

The pkginfograb Package

Version 1.1

Alceu Frigeri*

November 2025

Abstract

This package is aimed at package writers and offers a way to collect/document L^AT_EX package's info (name, version, description, etc.) in a systemized way, including a mechanism to check package's version. Just a few functions are defined, to document/set package's info, retrieve them and to verify package's version (for instance, to verify if a loaded package is newer than a given reference).

1 Expl3 Commands

`\pkginfograb_set:nn` `\pkginfograb_set:nn` {<pack-name>} {<keyval-list>}

This will create a property list associated with <pack-name>. <keyval-list> might contain any set of keys, though, the functions below expect at least <version> or <date> (for version checking) and <name>, <version>, <date> and <description> (for `\pkginfograb_description:n`).

Note: An error will be raised if calling it twice for the same <pack-name>.

For Example:

```
\pkginfograb_set:nn {pkginfograb}
{
  name       = {pkginfograb} ,
  prefix     = {pkginfograb} ,
  date       = {2025/11/01},
  version    = {1.1} ,
  description = {Collecting~ package's~ info~ in~ a~ regular~ way}
}
```

`\pkginfograb_req_version:nnn` `\pkginfograb_req_version:nnn` {<your-pack>} {<pack-name>} {<min-version>}

This will verify if <pack-name>'s <version> (as stored with `\pkginfograb_set:nn`) is at least <min-version>. It expects <version> in one of three formats `[v]digits[letters]`, `[v]digits.digits[letters]` or `[v]digits.digits.digits[letters]` (the `[v]`, if present, is ignored).

Note: An error will be raised if <pack-name>'s info isn't defined, incorrect version format or <min-version> isn't satisfied, in which case the error will note that **your-pack** needs version <min-version> of <pack-name>.

`\pkginfograb_req_date:nnn` `\pkginfograb_req_date:nnn` {<your-pack>} {<pack-name>} {<min-date>}

This will verify if <pack-name>'s <date> (as stored with `\pkginfograb_set:nn`) is at least <min-date>. It expects <date> in one of three formats `YYYY/MM/DD`, `YYYY-MM-DD` or `YYYY.MM.DD`.

Note: An error will be raised if <pack-name>'s info isn't defined, incorrect version format or <min-version> isn't satisfied, in which case the error will note that **your-pack** needs version <min-date> of <pack-name>.

`\pkginfograb_set_aliases:` `\pkginfograb_set_aliases:`

This will set L^AT_EX 2_ε aliases for the `expl3` in this package. Note that none of the commands in 2 are defined by default, except `\PkgInfoSetAliases` which is an alias for this command.

*<https://github.com/alceu-frigeri/pkginfograb>

`\pkginfograb_get:nn` ★ `\pkginfograb_get:nn` {<pack-name>} {<key>}

This will retrieve <key>'s value. If <pack-name> or <key> doesn't exist, this will expand to nothing.

`\pkginfograb_get:nnN` `\pkginfograb_get:nnN` {<pack-name>} {<key>} {<t1-var>}

This will store <key>'s value at <t1-var>. If <pack-name> or <key> doesn't exist, <t1-var> will be cleared.

`\pkginfograb_description:n` `\pkginfograb_description:n` {<pack-name>}

This will typeset a small paragraph (for data validation) with following <pack-name>'s info: <name>, <version>, <date> and <description>

For example:

```
\ExplSyntaxOn
\pkginfograb_description:n{pkginfograb}
\ExplSyntaxOff
```

Package **pkginfograb** Version: 1.1 - 2025/11/01
Collecting package's info in a regular way

`\pkginfograb_if_set_p:n` `\pkginfograb_if_set_p:n` {<pack-name>}

`\pkginfograb_if_set:nTF` ★ `\pkginfograb_if_set:nTF` {<pack-name>} {<if-true>} {<if-false>}

new: 2025/11/01

This will test if the given <pack-name> was set with `\pkginfograb_set:nn` and <if-true> or <if-false> will be properly executed.

`\pkginfograb_map_inline:n` `\pkginfograb_map_inline:n` {<inline-code>}

new: 2025/11/01

<inline-code> will receive (as <#1>) the name of each package set with `\pkginfograb_set:nn` in the order they were set.

```
\ExplSyntaxOn
\pkginfograb_map_inline:n
{
  \pkginfograb_description:n{#1}
}
\ExplSyntaxOff
```

Package **pkginfograb** Version: 1.1 - 2025/11/01
Collecting package's info in a regular way
Package **codedescribe** Version: 1.16b - 2025/10/30
L^AT_EX Code Description/Documentation
Package **xpeekahead** Version: 1.3 - 2025/10/23
A simplistic peek ahead set up
Package **codelisting** Version: 1.16b - 2025/10/30
L^AT_EX Code Listing

2 LaTeX2e Commands' Aliases

All commands below are aliases to their `expl3` counterparts. Aside from `\PkgInfoSetAliases` all other L^AT_EX2e aliases aren't defined by default. Call either `\pkginfograb_set_aliases:` or `\PkgInfoSetAliases` before using them.

`\PkgInfoSet` `PkgInfoSet` {<pack-name>} {<keyval-list>}

This will create a property list associated with <pack-name>. <keyval-list> might contain any set of keys, though, the functions below expect at least <version> (for version checking) and <name>, <version>, <date> and <description> (for `\PkgInfoDescription`).

Note: An error will be raised if calling it twice for the same <pack-name>.

For Example:

```
\PkgInfoSet {pkginfograb}
{
  name      = {pkginfograb} ,
  prefix    = {pkginfograb} ,
  date      = {2025/11/01},
  version   = {1.1} ,
  description = {Collecting~ package's~ info~ in~ a~ regular~ way}
}
```

`\PkgInfoReqVersion` `\PkgInfoReqVersion {<your-pack>} {<pack-name>} {<min-version>}`

This will verify if `<pack-name>`'s `<version>` (as stored with `\PkgInfoSet`) is at least `<min-version>`. It expects `<version>` in one of three formats `[v]digits[letters]`, `[v]digits.digits[letters]` or `[v]digits.digits.digits[letters]` (the `[v]`, if present, is ignored).

Note: An error will be raised if `<pack-name>`'s info isn't defined, incorrect version format or `<min-version>` isn't satisfied, in which case the error will note that **your-pack** needs version `<min-version>` of `<pack-name>`.

`\PkgInfoReqDate` `\PkgInfoReqDate {<your-pack>} {<pack-name>} {<min-date>}`

This will verify if `<pack-name>`'s `<date>` (as stored with `\PkgInfoSet`) is at least `<min-date>`. It expects `<date>` in one of three formats `YYYY/MM/DD`, `YYYY-MM-DD` or `YYYY.MM.DD`.

Note: An error will be raised if `<pack-name>`'s info isn't defined, incorrect version format or `<min-version>` isn't satisfied, in which case the error will note that **your-pack** needs version `<min-date>` of `<pack-name>`.

`\PkgInfoSetAliases` `\PkgInfoSetAliases`

This will set L^AT_EX 2_ε aliases for the `expl3` in this package. Note that none of the commands in 2 are defined by default, except this.

`\PkgInfo` ★ `\PkgInfo {<pack-name>} {<key>}`

This will retrieve `<key>`'s value. If `<pack-name>` or `<key>` doesn't exist, this will expand to nothing.

`\PkgInfoGet` `\PkgInfoGet {<pack-name>} {<key>} {<macro>}`

This will store `<key>`'s value in `<macro>`. If `<pack-name>` or `<key>` doesn't exist, `<macro>` will be cleared.

`\PkgInfoDescription` `\PkgInfoDescription {<pack-name>}`

This will typeset a small paragraph (for data validation) with following `<pack-name>`'s info: `<name>`, `<version>`, `<date>` and `<description>`

For example:

```
\PkgInfoDescription{pkginfograb} Package pkginfograb Version: 1.1 - 2025/11/01  
Collecting package's info in a regular way
```

`\PkgInfoIfSet` `\PkgInfoIfSet {<pack-name>} {<if-true>} {<if-false>}`

new: 2025/11/01 This will test if the given `<pack-name>` was set with `\PkgInfoSet` and `<if-true>` or `<if-false>` will be properly executed.

`\PkgInfoMapOver` `\PkgInfoMapOver {<inline-code>}`

new: 2025/11/01 `<inline-code>` will receive (as `<#1>`) the name of each package set with `\PkgInfoSet` in the order they were set.

```
\PkgInfoMapOver Package pkginfograb Version: 1.1 - 2025/11/01  
{ Collecting package's info in a regular way  
  \PkgInfoDescription{#1} Package codedescribe Version: 1.16b - 2025/10/30  
  } LATEXCode Description/Documentation  
Package xpeekahead Version: 1.3 - 2025/10/23  
  A simplistic peek ahead set up  
Package codelisting Version: 1.16b - 2025/10/30  
  LATEXCode Listing
```
