

keyTouch 2.1 file documentation

Syntax version 1.1

Author: Marvin Raaijmakers

Date: 16 April 2006

Table of Contents

1. Introduction.....	3
1.1 Formatting styles.....	3
2. File locations.....	4
3. The keyboard file.....	5
3.1 file-info.....	5
3.2 keyboard-info.....	6
3.3 key-list and key element.....	6
4. The keyboard configuration file.....	8
5. The preferences file.....	9
6. The current keyboard file.....	10

1. Introduction

This document discusses the files used by keyTouch.

KeyTouch has for each keyboard two files: one available over the entire system and one in the home directory of the user. The first file is the so called “keyboard file”, which contains the information about the keyboard such as the scancodes of the keys. The second file contains the configuration of the keyboard, which means that it contains for every key, defined in the “keyboard file”, the action to perform when the key is pressed. We call this file the “keyboard configuration file”.

KeyTouch also has file that contains the preferences of keyTouch user. This file is located in the home directory of the user and is called the “preferences file”.

The name of the keyboard that was chosen by the user is stored in the “current keyboard file”.

All these configuration files are well-formed, invalid XML files. They are invalid XML files because they do not contain a document type declaration. For more information about XML, read the XML specification:

<http://www.w3.org/TR/2004/REC-xml-20040204/>

1.1 Formatting styles

This document uses several formatting styles to denote the context of some text. Below the meaning of every style is written in that style:

This style is used for the contents of a file.

This style is used for an XML element type.

This style is used for an XML attribute name.

This style is used for an XML attribute value.

Below you can read an example text that uses all these styles:

An XML document contains:

```
<document>
  <hello lang="en">hello</hello>
  <hello lang="nl">hallo</hello>
  <world lang="en">world</world>
  <world lang="nl">wereld</world>
</document>
```

So the root element of this document is **document** and contains four elements of two different types:

hello and **world**.

The first **hello** element has an attribute named `lang`. The value of this attribute is equal to `en`. For the second **hello** element this value is equal to `nl`.

2. File locations

<i>Location</i>	<i>File(s)</i>
~/.keytouch2/ (directory)	Contains all keyboard configuration files of the user.
~/.keytouch2/preferences.xml	The users preferences file.
{1}/etc/keytouch/current_keyboard.xml	The current keyboard file.
{2}/share/keytouch/keyboards/ (directory)	Contains all keyboard files.

{1} and {2} depend on the installation of keyTouch.

3. The keyboard file

The keyboard file is needed to support a keyboard. Once the file is on the system it should not change. A keyboard file contains:

- Keyboard model and manufacturer name.
- Information about the file: syntax-version, date of last change (optional), author(optional).
- Information for every extra function key.

The syntax-version is the version of the syntax that this document describes.

For every extra function key, the file contains the following information:

- **Keycode**: The name of the code. The list of keycode names may be found in the header file `<linux/input.h>` (without `KEY_` prefix). Note that this keycode is the keycode which is used by the kernel and **not** by X.
- The name of the key.
- The default action to run when the key is pressed.

If the key is not of the type “acpi-hotkey” it will have a scancode (= the code the kernel receives from the keyboard). If the key is an “acpi-hotkey” the key will have an event description.

The keyboard file has one root element, of type **keyboard**, and contains the information described above:

```
<keyboard>
  ...
</keyboard>
```

The three dots are by the following elements:

- **file-info**: Contains the file information.
- **keyboard-info**: Contains information about the keyboard.
- **key-list**: The list of extra function keys.

So it will look like:

```
<keyboard>
  <file-info>
    ...
  </file-info>
  <keyboard-info>
    ...
  </keyboard-info>
  <key-list>
    ...
  </key-list>
</keyboard>
```

The contents of these elements are discussed in the following subsections.

3.1 file-info

Contains the following elements:

- **syntax-version**: The syntax version.
- **last-change**: The date of the last change to the file (optional).
- **author**: The name of the author of the file (optional).

The date of the last change may be written in any form. The format is specified in the `format` argument of the **last-change** element. This format is described in the text below, which is copied from the `strptime()` manual page:

The `strptime()` function processes the input string from left to right. Each of the three possible input elements (whitespace, literal, or for-

mat) are handled one after the other. If the input cannot be matched to the format string the function stops. The remainder of the format and input strings are not processed.

The supported input field descriptors are listed below. In case a text string (such as a weekday or month name) is to be matched, the comparison is case insensitive. In case a number is to be matched, leading zeros are permitted but not required.

- `%%` The `%` character.
- `%b` or `%B` or `%h`
The month name according to the current locale, in abbreviated form or the full name.
- `%d` or `%e`
The day of month (1-31).
- `%D`
Equivalent to `%m/%d/%y`. (This is the American style date, very confusing to non-Americans, especially since `%d/%m/%y` is widely used in Europe. The ISO 8601 standard format is `%Y-%m-%d`.)
- `%m`
The month number (1-12).
- `%x`
The date, using the locale's date format.
- `%y`
The year within century (0-99). When a century is not otherwise specified, values in the range 69-99 refer to years in the twentieth century (1969-1999); values in the range 00-68 refer to years in the twenty-first century (2000-2068).
- `%Y`
The year, including century (for example, 1991).

So the **file-info** element can look like:

```
<file-info>
  <syntax-version>1.2</syntax-version>
  <last-change format="%d-%m-%y">4-9-2005</last-change>
  <author>Author Name</author>
</file-info>
```

3.2 keyboard-info

The **keyboard-info** element only contains one element: **keyboard-name**. This element contains the **manufacturer** and **model** elements. These two elements contain the names of the manufacturer and the model respectively.

So the **keyboard-info** element can look like:

```
<keyboard-info>
  <keyboard-name>
    <manufacturer>The Keyboard Company</manufacturer>
    <model>Keyboard Number One</model>
  </keyboard-name>
</keyboard-info>
```

3.3 key-list and key element

The **key-list** element contains for every extra function key an element called **key**.

A **key** element can have an `key-type` attribute. If the value of this attribute is `acpi-hotkey` the key

is an ACPI hotkey. If the element has no `key-type` attribute, or the attributes value is not `acpi-hotkey`, the key is a normal key which is part of the keyboard

A **key** element contains the following elements:

- **scancode** (for normal keys): Contains the scancode that the kernel receives from the keyboard.
- **event-descri** (for ACPI hotkeys): Contains the ACPI event description of the key.
- **keycode**: Contains the name of the code. The list of keycode names may be found in the header file `<linux/input.h>` (without `KEY_` prefix). Note that this keycode is the keycode which is used by the kernel and **not** by X.
- **name**: Contains the name of the key.
- **default-action**: Contains the default action to run when the key is pressed.

The **default-action** element can have one attribute named `action-type`. This attribute can have two values: `program` or `plugin`. Explanations for the meanings of the values are obvious. If the attribute is `plugin`, the **default-action** element will have a **plugin-name** (containing the name of the plugin) and a **plugin-function** (containing the name of the plugin function) element. If the attribute is `program` the **default-action** element will contain the command of the program to run. The **default-action** element will also contain the command when `action-type` is not set.

So the **key-list** element can look like:

```
<key-list>
  <key>
    <name>Play/Pause</name>
    <scancode>148</scancode>
    <keycode>PLAY</keycode>
    <default-action action-type="plugin">
      <plugin-name>XMMS</plugin-name>
      <plugin-function>Play/Pause</plugin-function>
    </default-action>
  </key>
  <key>
    <name>WWW Home</name>
    <scancode>150</scancode>
    <keycode>WWW</keycode>
    <default-action action-type="plugin">
      <plugin-name>WWW Browser</plugin-name>
      <plugin-function>Home</plugin-function>
    </default-action>
  </key>
  <key key-type="acpi-hotkey">
    <name>E-Mail</name>
    <event-descri>hotkey ATKD 0000005d 00000002</event-descri>
    <keycode>EMAIL</keycode>
    <default-action action-type="plugin">
      <plugin-name>E-Mail</plugin-name>
      <plugin-function>E-Mail</plugin-function>
    </default-action>
  </key>
</key-list>
```

4. The keyboard configuration file

The keyboard configuration file contains information about the actions for all keys. It will be written by the graphical configuration program “keytouch”. The file has one root element called **keyboard**, which contains the following elements:

- **keyboard-name**: Contains the name of the keyboard.
- **key-list**: Contains for every key the settings.

The **keyboard-name** element contains two elements: **manufacturer** and **model**. These two elements contain the names of the manufacturer and the model respectively.

So the **keyboard-name** element can look like:

```
<keyboard-name>
  <manufacturer>The Keyboard Company</manufacturer>
  <author>Keyboard Number One</author>
</keyboard-name>
```

The **key-list** element has for every key an element called **key**. The key element has two elements: name and action (the contents of them are obvious). The action element has an attribute called *isdefault* which can have the values *true* or *false*. It indicates if the action is the default action for that key. The action element also has another attribute called *action-type*. This attribute can have two values: *program* or *plugin*. Explanations for the meanings of the values are obvious. If the attribute is *plugin*, the **default-action** element will have a **plugin-name** (containing the name of the plugin) and a **plugin-function** (containing the name of the plugin function) element. If the attribute is *program* the **default-action** element will contain the command of the program to run. The **default-action** element will also contain the command when *action-type* is not set.

So a **key-setting** element can look like:

```
<key>
  <name>WWW Home</name>
  <action isdefault="true" action-type="plugin">
    <plugin-name>WWW</plugin-name>
    <plugin-function>Home</plugin-function>
  </action>
</key>
```

5. The preferences file

The preferences file contains the users preferences and is written by the program “keytouch”. The preferences can be used by the special action plugins of keyTouch.

The file has a root element named **preferences**. This element contains the following sub-elements:

- **documents-dir**: Contains the directory where the user stores his/her documents.
- **home-page**: Contains the URL of the users home page.
- **search-page**: Contains the URL of users favorite search engine.
- **browser**: Contains the command of the preferred web browser.
- **email-program**: Contains the command of the preferred e-mail client.
- **chat-program**: Contains the command of the preferred chat client.

So a preferences file can look like:

```
<preferences>
  <documents-dir>~/Documents</documents-dir>
  <home-page>http://keytouch.sf.net</home-page>
  <search-page>http://www.google.com</search-page>
  <browser>mozilla-firefox</browser>
  <email-program>evolution</email-program>
  <chat-program>gaim</chat-program>
</preferences>
```

6. The current keyboard file

The current keyboard file contains the name of the keyboard that was chosen by the user. The root element is called **current-keyboard** and has two elements: **manufacturer** and **model**. These two sub-elements contain the name of the manufacturer and the name of the model respectively.

Example:

```
<current-keyboard>  
  <manufacturer>The Keyboard Company</manufacturer>  
  <model>Keyboard Number One</model>  
</current-keyboard>
```