The pgfkeysearch Package
A Search Extension for pgfkeys
Version 1.4a

Alceu Frigeri[*]

October 2025

Abstract

The command \pgfkeysvalueof, unlike \pgfkeys command, doesn't use the *.unknown* handler or offers the option to search for a key in other paths, and raises an error if the key isn't defined in the given path.

The following commands will recursively search for a key in a collection of paths.

Contents

1 Package Options

The default search behaviour assumes that all keys defined by a package or document are under a uniquely defined path, meaning, no root keys. For instance, given the path /A/B/C/D, the following commands will look, first, at /A/B/C/D/⟨key⟩, then /A/B/C/⟨key⟩, and so on, until /A/⟨key⟩, stopping at the first hit. This can be changed with the following package option:

root search If set, the *path root* will also be included in the search, meaning it will look if /⟨key⟩, as last resort, is defined.

> ***Note:*** If set, the root key (/⟨key⟩) will be look at for every path in the path list. For instance `\pgfkeysearch {/A/B/C,/X/Y,/Z/T}{key}`, /⟨key⟩ will be tried up to three times.

\pgfkeysearchsettings

new: 2025/05/27

`\pgfkeysearchsettings {⟨options⟩}`

To change the search behaviour, middle document, including or not the path root. ⟨options⟩ are any valid package option (for now just *root search*).

2 User Document Commands

Those commands are meant to be used at Document level. For packages, one is advised to use the ones defined at 3.

[*] https://github.com/alceu-frigeri/pgfkeysearch

\pgfkeysearchvalueof
\pgfkeysearch

updated: 2024/01/11

```
\pgfkeysearchvalueof {⟨path-list⟩}{⟨key⟩}{⟨macro⟩}
\pgfkeysearch {⟨path-list⟩}{⟨key⟩}{⟨macro⟩}
```

⟨path-list⟩ is a comma separated list (clist) of paths (can be a single one). ⟨key⟩ is the desired key, and ⟨macro⟩ is the macro/command that will receive (store) the key value (if one is found). ⟨key⟩ will be searched for in the many paths from ⟨path-list⟩ as described in 1. ⟨macro⟩ will be set with the found (if any) value.

Note: \pgfkeysearch and \pgfkeysearchvalueof are aliases to each other.

Note: These commands aren't expandable, though, once retrieved, the returning macro can be used in an expandable context.

Note: If ⟨key⟩ isn't found, ⟨macro⟩ will be empty, no warning or error will be raised.

\pgfkeysearchvalueof*TF*
\pgfkeysearch*TF*

updated: 2024/01/11

```
\pgfkeysearchvalueofTF {⟨path-list⟩}{⟨key⟩}{⟨macro⟩}{⟨if-found⟩}{⟨if-not⟩}
\pgfkeysearchTF {⟨path-list⟩}{⟨key⟩}{⟨macro⟩}{⟨if-found⟩}{⟨if-not⟩}
```

⟨path-list⟩ is a comma separated list (clist) of paths (can be a single one). ⟨key⟩ is the desired key and ⟨macro⟩ is the macro/command that will receive (store) the key value (if one was found). These branch versions will also execute either ⟨if-found⟩ or ⟨if-not⟩.

Note: \pgfkeysearchvalueof*TF* and \pgfkeysearch*TF* are aliases to each other.

Note: These commands aren't expandable, though, once retrieved, the returning macro can be used in an expandable context.

Note: If ⟨key⟩ isn't found, ⟨macro⟩ will be empty, no warning or error will be raised.

2.1 Example

Given the following pgfkeys:

```
\pgfkeys{%
   /tikz/A/.cd,
   keyA/.initial={keyA at /tikz/A},
   keyB/.initial={keyB at /tikz/A},
   %
   B/.cd,
   keyA/.initial={keyA at /tikz/A/B},
   keyC/.initial={keyC at /tikz/A/B},
   %
   C/.cd,
   keyX/.initial={keyX at /tikz/A/B/C}
 }
```

Key values can be retrieved:

```
\pgfkeysearch{/tikz/X,/tikz/A/B/C}{keyA}{\VALkeyA}
\pgfkeysearch{/tikz/X/Y,/tikz/A/B/C}{keyB}{\VALkeyB}
\pgfkeysearch{/tikz/X/Y,/tikz/Y/Y,/tikz/A/B/C}{keyC}{\VALkeyC}
\pgfkeysearch{/tikz/X/Y,/tikz/Y/Y,/tikz/A/B/C}{keyX}{\VALkeyX}
```

and used as:

```
I got for keyA: \textbf{\VALkeyA} \par
I got for keyB: \textbf{\VALkeyB} \par
I got for keyC: \textbf{\VALkeyC} \par
I got for keyX: \textbf{\VALkeyX} \par
```

I got for keyA: **keyA at /tikz/A/B**
I got for keyB: **keyB at /tikz/A**
I got for keyC: **keyC at /tikz/A/B**
I got for keyX: **keyX at /tikz/A/B/C**

3 Expl3 Commands

\pgfkeysearch_settings:n

new: 2025/05/27

\pgfkeysearch_settings:n {⟨options⟩}

To change the search behaviour, middle document, including or not the path root. ⟨options⟩ are any valid package option (for now just *root search*, see 1, notice the space...).

\pgfkeysearch_keysearch:nnN*TF*

updated: 2025/05/26

\pgfkeysearch_keysearch:nnN*TF* {⟨single-path⟩} {⟨key⟩} {⟨tl-var⟩} {⟨if-found⟩} {⟨if-not⟩}

⟨key⟩ is the desired key, and ⟨tl-var⟩ is a token list variable that will receive the key value, if one is found. ⟨key⟩ will be searched for in ⟨single-path⟩ as described in 1.
\pgfkeysearch_keysearch:nnN*TF* is slightly faster than the more generic multi-path version.

Note: If ⟨key⟩ isn't found, no assignment will be made to ⟨tl-var⟩) and no warning or error will be raised.
Note: The old signature \pgfkeysearch_keysearch:nnn*TF* is deprecated, and will raise a warning if used.

\pgfkeysearch_multipath_keysearch:nnN*TF*

updated: 2025/05/26

\pgfkeysearch_multipath_keysearch:nnN*TF* {⟨path-list⟩} {⟨key⟩} {⟨tl-var⟩} {⟨if-found⟩} {⟨if-not⟩}

Given a comma separated ⟨path-list⟩, this will call \pgfkeysearch_keysearch:nnN*TF* for each path in ⟨path-list⟩, until ⟨key⟩ is found.

Note: If ⟨key⟩ isn't found, no assignment will be made to ⟨tl-var⟩) and no warning or error will be raised.
Note: The document level commands (in 2) are just wrappers to this command.
Note: The old signature \pgfkeysearch_multipath_keysearch:nnn*TF* is deprecated, and will raise a warning if used.