representation.

```
\def\@texosquery@paddigits@pos#1{%
  \ifnum#1<10
   000000000\number#1
  \else
   \ifnum#1<100
    00000000\number#1
   \else
     \ifnum#1<1000
      0000000\number#1
     \else
       \ifnum#1<10000
        000000\number#1
```

```
      \else
        \ifnum#1<100000
         00000\number#1
        \else
          \ifnum#1<1000000
           0000\number#1
          \else
            \ifnum#1<10000000
             000\number#1
            \else
              \ifnum#1<100000000
               00\number#1
              \else
                \ifnum#1<1000000000
                 0\number#1
                \else
                  \number#1
                \fi
              \fi
            \fi
          \fi
        \fi
      \fi
    \fi
  \fi
}
```

`\@texosquery@paddigits` This will expand to 11 characters (sign followed by 10 digits).

```
\def\@texosquery@paddigits#1{%
 \ifnum#1<0
```

Move the minus sign outside.

```
   -\expandafter\@texosquery@paddigits@pos\expandafter
     {\@texosquery@gobble#1}%
 \else
   +\@texosquery@paddigits@pos{#1}%
 \fi
}
```

`\@texosquery@paddigits@trailing` Pad trailing zeros.

```
\def\@texosquery@paddigits@trailing#1{%
  \expandafter\@texosquery@tenoften@then@gobble
    #10000000000\@texosquery@end@tenoften
}
```

`texosquery@tenoften@then@gobble`

```
\def\@texosquery@tenoften@then@gobble#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4#5#6#7#8#9%
  \@texosquery@lastoften@gobble
```

```
}
```

`\@texosquery@lastoften@gobble`

```
\def\@texosquery@lastoften@gobble#1#2\@texosquery@end@tenoften{#1}
```

Provide commands to select certain digits. (Sign not included.)

`\@texosquery@firstoften` First of ten.

```
\def\@texosquery@firstoften#1#2#3#4#5#6#7#8#9{%
  #1%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@secondoften` Second of ten.

```
\def\@texosquery@secondoften#1#2#3#4#5#6#7#8#9{%
  #2%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@thirdoften` Third of ten.

```
\def\@texosquery@thirdoften#1#2#3#4#5#6#7#8#9{%
  #3%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@fourthoften` Fourth of ten.

```
\def\@texosquery@fourthoften#1#2#3#4#5#6#7#8#9{%
  #4%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@fifthoften` Fifth of ten.

```
\def\@texosquery@fifthoften#1#2#3#4#5#6#7#8#9{%
  #5%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@sixthoften` Sixth of ten.

```
\def\@texosquery@sixthoften#1#2#3#4#5#6#7#8#9{%
  #6%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@seventhoften` Seventh of ten.

```
\def\@texosquery@seventhoften#1#2#3#4#5#6#7#8#9{%
  #7%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@eighthoften` Eighth of ten.

```
\def\@texosquery@eighthoften#1#2#3#4#5#6#7#8#9{%
  #8%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@ninthoften` Ninth of ten.

```
\def\@texosquery@ninthoften#1#2#3#4#5#6#7#8#9{%
  #9%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`\@texosquery@tenthoften` Tenth of ten.

```
\def\@texosquery@tenthoften#1#2#3#4#5#6#7#8#9{%
  \@texosquery@firstofone
}
```

Now macros to select first n of ten.

`@texosquery@firsttwooften` First two of ten.

```
\def\@texosquery@firsttwooften#1#2#3#4#5#6#7#8#9{%
  #1#2%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@firstthreeoften` First three of ten.

```
\def\@texosquery@firstthreeoften#1#2#3#4#5#6#7#8#9{%
  #1#2#3%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@firstfouroften` First four of ten.

```
\def\@texosquery@firstfouroften#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@firstfiveoften` First five of ten.

```
\def\@texosquery@firstfiveoften#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4#5%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@firstsixoften` First six of ten.

```
\def\@texosquery@firstsixoften#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4#5#6%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@firstsevenoften` First seven of ten.

```
\def\@texosquery@firstsevenoften#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4#5#6#7%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@firsteightoften` First eight of ten.

```
\def\@texosquery@firsteightoften#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4#5#6#7#8%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@firstnineoften` First nine of ten.

```
\def\@texosquery@firstnineoften#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4#5#6#7#8#9%
```

Grab tenth argument and discard.

```
  \@texosquery@gobble
}
```

`@texosquery@alltenoften` All ten.

```
\def\@texosquery@alltenoften#1#2#3#4#5#6#7#8#9{%
  #1#2#3#4#5#6#7#8#9%
  \@texosquery@firstofone
}
```

Select last n of ten.

`@texosquery@lasttwooften` Last two of ten.

```
\def\@texosquery@lasttwooften#1#2#3#4#5#6#7#8#9{%
 #9%
 \@texosquery@firstofone
}
```

`@texosquery@lastthreeoften` Last three of ten.

```
\def\@texosquery@lastthreeoften#1#2#3#4#5#6#7#8#9{%
 #8#9%
 \@texosquery@firstofone
}
```

`@texosquery@lastfouroften` Last four of ten.

```
\def\@texosquery@lastfouroften#1#2#3#4#5#6#7#8#9{%
 #7#8#9%
 \@texosquery@firstofone
}
```

`@texosquery@lastfiveoften` Last five of ten.

```
\def\@texosquery@lastfiveoften#1#2#3#4#5#6#7#8#9{%
 #6#7#8#9%
 \@texosquery@firstofone
}
```

`@texosquery@lastsixoften` Last six of ten.

```
\def\@texosquery@lastsixoften#1#2#3#4#5#6#7#8#9{%
 #5#6#7#8#9%
 \@texosquery@firstofone
}
```

`@texosquery@lastsevenoften` Last seven of ten.

```
\def\@texosquery@lastsevenoften#1#2#3#4#5#6#7#8#9{%
 #4#5#6#7#8#9%
 \@texosquery@firstofone
}
```

`@texosquery@lasteightoften` Last eight of ten.

```
\def\@texosquery@lasteightoften#1#2#3#4#5#6#7#8#9{%
 #3#4#5#6#7#8#9%
 \@texosquery@firstofone
}
```

`@texosquery@lastnineoften` Last nine of ten.

```
\def\@texosquery@lastnineoften#1#2#3#4#5#6#7#8#9{%
 #2#3#4#5#6#7#8#9%
 \@texosquery@firstofone
}
```

`\@texosquery@fmtminus` Minus symbol for use in date-time patterns.

```
\def\@texosquery@fmtminus{\texosquerypatfmtminus}
```

`\@texosquery@fmtplus` Plus symbol for use in date-time patterns. Omit by default.

```
\def\@texosquery@fmtplus{}
```

`\@texosquery@fmtsign` Plus or minus sign for use in date-time patterns.

```
\def\@texosquery@fmtsign#1{%
 \ifx#1+\@texosquery@fmtplus\else\@texosquery@fmtminus\fi
}
```

`\@texosquery@atleastonedigit` At least one digit with leading zeros removed.

```
\def\@texosquery@atleastonedigit#1{%
  \ifnum#1<0
   \@texosquery@fmtminus\number-#1
  \else
   \number#1
  \fi
}
```

`\@texosquery@atleastfourdigits` At least four digits, possible padded with zeros to make up four. The first argument is the sign, then follow the ten digits.

```
\def\@texosquery@atleastfourdigits#1{%
  \@texosquery@at@leastfourdigits#1\@texosquery@end@atleastfourdigits
}
```

`\@texosquery@at@leastfourdigits` At least four digits, possible padded with zeros to make up four. The first argument is the sign, then follow the ten digits.

```
\def\@texosquery@at@leastfourdigits#1#2\@texosquery@end@atleastfourdigits{%
  \@texosquery@fmtsign{#1}%
  \ifnum#2<1000
    \@texosquery@lastfouroften#2%
  \else
    \number#2
  \fi
}
```

`\@texosquery@threedigitsexactly` Exactly three digits.

```
\def\@texosquery@threedigitsexactly#1{%
  \@texosquery@threedigits@exactly#1\@texosquery@threedigits@exactly
}%
```

`@texosquery@threedigits@exactly` Exactly three digits.

```
\def\@texosquery@threedigits@exactly#1#2\@texosquery@threedigits@exactly{%
  \@texosquery@fmtsign{#1}%
  \@texosquery@lastthreeoften#2%
}%
```

\@texosquery@twodigitsexactly Exactly two digits.

```
\def\@texosquery@twodigitsexactly#1{%
  \@texosquery@twodigits@exactly#1\@texosquery@twodigits@exactly
}%
```

\@texosquery@twodigits@exactly Exactly two digits.

```
\def\@texosquery@twodigits@exactly#1#2\@texosquery@twodigits@exactly{%
  \@texosquery@fmtsign{#1}%
  \@texosquery@lasttwooften#2%
}%
```

\@texosquery@fmt@dtf `\@texosquery@fmt@dtf{`⟨*n*⟩`}{`⟨*designator*⟩`}`

When formatting a date-time pattern `\texosquerydtf` will temporarily be redefined to this command. This command indicates the format obtained by ⟨*n*⟩ instances of ⟨*designator*⟩. For example, `{2}{M}` indicates the format `MM`. This command tests for `\texosqueryfmtpat`⟨*format*⟩, which should take a single argument. If defined, that's used, otherwise use one of the numeric commands defined above. The locale package defines `\texosqueryfmtpatMMM` and so on to use the locale's month names etc.

```
\def\@texosquery@fmt@dtf#1#2{%
  \@texosquery@ifundef{@texosquery@fmt@#2}%
  {\@texosquery@warn{Unknown date-time pattern designator '#2'}}%
  {%
    \ifcase#1
    \or
      \@texosquery@ifundef{texosqueryfmtpat#2}%
      {%
        \expandafter\expandafter\expandafter
          \@texosquery@atleastonedigit
          \expandafter\expandafter\expandafter
         {\csname @texosquery@fmt@#2\endcsname}%
      }%
      {%
        \csname texosqueryfmtpat#2\expandafter\expandafter\expandafter\endcsname
         \expandafter\expandafter\expandafter
         {\csname @texosquery@fmt@#2\endcsname}%
      }%
    \or
      \@texosquery@ifundef{texosqueryfmtpat#2#2}%
      {%
        \expandafter\expandafter\expandafter
          \@texosquery@twodigitsexactly
          \expandafter\expandafter\expandafter
          {\csname @texosquery@fmt@#2\endcsname}%
      }%
      {%
        \csname texosqueryfmtpat#2#2\expandafter\expandafter\expandafter
        \endcsname \expandafter\expandafter\expandafter
         {\csname @texosquery@fmt@#2\endcsname}%
      }%
```

```
    \or
      \@texosquery@ifundef{texosqueryfmtpat#2#2#2}%
      {%
        \expandafter\expandafter\expandafter
          \@texosquery@threedigitsexactly
          \expandafter\expandafter\expandafter
        {\csname @texosquery@fmt@#2\endcsname}%
      }%
      {%
        \csname texosqueryfmtpat#2#2#2\expandafter\expandafter\expandafter
        \endcsname \expandafter\expandafter\expandafter
         {\csname @texosquery@fmt@#2\endcsname}%
      }%
    \else
      \@texosquery@ifundef{texosqueryfmtpat#2#2#2#2}%
      {%
        \expandafter\expandafter\expandafter
          \@texosquery@atleastfourdigits
          \expandafter\expandafter\expandafter
          {\csname @texosquery@fmt@#2\endcsname}%
      }%
      {%
        \csname texosqueryfmtpat#2#2#2#2\expandafter\expandafter\expandafter
        \endcsname \expandafter\expandafter\expandafter
         {\csname @texosquery@fmt@#2\endcsname}%
      }%
    \fi
  }%
}
```

Provide default commands for the time zone designators, since the time zone isn't supplied as a single integer.

`\texosqueryfmttimezonehr` Allow for -0 so append 1 to hour in test.

```
\def\texosqueryfmttimezonehr#1{%
  \ifnum#11<0\@texosquery@fmtminus
    \ifnum#1>-10 0\fi\number-#1
  \else
   +\ifnum#1<10 0\fi\number#1
  \fi
}
```

`\texosqueryfmttimezonenumhr` Like the above, but don't zero-pad or prefix with plus sign.

```
\def\texosqueryfmttimezonenumhr#1{%
  \ifnum#11<0\@texosquery@fmtminus
    \number-#1
  \else
    \number#1
  \fi
}
```

\texosqueryfmttimezonemin

```
\def\texosqueryfmttimezonemin#1{%
  \ifnum#1<10 0\fi\number#1
}
```

\@texosquery@firstoffour

```
\def\@texosquery@firstoffour#1#2#3#4{#1}
```

\@texosquery@secondoffour

```
\def\@texosquery@secondoffour#1#2#3#4{#2}
```

\@texosquery@thirdoffour

```
\def\@texosquery@thirdoffour#1#2#3#4{#3}
```

\@texosquery@fourthoffour

```
\def\@texosquery@fourthoffour#1#2#3#4{#4}
```

\texosqueryshorttimezone Maps id to short time zone display name. This will need redefining as appropriate. The default simply expands to the ID. Mappings can be obtained for a particular locale using the `-Z` or `--time-zones` action.

```
\def\texosqueryshorttimezone#1{#1}
```

\texosqueryshortdstzone Maps id to short daylight saving time zone display name. This will need redefining as appropriate. The default simply expands to the ID followed by (DST).

```
\def\texosqueryshortdstzone#1{#1 (DST)}
```

\texosquerylongtimezone Maps id to long time zone display name. This will need redefining as appropriate. The default simply expands to the ID.

```
\def\texosquerylongtimezone#1{#1}
```

\texosquerylongdstzone Maps id to long daylight saving time zone display name. This will need redefining as appropriate. The default simply expands to the ID followed by (DST).

```
\def\texosquerylongdstzone#1{#1 (DST)}
```

\texosquerytimesep

```
\def\texosquerytimesep{:}
```

\texosqueryfmtpatz Default time zone format for `z` designator.

```
\def\texosqueryfmtpatz#1{%
 \expandafter\ifnum\@texosquery@fourthoffour#1=0
  \expandafter\texosqueryshorttimezone\expandafter{\@texosquery@thirdoffour#1}%
 \else
  \expandafter\texosqueryshortdstzone\expandafter{\@texosquery@thirdoffour#1}%
 \fi
}
```

\texosqueryfmtpatzz Default time zone format for zz designator.

```
\def\texosqueryfmtpatzz#1{%
 \expandafter\ifnum\@texosquery@fourthoffour#1=0
   \expandafter\texosqueryshorttimezone\expandafter{\@texosquery@thirdoffour#1}%
 \else
   \expandafter\texosqueryshortdstzone\expandafter{\@texosquery@thirdoffour#1}%
 \fi
}
```

\texosqueryfmtpatzzz Default time zone format for zzz designator.

```
\def\texosqueryfmtpatzzz#1{%
 \expandafter\ifnum\@texosquery@fourthoffour#1=0
   \expandafter\texosquerylongtimezone\expandafter{\@texosquery@thirdoffour#1}%
 \else
   \expandafter\texosquerylongdstzone\expandafter{\@texosquery@thirdoffour#1}%
 \fi
}
```

\texosqueryfmtpatzzzz Default time zone format for zzzz designator.

```
\def\texosqueryfmtpatzzzz#1{%
 \expandafter\ifnum\@texosquery@fourthoffour#1=0
   \expandafter\texosquerylongtimezone\expandafter{\@texosquery@thirdoffour#1}%
 \else
   \expandafter\texosquerylongdstzone\expandafter{\@texosquery@thirdoffour#1}%
 \fi
}
```

\texosqueryfmtpatZ Default time zone format for Z designator.

```
\def\texosqueryfmtpatZ#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
   {\@texosquery@firstoffour#1}%
  \expandafter\texosqueryfmttimezonemin\expandafter
   {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpatZZ Default time zone format for ZZ designator.

```
\def\texosqueryfmtpatZZ#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
    {\@texosquery@firstoffour#1}%
  \expandafter\texosqueryfmttimezonemin\expandafter
    {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpatZZZ Default time zone format for ZZZ designator.

```
\def\texosqueryfmtpatZZZ#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
    {\@texosquery@firstoffour#1}%
  \expandafter\texosqueryfmttimezonemin\expandafter
    {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpatZZZZ Default time zone format for ZZZZ designator.

```
\def\texosqueryfmtpatZZZZ#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
    {\@texosquery@firstoffour#1}%
  \expandafter\texosqueryfmttimezonemin\expandafter
    {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpatX Default time zone format for X designator.

```
\def\texosqueryfmtpatX#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
    {\@texosquery@firstoffour#1}%
  \texosquerytimesep
  \expandafter\texosqueryfmttimezonemin\expandafter
    {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpatXX Default time zone format for XX designator.

```
\def\texosqueryfmtpatXX#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
    {\@texosquery@firstoffour#1}%
  \texosquerytimesep
  \expandafter\texosqueryfmttimezonemin\expandafter
    {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpatXXX Default time zone format for XXX designator.

```
\def\texosqueryfmtpatXXX#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
    {\@texosquery@firstoffour#1}%
  \texosquerytimesep
  \expandafter\texosqueryfmttimezonemin\expandafter
    {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpatXXXX Default time zone format for XXXX designator.

```
\def\texosqueryfmtpatXXXX#1{%
  \expandafter\texosqueryfmttimezonehr\expandafter
    {\@texosquery@firstoffour#1}%
  \texosquerytimesep
  \expandafter\texosqueryfmttimezonemin\expandafter
    {\@texosquery@secondoffour#1}%
}
```

\texosqueryfmtpata Default am/pm designator for the a designator.

```
\def\texosqueryfmtpata#1{%
 \ifnum#1=0 AM\else PM\fi
}
```

\texosqueryfmtpataa Default am/pm for the aa designator. Just make it the same as the a designator.

```
\def\texosqueryfmtpataa{\texosqueryfmtpata}
```

\texosqueryfmtpataaa Default am/pm for the aaa designator. Just make it the same as the a designator.

```
\def\texosqueryfmtpataaa{\texosqueryfmtpata}
```

\texosqueryfmtpataaaa Default am/pm for the aaaa designator. Just make it the same as the a designator.

```
\def\texosqueryfmtpataaaa{\texosqueryfmtpata}
```

\texosqueryfmtpatG Default era designator for the G designator.

```
\def\texosqueryfmtpatG#1{%
 \ifnum#1=1 AD\else BC\fi
}
```

\texosqueryfmtpatGG Default era for the GG designator. Just make it the same as the G designator.

```
\def\texosqueryfmtpatGG{\texosqueryfmtpatG}
```

\texosqueryfmtpatGGG Default era for the GGG designator. Just make it the same as the G designator.

```
\def\texosqueryfmtpatGGG{\texosqueryfmtpatG}
```

\texosqueryfmtpatGGGG Default era for the GGGG designator. Just make it the same as the G designator.

```
\def\texosqueryfmtpatGGGG{\texosqueryfmtpatG}
```

3.1.5 Applying Numeric Patterns

\texosqueryfmtnumber \texosqueryfmtnumber{⟨*pattern*⟩}{⟨*int*⟩}{⟨*frac*⟩}{⟨*mantissa*⟩}

General purpose low-level number formatting command. The first argument ⟨*pattern*⟩ is the pattern. The other arguments are unformatted integers and must be present and not exceed 10 digits each. The ⟨*frac*⟩ part must not start with a sign. The minus sign should go at the start of ⟨*int*⟩ for negative numbers. The plus sign is optional for positive ⟨*int*⟩ or ⟨*mantissa*⟩ and not permitted in ⟨*frac*⟩. The arguments may each be the actual numerical value or be a single control sequence whose replacement text is the value. Avoid anything more complicated than that.

This package doesn't provide a higher level command that can split a number into integer, fractional and mantissa parts.

```
\def\texosqueryfmtnumber#1#2#3#4{%
  \begingroup
   \let\texosquerypatstr\texosquerypatfmtstr
   \let\texosquerypatquote\texosquerypatfmtquote
   \let\texosquerypatplusminus\texosquerypatfmt@plusminus
   \let\texosquerypatnum\texosquerypatfmt@num
   \let\texosquerypatsinum\texosquerypatfmt@sinum
   \let\texosquerypatdec\texosquerypatfmt@dec
   \let\texosquerypatprefixcurrency\texosquery@patfmt@prefixcurrency
   \let\texosquerypatprefixicurrency\texosquery@patfmt@prefixicurrency
   \let\texosquerypatsuffixcurrency\texosquery@patfmt@suffixcurrency
   \let\texosquerypatsuffixicurrency\texosquery@patfmt@suffixicurrency
```

```
\let\texosquerypatdigit\texosquerypatfmt@digit
\let\texosquerypatdigitnozero\texosquerypatfmt@digitnozero
\let\texosquerypatgroupsep\texosquerypatfmt@groupsep
\let\texosquerypatprefixpercent\texosquery@patfmt@prefixpercent
\let\texosquerypatsuffixpercent\texosquery@patfmt@suffixpercent
\let\texosquerypatprefixpermill\texosquery@patfmt@prefixpermill
\let\texosquerypatsuffixpermill\texosquery@patfmt@suffixpermill
\let\texosquerypatminus\@texosquerypat@numfmt@sign
\let\texosquerypatfmt@decsep\texosquerypatfmtdecsep
```

Allow for negative zero in the ⟨*int*⟩ part. To avoid overflow, first check for 0 and then append 1 to the number to catch `-0`.

```
\edef\@texosquery@sgn{%
  \ifnum#2=0
     \expandafter\ifnum#21<0 -\else+\fi
  \else
     \ifnum#2<0 -\else+\fi
  \fi
 }%
```

Allow for arguments passed as control sequences that expand to a number.

```
\edef\@texosquery@int{\expandafter\@texosquery@paddigits
   \expandafter{\number#2}}%
\let\@texosquery@si@int\@texosquery@int
```

Can't use `\number` here as we'll lose any leading zeros.

```
\edef\@texosquery@frac{\expandafter\@texosquery@paddigits@trailing
  \expandafter{#3}}%
\let\@texosquery@si@frac\@texosquery@frac
\edef\@texosquery@mantissa{\expandafter\@texosquery@paddigits
  \expandafter{\number#4}}%
```

Is the mantissa non-zero?

```
\ifnum#4=0\relax
\else
  \expandafter\ifx\@texosquery@sgn-%
    \edef\@texosquery@int{\expandafter
       \@texosquery@paddigits@pos\expandafter{\number-#2}}%
  \else
    \edef\@texosquery@int{\@texosquery@paddigits@pos{#2}}%
  \fi
```

Shift.

```
  \ifnum#4<0
    \expandafter\@texosquery@neg@shift\expandafter{\number-#4}%
  \else
    \@texosquery@pos@shift{#4}%
  \fi
  \expandafter\ifx\@texosquery@sgn-%
    \edef\@texosquery@int{\expandafter\@texosquery@paddigits
      \expandafter{\number-\@texosquery@int}}%
  \else
```

```
      \edef\@texosquery@int{\expandafter\@texosquery@paddigits
        \expandafter{\number\@texosquery@int}}%
    \fi
    \edef\@texosquery@frac{\@texosquery@paddigits@trailing{\@texosquery@frac}}%
  \fi
  \edef\@texosquery@current{\expandafter\@texosquery@gobble\@texosquery@int}%
  \let\@texosquery@zerodigit\@texosquery@zerodigit@leading
  \expandafter\ifx\@texosquery@sgn-%
    \let\@texosquery@currentsign\texosquerypatfmtminus
  \else
    \let\@texosquery@currentsign\texosquerypatfmtplus
  \fi
  \@texosquery@digitindex=0\relax
  \let\@texosquery@patfmt@dosep\empty
  \@texosquery@digitfoundfalse
  #1%
 \endgroup
}
```

`\@texosquery@digitindex`

```
\newcount\@texosquery@digitindex
```

`\if@texosquery@digitfound`

```
\newif\if@texosquery@digitfound
```

Macros to shift the decimal place.

`\@texosquery@pos@shift`

```
\def\@texosquery@pos@shift#1{%
  \ifcase#1
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lastnineoften\@texosquery@int
      \expandafter\@texosquery@firstoften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lastnineoften\@texosquery@frac
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lasteightoften\@texosquery@int
      \expandafter\@texosquery@firsttwooften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lasteightoften\@texosquery@frac
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lastsevenoften\@texosquery@int
      \expandafter\@texosquery@firstthreeoften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lastsevenoften\@texosquery@frac
```

```
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lastsixoften\@texosquery@int
      \expandafter\@texosquery@firstfouroften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lastsixoften\@texosquery@frac
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lastfiveoften\@texosquery@int
      \expandafter\@texosquery@firstfiveoften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lastfiveoften\@texosquery@frac
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lastfouroften\@texosquery@int
      \expandafter\@texosquery@firstsixoften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lastfouroften\@texosquery@frac
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lastthreeoften\@texosquery@int
      \expandafter\@texosquery@firstsevenoften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lastthreeoften\@texosquery@frac
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@lasttwooften\@texosquery@int
      \expandafter\@texosquery@firsteightoften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@lasttwooften\@texosquery@frac
    }%
  \or
    \edef\@texosquery@int{%
      \expandafter\@texosquery@tenthoften\@texosquery@int
      \expandafter\@texosquery@firstnineoften\@texosquery@frac}%
    \edef\@texosquery@frac{%
       \expandafter\@texosquery@tenthoften\@texosquery@frac
    }%
  \or
    \let\@texosquery@int\@texosquery@frac
    \edef\@texosquery@frac{0}%
```

Anything larger will require scientific notation. Hopefully the pattern supports this.

```
  \fi
}
```

\@texosquery@neg@shift

```
\def\@texosquery@neg@shift#1{%
  \ifcase#1
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lastoneoften\@texosquery@int
      \expandafter\@texosquery@firstnineoften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firstnineoften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lasttwooften\@texosquery@int
      \expandafter\@texosquery@firsteightoften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firsteightoften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lastthreeoften\@texosquery@int
      \expandafter\@texosquery@firstsevenoften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firstsevenoften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lastfouroften\@texosquery@int
      \expandafter\@texosquery@firstsixoften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firstsixoften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lastfiveoften\@texosquery@int
      \expandafter\@texosquery@firstfiveoften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firstfiveoften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lastsixoften\@texosquery@int
      \expandafter\@texosquery@firstfouroften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
```

```
      \expandafter\@texosquery@firstfouroften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lastsevenoften\@texosquery@int
      \expandafter\@texosquery@firstthreeoften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firstthreeoften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lasteightoften\@texosquery@int
      \expandafter\@texosquery@firsttwooften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firsttwooften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{%
      \expandafter\@texosquery@lastnineoften\@texosquery@int
      \expandafter\@texosquery@firstoften\@texosquery@frac
    }%
    \edef\@texosquery@int{%
      \expandafter\@texosquery@firstoften\@texosquery@int
    }%
  \or
    \edef\@texosquery@frac{\@texosquery@int\@texosquery@frac}%
    \edef\@texosquery@int{0}%
%   \end{macrocode}
%Anything beyond this will require scientific notation. Hopefully the
%pattern supports it.
%   \begin{macrocode}
  \fi
}
```

`\@texosquerypat@numfmt@sign`

```
\def\@texosquerypat@numfmt@sign{%
  \@texosquery@currentsign
  \let\@texosquery@currentsign\empty
}
```

`\texosquerypatfmtstr`

```
\def\texosquerypatfmtstr#1{#1}
```

`\texosquerypatfmtquote`

```
\def\texosquerypatfmtquote{'}
```

`\texosquerypatfmt@plusminus`

```
\def\texosquerypatfmt@plusminus#1#2{%
 \edef\@texosquery@current{\expandafter\@texosquery@gobble\@texosquery@int}%
 \@texosquery@digitindex=0\relax
 \let\@texosquery@patfmt@dosep\empty
 \@texosquery@digitfoundfalse
 \expandafter\ifx\@texosquery@sgn-%
  #2%
  \ifnum\@texosquery@digitindex=10
  \else
   \@texosquery@invalidpattern{#2}%
  \fi
 \else
  #1%
  \ifnum\@texosquery@digitindex=10
  \else
   \@texosquery@invalidpattern{#1}%
  \fi
 \fi
}
```

`\texosquerypatfmt@num`

```
\def\texosquerypatfmt@num#1{#1}
```

`\texosquerypatfmtexp` Exponent symbol. Change as appropriate.

```
\def\texosquerypatfmtexp{E}
```

`\texosquerypatfmt@sinum`

```
\def\texosquerypatfmt@sinum#1#2{%
 \let\@texosquery@int\@texosquery@si@int
 \let\@texosquery@frac\@texosquery@si@frac
 \let\@texosquery@current\@texosquery@int
 #1%
 \texosquerypatfmtexp
 {\let\@texosquery@zerodigit\@texosquery@zerodigit@leading
 \ifnum\@texosquery@mantissa<0\relax
   \let\@texosquery@currentsign\texosquerypatfmtminus
 \else
   \let\@texosquery@currentsign\texosquerypatfmtplus
 \fi
 \edef\@texosquery@current{\expandafter\@texosquery@gobble\@texosquery@mantissa}%
 \@texosquery@digitindex=0\relax
 \let\@texosquery@patfmt@dosep\empty
 \@texosquery@digitfoundfalse
 #2}}
```

`\texosquerypatfmtdecsep` Decimal separator. Change as appropriate.

```
\def\texosquerypatfmtdecsep{.}
```

`\texosquerypatfmtcurdecsep` Currency decimal separator. Change as appropriate.

```
\def\texosquerypatfmtcurdecsep{.}
```

\texosquerypatfmt@dec

```
\def\texosquerypatfmt@dec#1#2{%
 \edef\@texosquery@current{\expandafter\@texosquery@gobble\@texosquery@int}%
 \@texosquery@digitindex=0\relax
 \let\@texosquery@patfmt@dosep\empty
 \@texosquery@digitfoundfalse
 \let\@texosquery@zerodigit\@texosquery@zerodigit@leading
 #1%
 \ifnum\@texosquery@digitindex=10
 \else
  \@texosquery@invalidpattern{#1}%
 \fi
```

We can't display the decimal separator just yet as the pattern may not show any digits for the fractional part. So only do the separator just before the first digit.

```
 \let\@texosquery@patfmt@dosep\texosquerypatfmt@decsep
 \let\@texosquery@current\@texosquery@frac
 \@texosquery@digitindex=0\relax
 \@texosquery@digitfoundfalse
 \let\@texosquery@zerodigit\@texosquery@zerodigit@trailing
 \let\@texosquery@currentsign\empty
 #2%
 \ifnum\@texosquery@digitindex=10
 \else
  \@texosquery@invalidpattern{#2}%
 \fi
}
```

\texosquerypatfmtint

```
\def\texosquerypatfmtint#1{%
 \edef\@texosquery@current{\expandafter\@texosquery@gobble\@texosquery@int}%
 \@texosquery@digitindex=0\relax
 \@texosquery@digitfoundfalse
 \let\@texosquery@zerodigit\@texosquery@zerodigit@leading
 #1%
 \ifnum\@texosquery@digitindex=10
 \else
  \@texosquery@invalidpattern{#1}%
 \fi
}
```

\@texosquery@setpatdisplay

```
\def\@texosquery@setpatdisplay{%
   \def\texosquerypatstr##1{'##1'}%
   \def\texosquerypatquote{''}%
   \def\texosquerypatplusminus##1##2{##1;##2}%
   \def\texosquerypatnum##1{##1}%
   \def\texosquerypatsinum##1##2{##1E##2}%
   \def\texosquerypatdec##1##2{##1.##2}%
   \def\texosquerypatprefixcurrency##1##2{##2␣##1}%
```

```
    \def\texosquerypatprefixicurrency##1##2{##2␣␣##1}%
    \def\texosquerypatsuffixcurrency##1##2{##1␣##2}%
    \def\texosquerypatsuffixicurrency##1##2{##1␣␣##2}%
    \def\texosquerypatdigit{0}%
    \def\texosquerypatdigitnozero{\#}%
    \def\texosquerypatminus{-}%
    \def\texosquerypatgroupsep{,}%
    \def\texosquerypatprefixpercent##1##2{##2\%##1}%
    \def\texosquerypatsuffixpercent##1##2{##1\%##2}%
    \def\texosquerypatprefixpermill##1##2{##2‰##1}%
    \def\texosquerypatsuffixpermill##1##2{##1‰##2}%
    \def\texosquerypatfmt@decsep{.}%
}
```

`\@texosquery@invalidpattern`

```
\def\@texosquery@invalidpattern#1{%
  \begingroup
   \@texosquery@setpatdisplay
   \@texosquery@err{10 digit specifiers expected in
    numeric pattern #1. Found \number\@texosquery@digitindex}%
    {Each integer element of a numeric pattern must have exactly
     10 digit specifiers (0 or \#)}%
  \endgroup
}
```

`\texosquerypatfmtcurrencysign` Currency symbol. Redefine as appropriate.

```
\def\texosquerypatfmtcurrencysign{\$}
```

`\texosquerypatfmticurrencysign` International currency symbol. There's no generic fallback that's independent of the input encoding, so this uses a UTF-8 character on the assumption that if `\textcurrency` isn't available (for example, through textcomp, then the user may be using XeTeX or LuaTeX). If this isn't the case, and there's no UTF-8 support, then this command will need to be redefined as appropriate.

```
\ifx\textcurrency\undefined
  \def\texosquerypatfmticurrencysign{¤}
\else
  \def\texosquerypatfmticurrencysign{\textcurrency}
\fi
```

`exosquery@patfmt@prefixcurrency`

```
\def\texosquery@patfmt@prefixcurrency#1#2{%
 \let\texosquerypatfmt@decsep\texosquerypatfmtcurdecsep
 #2\texosquerypatfmtcurrencysign#1%
}
```

`xosquery@patfmt@prefixicurrency`

```
\def\texosquery@patfmt@prefixicurrency#1#2{%
 \let\texosquerypatfmt@decsep\texosquerypatfmtcurdecsep
 #2\texosquerypatfmticurrencysign#1%
}
```

exosquery@patfmt@suffixcurrency

```
\def\texosquery@patfmt@suffixcurrency#1#2{%
 \let\texosquerypatfmt@decsep\texosquerypatfmtcurdecsep
 #1\texosquerypatfmtcurrencysign#2%
}
```

xosquery@patfmt@suffixicurrency

```
\def\texosquery@patfmt@suffixicurrency#1#2{%
 \let\texosquerypatfmt@decsep\texosquerypatfmtcurdecsep
 #1\texosquerypatfmticurrencysign#2%
}
```

\texosquerypatfmt@digit

```
\def\texosquerypatfmt@digit{%
 \advance\@texosquery@digitindex by 1\relax
 \if@texosquery@digitfound
 \else
   \ifx\@texosquery@currentsign\texosquerypatfmtminus
    \texosquerypatfmtminus
    \let\@texosquery@currentsign\empty
   \else
    \@texosquery@patfmt@dosep
    \let\@texosquery@patfmt@dosep\empty
   \fi
 \fi
 \@texosquery@digitfoundtrue
 \ifcase\@texosquery@digitindex
 \or
   \expandafter\@texosquery@firstoften\@texosquery@current
 \or
   \expandafter\@texosquery@secondoften\@texosquery@current
 \or
   \expandafter\@texosquery@thirdoften\@texosquery@current
 \or
   \expandafter\@texosquery@fourthoften\@texosquery@current
 \or
   \expandafter\@texosquery@fifthoften\@texosquery@current
 \or
   \expandafter\@texosquery@sixthoften\@texosquery@current
 \or
   \expandafter\@texosquery@seventhoften\@texosquery@current
 \or
   \expandafter\@texosquery@eighthoften\@texosquery@current
 \or
   \expandafter\@texosquery@ninthoften\@texosquery@current
 \or
   \expandafter\@texosquery@tenthoften\@texosquery@current
 \fi
}
```

\texosquerypatfmt@digitnozero

```
\def\texosquerypatfmt@digitnozero{%
 \advance\@texosquery@digitindex by 1\relax
 \edef\@texosquery@digit{%
   \ifcase\@texosquery@digitindex
    0%
   \or
     \expandafter\@texosquery@firstoften\@texosquery@current
   \or
     \expandafter\@texosquery@secondoften\@texosquery@current
   \or
     \expandafter\@texosquery@thirdoften\@texosquery@current
   \or
     \expandafter\@texosquery@fourthoften\@texosquery@current
   \or
     \expandafter\@texosquery@fifthoften\@texosquery@current
   \or
     \expandafter\@texosquery@sixthoften\@texosquery@current
   \or
     \expandafter\@texosquery@seventhoften\@texosquery@current
   \or
     \expandafter\@texosquery@eighthoften\@texosquery@current
   \or
     \expandafter\@texosquery@ninthoften\@texosquery@current
   \or
     \expandafter\@texosquery@tenthoften\@texosquery@current
   \else
   0%
 \fi
 }%
 \ifnum\@texosquery@digit=0\relax
   \@texosquery@zerodigit
 \else
   \if@texosquery@digitfound
   \else
     \ifx\@texosquery@currentsign\texosquerypatfmtminus
      \texosquerypatfmtminus
      \let\@texosquery@currentsign\empty
     \else
       \@texosquery@patfmt@dosep
       \let\@texosquery@patfmt@dosep\empty
     \fi
   \fi
   \@texosquery@digitfoundtrue
   \@texosquery@digit
 \fi
}
```

\@texosquery@zerodigit@leading

```
\def\@texosquery@zerodigit@leading{%
 \edef\@texosquery@digit{%
   \ifcase\@texosquery@digitindex
    0%
   \or
     \expandafter\@texosquery@firstoften\@texosquery@current
   \or
     \expandafter\@texosquery@firsttwooften\@texosquery@current
   \or
     \expandafter\@texosquery@firstthreeoften\@texosquery@current
   \or
     \expandafter\@texosquery@firstfouroften\@texosquery@current
   \or
     \expandafter\@texosquery@firstfiveoften\@texosquery@current
   \or
     \expandafter\@texosquery@firstsixoften\@texosquery@current
   \or
     \expandafter\@texosquery@firstsevenoften\@texosquery@current
   \or
     \expandafter\@texosquery@firsteightoften\@texosquery@current
   \or
     \expandafter\@texosquery@firstnineoften\@texosquery@current
   \or
     \@texosquery@current
   \else
    0%
   \fi
 }%
 \ifnum\@texosquery@digit>0\relax
  \if@texosquery@digitfound
  \else
    \ifx\texosquerypatminus\texosquerypatfmtminus
     \texosquerypatfmtminus
    \else
      \@texosquery@patfmt@dosep
      \let\@texosquery@patfmt@dosep\empty
    \fi
  \fi
  \@texosquery@digitfoundtrue
  0%
 \fi
}
```

\@texosquery@zerodigit@trailing

```
\def\@texosquery@zerodigit@trailing{%
 \edef\@texosquery@digit{%
   \ifcase\@texosquery@digitindex
    0%
   \or
     \@texosquery@current
```

```
  \or
    \expandafter\@texosquery@lastnineoften\@texosquery@current
  \or
    \expandafter\@texosquery@lasteightoften\@texosquery@current
  \or
    \expandafter\@texosquery@lastsevenoften\@texosquery@current
  \or
    \expandafter\@texosquery@lastsixoften\@texosquery@current
  \or
    \expandafter\@texosquery@lastfiveoften\@texosquery@current
  \or
    \expandafter\@texosquery@lastfouroften\@texosquery@current
  \or
    \expandafter\@texosquery@lastthreeoften\@texosquery@current
  \or
    \expandafter\@texosquery@lasttwooften\@texosquery@current
  \or
    \expandafter\@texosquery@tenthoften\@texosquery@current
  \else
   0%
  \fi
 }%
 \ifnum\@texosquery@digit>0\relax
  \if@texosquery@digitfound
  \else
    \ifx\texosquerypatminus\texosquerypatfmtminus
     \texosquerypatfmtminus
    \else
      \@texosquery@patfmt@dosep
      \let\@texosquery@patfmt@dosep\empty
    \fi
  \fi
  \@texosquery@digitfoundtrue
  0%
 \fi
}
```

`\texosquerypatfmtminus` Formatted minus sign. Redefined as appropriate.

```
\def\texosquerypatfmtminus{\ifmmode-\else$-$\fi}
```

`\texosquerypatfmtplus` Formatted plus sign. Redefined as appropriate.

```
\def\texosquerypatfmtplus{\ifmmode+\else$+$\fi}
```

`\texosquerypatfmtgroupsep`

```
\def\texosquerypatfmtgroupsep{,}
```

`\texosquerypatfmt@groupsep`

```
\def\texosquerypatfmt@groupsep{%
  \if@texosquery@digitfound\texosquerypatfmtgroupsep\fi}
```

\texosquerypatfmtpercentsign Percent sign used in number format.

```
\def\texosquerypatfmtpercentsign{\%}
```

\texosquerypatfmtpermillsign Per-mill sign used in number format. Redefine as appropriate.

```
\def\texosquerypatfmtpermillsign{‰}
```

\@texosquery@adjust@per

```
\def\@texosquery@adjust@per#1{%
 \@texosquery@pos@shift{#1}%
 \edef\@texosquery@int{\expandafter\@texosquery@paddigits
     \expandafter{\number\@texosquery@int}}%
 \edef\@texosquery@frac{\@texosquery@paddigits@trailing{\@texosquery@frac}}%
 \edef\@texosquery@current{\expandafter\@texosquery@gobble\@texosquery@int}%
}
```

\texosquery@patfmt@prefixpercent

```
\def\texosquery@patfmt@prefixpercent#1#2{%
 \@texosquery@adjust@per{2}%
  #2\texosquerypatfmtpercentsign#1%
}
```

\texosquery@patfmt@suffixpercent

```
\def\texosquery@patfmt@suffixpercent#1#2{%
 \@texosquery@adjust@per{2}%
 #1\texosquerypatfmtpercentsign#2%
}
```

\texosquery@patfmt@prefixpermill

```
\def\texosquery@patfmt@prefixpermill#1#2{%
 \@texosquery@adjust@per{3}%
 #2\texosquerypatfmtpermillsign#1%
}
```

\texosquery@patfmt@suffixpermill

```
\def\texosquery@patfmt@suffixpermill#1#2{%
 \@texosquery@adjust@per{3}%
 #1\texosquerypatfmtpermillsign#2%
}
```

All done. Restore the category code of `@`:

```
\@texosquery@restore@at
```

3.2 LaTeX Code

This is just a simple wrapper for `texosquery.tex` so that it can be loaded using LaTeX's standard \usepackage method. Identify package:

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{texosquery}[2017/03/31 v1.3 (NLCT)]
```

Load `texosquery.tex`:

```
\input{texosquery}
```

3.3 Configuration File (`texosquery.cfg`)

The configuration file. This will need to be edited as appropriate to the system.

```
%If this configuration file is added to TeX's path, it can
%be used to set up the texosquery defaults for the installation.

%Default application (must be installed on the operating system's
%path). Change as appropriate. Available options:
% * texosquery-jre8 (at least Java 8 required)
% * texosquery (at least Java 7 required)
% * texosquery-jre5 (at least Java 5 required)
%(bash users need to check that the .sh extension has been removed
%from the bash scripts, or add the extension to the invoker name.)
\def\TeXOSInvokerName{texosquery}

% If the invoker name given above is on the restricted list,
% allow it to be run in restricted mode:
%\TeXOSQueryAllowRestricted
```

3.4 Bash Scripts

These are the bash scripts for Unix-like systems. The first line

```
#!/bin/sh
```

is added when the files are extracted by `texosquery.ins` (since `\nopreamble` automatically inserts a blank line at the start of the file).

3.4.1 `texosquery.sh`

```
jarpath=‘kpsewhich --progname=texosquery --format=texmfscripts texosquery.jar‘
java -jar "$jarpath" "$@"
```

3.4.2 `texosquery-jre8.sh`

```
jarpath=‘kpsewhich --progname=texosquery --format=texmfscripts texosquery-jre8.jar‘
java -Djava.locale.providers=CLDR,JRE -jar "$jarpath" "$@"
```

3.4.3 `texosquery-jre5.sh`

```
jarpath=‘kpsewhich --progname=texosquery --format=texmfscripts texosquery-jre5.jar‘
java -jar "$jarpath" "$@"
```

3.5 Windows Batch Scripts

These are the batch scripts for Windows. TEX on Windows doesn't allow the creation of `.bat` files, so `.ins` file creates these with the extension `.batch` which will need to be changed to `.bat` after extraction.

3.5.1 texosquery.bat

```
@ECHO OFF
FOR /F "tokens=*" %%I IN ('kpsewhich --progname=texosquery --format=texmfscripts texosquery.jar
java -jar "%JARPATH%" %*
```

3.5.2 texosquery-jre8.bat

```
@ECHO OFF
FOR /F "tokens=*" %%I IN ('kpsewhich --progname=texosquery --format=texmfscripts texosquery-jre8
java -Djava.locale.providers=CLDR,JRE -jar "%JARPATH%" %*
```

3.5.3 texosquery-jre5.bat

```
@ECHO OFF
FOR /F "tokens=*" %%I IN ('kpsewhich --progname=texosquery --format=texmfscripts texosquery-jre5
java -jar "%JARPATH%" %*
```

Abbreviations

ASCII American Standard Code for Information Interchange

BCP Best Common Practice

CLDR Unicode Consortium's Common Locale Data Repository

CTAN Comprehensive TEX Archive Network

IETF Internet Engineering Task Force

ISO International Organization for Standardization

JRE Java Runtime Environment

OS operating system

POSIX Portable Operating System Interface

UTF Unicode Transformation Format

Change History

Index