

# Собственные уведомления (notify) в Gnome

- Настройка Linux

Речь пойдет об извещениях которые показываю практически все приложения от Rhythmbox до сетевых подключений. Хотелось иметь возможность показывать в таком виде собственные и поэтому я взял бубен и начал капать )

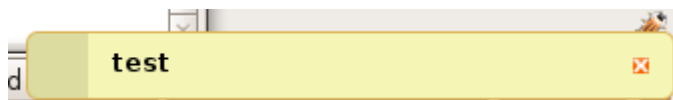
для начала понадобится установить пакет libnotify-bin, я под ubuntu запустил команду

```
sudo apt-get install libnotify-bin
```

после этого небольшой тест. Проверяем в консоле

```
notify-send test
```

если у вас появилась такое окошко значит можно двигаться дальше и разбираться с настройками



notify-send [OPTION...] [BODY] — create a notification

**-u, --urgency=LEVEL**

Вариантов здесь не так много, low, normal, critical. Разница лишь в полоске которая появляется с левой стороны.

**-t, --expire-time=TIME** Specifies the timeout in milliseconds at which to expire the notification.

Здесь все понятно, указываем время в миллисекундах которе извещение будет висеть на экране.

**-i, --icon=ICON[,ICON...]** Specifies an icon filename or stock icon to display.

С иконкой тоже все замечательно, Во первых можно прямо указать путь на любое изображение на диске. Другой вариант это открыть /usr/share/icons/hicolor/48x48/apps/ и выбрать там любую и указать ее название только без расширения notify-send -i pidgin test

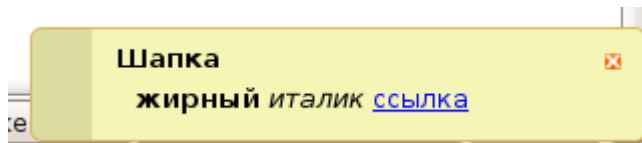
**-c, --category=TYPE[,TYPE...]** Specifies the notification category.

по идеи задание категории должно было дать результат но я поверьте его так и не заметил, поэтому пойдем дальше

**-h, --hint=TYPE:NAME:VALUE** Specifies basic extra data to pass. Valid types are int, double, string and byte.

хинты дают возможность большу возможность по настройке, например местоположение на экране где должно появится извещение -h int:x:500 -h int:y:500

теперь по тексту, принимаются 2 текста, первый идет в шапку, второй это тело с небольшими возможностями гипертекста



```
notify-send «Шапка» "<b>жирный <i>италик <a href=«a.html»>ссылка</a>"
```

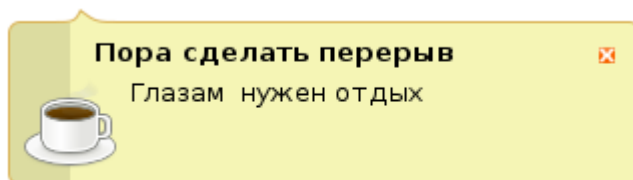
после того как разобрались с настройками сразу начинается мысль, а куда это применить и первое это собственный органайзер. Все время говорят что надо отрываться от монитора каждые 45 минут, значит есть шанс это сделать. Шаг первый это написать небольшой скриптик

```
#!/bin/sh
```

```
export DISPLAY=:0.0
PATH=/usr/bin/:/bin/
notify-send --urgency normal --expire-time=10000 -i typing-monitor -h int:x:500 -
h int:y:500 "Пора сделать перерыв" "Глазам нужен отдых"
```

что здесь важно, это DISPLAY, крон его не видит в упор и поэтому команды хоть и будут выполняться на экране ничего не появится. сохраняем ставим на исполнение и проверяем

```
chmod a+x sh1.sh
./sh1.sh
```



результат понравился? тогда дело за настройкой крона

```
45 * * * * /home/rus/sh1.sh
```

Ну вот

за дополнительной информацией обращайтесь на <http://www.galago-project.org/specs/notification/0.9/x211.html>

кроспостинг <http://mymans.org/2008/12/26/500>

Дополнение от [spiritedflow](#)

Насчет DISPLAY=:0.0. Жестковбитый DISPLAY — не хорошо. Он время от времени может меняться, даже, если всего один пользователь за компьютером. Кроме того, если ты закрыл сессию и ушел, крон будет все равно работать и ошибки будут сыпаться тебе в почту.

Лучше определять его автоматом, например, с помощью вот такой вот функции:

```
# get_display [USER] — Returns $DISPLAY of USER.  
# If first param is omitted, then $LOGNAME will be used.  
get_display () {  
  who \  
  | grep ${1:-$LOGNAME} \  
  | perl -ne 'if ( m!\(:\d+\)\$! ) {print ":$1.0\n"; $ok = 1; last} END {exit !$ok}'  
}
```

И в скрипте поправить вот так (в одной строчке нельзя, так как export сбивает \$?):

```
DISPLAY=$(get_display) || exit  
export DISPLAY
```

Теги:

- [ubuntu](#)
- [gnome](#)