

ubuntu 

Ciclo di sviluppo

Paolo Sammiceli <xdatap1@ubuntu.com>
Andrea Colangelo <warp10@ubuntu.com>

Ubuntu Party 2011
Schio, Palazzo Toaldi Capra

1 Maggio 2011



Salve a tutti e benvenuti!

Oggi vedremo come partecipare al ciclo di sviluppo di Ubuntu.



Due parole su di me, mi chiamo **Paolo Sammicheli**, sono un informatico di professione e, nel tempo libero, partecipo allo sviluppo di Ubuntu.

In Ubuntu mi occupo di diverse cose. Con il gruppo italiano mi occupo di Traduzioni, di Marketing e Comunicazione ed inoltre coordino il gruppo italiano di Quality Assurance, ovvero facciamo i test del software in corso di sviluppo.

Io invece sono **Andrea Colangelo**, studio Ingegneria Informatica e anche io nel tempo libero sono coinvolto in Ubuntu.

Sono un Ubuntu Developer da oltre tre anni, e mi occupo in particolare di Quality Assurance e di pacchetti in Python. Sono attivo nella comunità italiana nel Gruppo Promozione, in particolare nella newsletter, nel Progetto CD e nel Progetto Relatori.



www.ubuntu-party-it/slide/partecipare2.pdf

Intanto una buona notizia. Non avete da prendere appunti. Le slide che vi mostreremo, complete di note con quello che diciamo sono già online a questo indirizzo.



Vediamo intanto che cosa è Ubuntu per chi non avesse partecipato al talk di ieri.



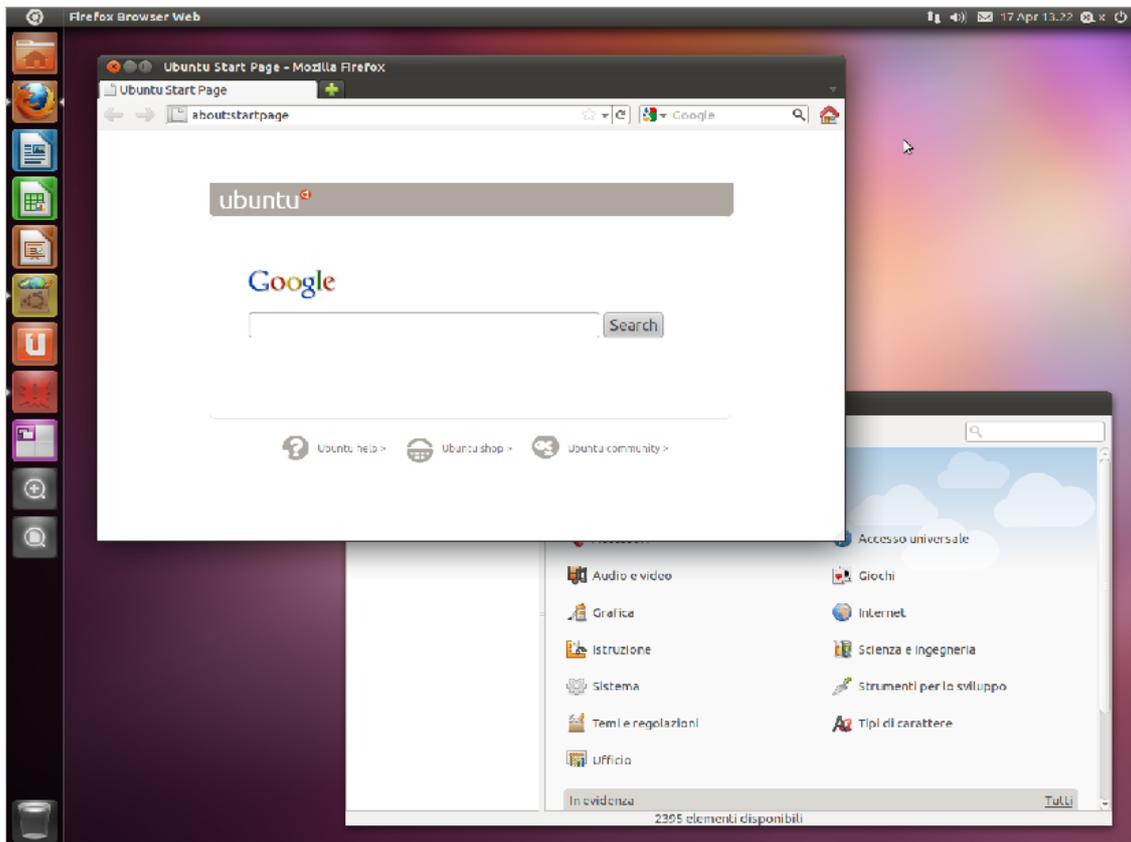
Ubuntu è innanzitutto un'antica parola Africana dal significato molto profondo.



**«lo sono ciò che sono
per merito di ciò
che siamo tutti»**

«lo sono ciò che sono per merito di ciò che siamo tutti» è una traduzione di questa parola.

Richiama il genere umano allo spirito di comunità anziché di individualismo.



Ma Ubuntu è anche una distribuzione GNU/Linux.



Ubuntu è stata fondata da Mark Shuttleworth, giovane imprenditore Sud Africano che nel 1999 ha venduto la propria azienda, Thawte ad una grossa azienda americana, Verisign, guadagnando 575 Milioni di Dollari Americani.

E cosa fa, secondo voi, un “ragazzo” di 26 anni con in mano 575 Milioni di Dollari? Beh, Mark si è pagato un viaggio nello spazio, è stato il secondo turista nello spazio.

Qui lo vedere dentro la stazione spaziale internazionale.



La comunità di Ubuntu è una comunità internazionale formata da Volontari e professionisti che collaborano per creare la distribuzione e da un'azienda: Canonical (<http://www.canonical.com>).

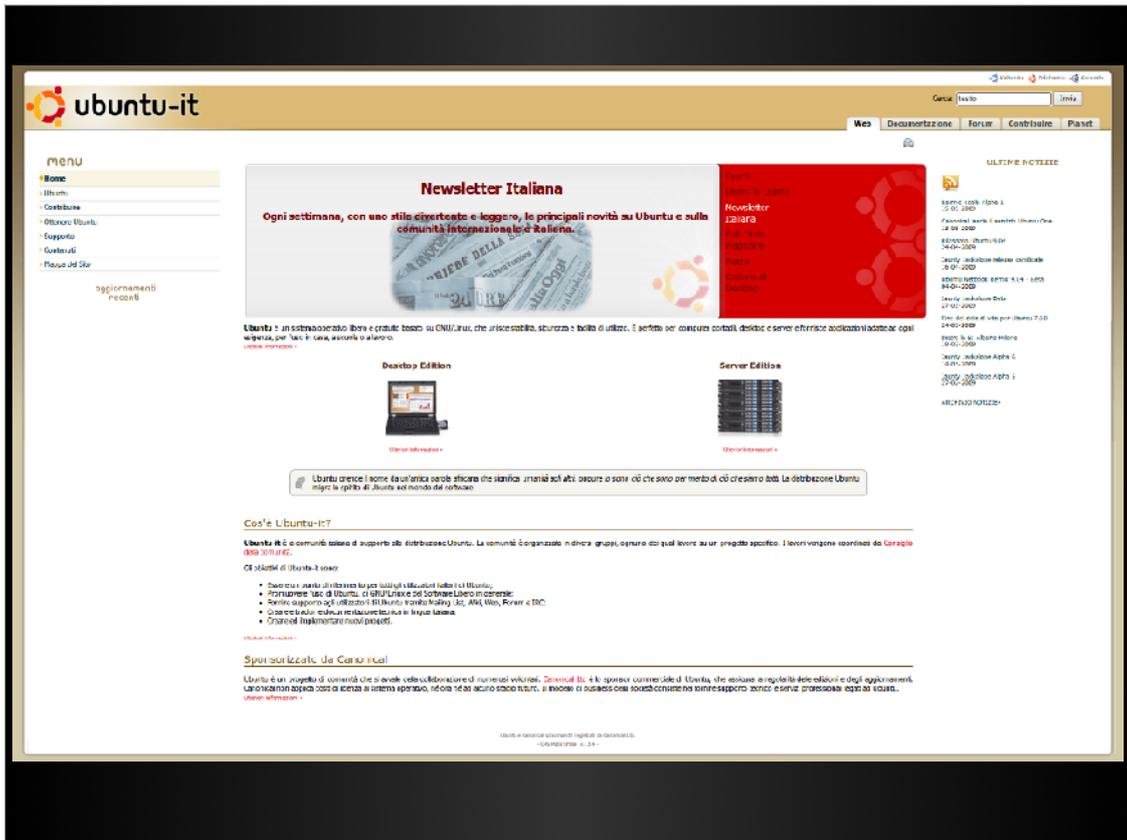
Canonical è l'azienda fondata da Mark per sponsorizzare lo sviluppo di Ubuntu.



Esistono poi delle organizzazioni nazionali, chiamati LoCo Team, ovvero Local Community. Sono uno per ogni stato negli Stati Uniti e uno per ogni nazione nel resto del mondo.

Quella che vedete è la foto di un meeting della comunità Italiana.

Ulteriori informazioni:
<http://www.ubuntu-it.org>



Due parole sulla comunità Italiana:

Ubuntu-it ha un sito raggiungibile all'indirizzo www.ubuntu-it.org o anche www.ubuntu.it.



Che cosa offre Ubuntu-it?

Innanzitutto Supporto Tecnico agli utenti tramite molti strumenti: Forum, Mailing List, wiki, Irc, ecc.

Ulteriori informazioni:

<http://www.ubuntu-it.org/index.php?page=supporto-della-comunita>



Inoltre ci occupiamo di tradurre in Italiano il Software, oltre che molti documenti e articoli. Inoltre traduciamo anche una rivista dedicata ad Ubuntu: Full Circle Magazine. La potete scaricare liberamente dal sito.

Ulteriori informazioni:

<http://wiki.ubuntu-it.org/GruppoTraduzione>



E produciamo documentazione tecnica e guide in Italiano, sia traducendola dall'inglese che scrivendone di originali. Tutte le guide sono sul wiki. Sapete cosa è un wiki vero? È il motore che anima anche wikipedia. È un sito web che tutti possono modificare.

Ulteriori informazioni:

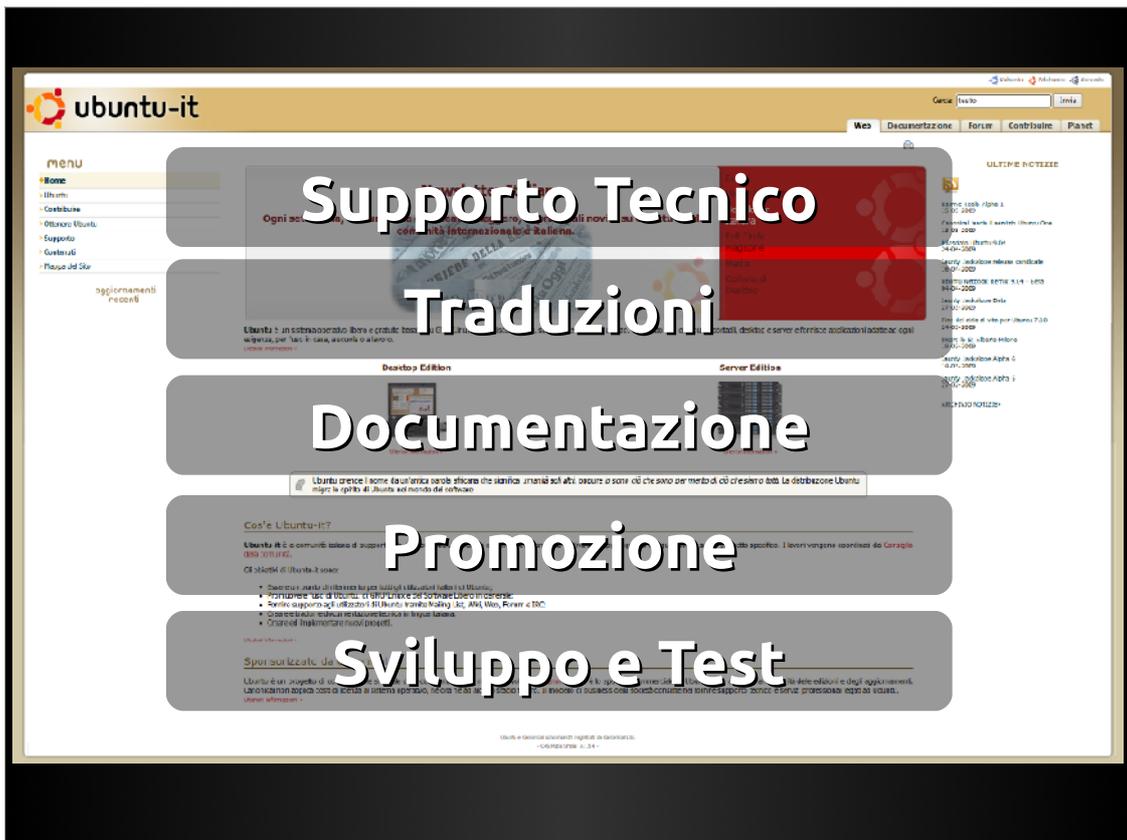
<http://wiki.ubuntu-it.org/GruppoDocumentazione>



E ci occupiamo della promozione e diffusione di Ubuntu, come ad esempio andare a delle conferenze e parlare di noi :)

Ulteriori informazioni:

<http://wiki.ubuntu-it.org/GruppoPromozione>

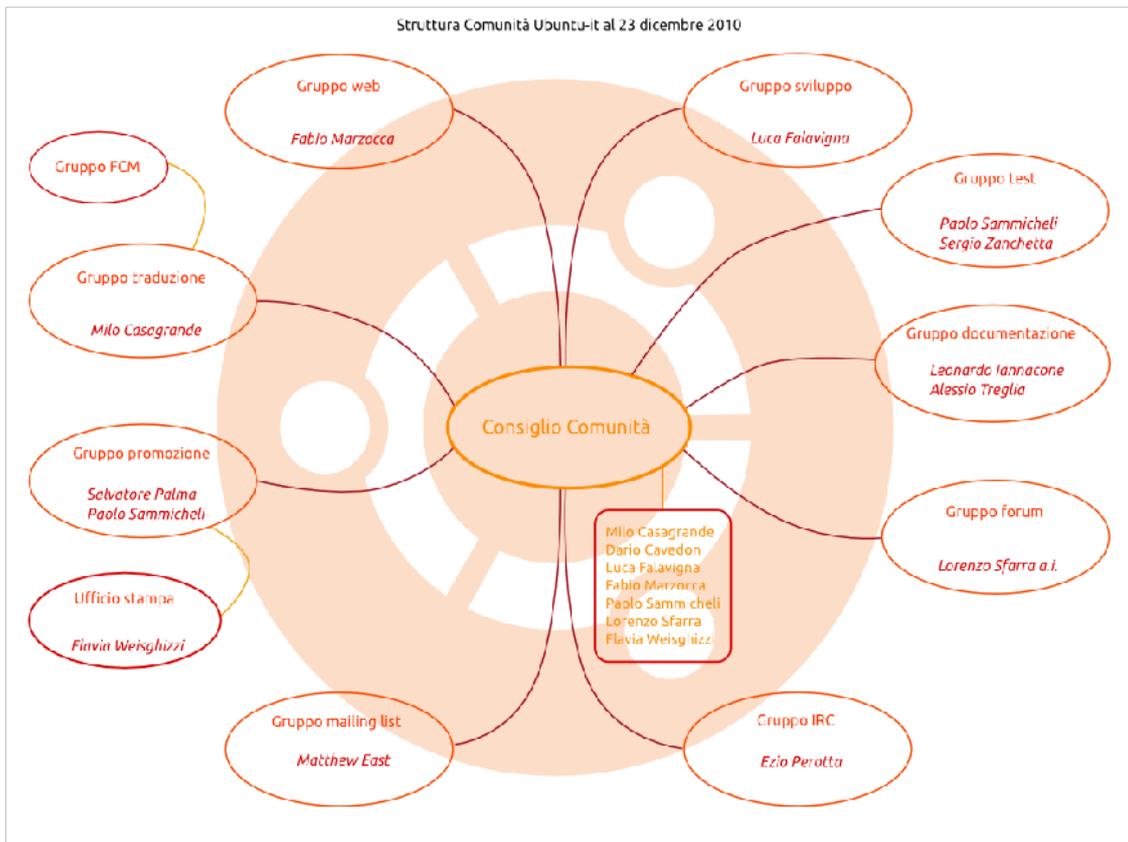


Inoltre partecipiamo con la comunità internazionale nello sviluppo della distribuzione stessa e nei test per cercare di migliorare i programmi che compongono Ubuntu.

Ulteriori informazioni:

<http://wiki.ubuntu-it.org/GruppoSviluppo>

<http://wiki.ubuntu-it.org/GruppoTest>



La comunità è divisa in gruppi di lavoro, che sono coordinati dal Consiglio della Comunità.

Ulteriori informazioni:

http://www.ubuntu-it.org/contribuire/Struttura_Com.shtml



Vediamo adesso come partecipare alla comunità di Ubuntu.

La prima cosa da fare è aprire i propri account e preparare la pagina personale sul wiki.

In pratica è un modo per essere riconoscibili all'interno della comunità. Considerate che la comunità di Ubuntu è molto vasta, quindi è difficile ricordarsi di tutti a memoria. La vostra pagina personale parla di voi e vi presenta agli altri.

Codice di Condotta

Vi viene anche chiesto, come prima cosa, di firmare il Codice di Condotta di Ubuntu con una chiave crittografica.

CODICE DI CONDOTTA

Siate premurosi. Il vostro lavoro sarà usato da altre persone, e voi a vostra volta dipenderete dal lavoro degli altri. Ogni decisione presa coinvolgerà utenti e colleghi, e ci aspettiamo che prendiate in considerazione le conseguenze di ogni decisione. Ad esempio, quando siamo in uno stato di "freeze", non fate drammatici upload di nuove versioni di software per sistemi critici, in quanto altre persone sono in fase di test dei sistemi "congelati" e non sono in grado di assorbire grandi variazioni.

Siate rispettosi. La comunità Ubuntu ed i suoi membri si rivolgono l'un l'altro con grande rispetto. Ciascuno può realizzare un valido contributo ad Ubuntu. Non possiamo sempre essere d'accordo, ma il disaccordo non è una scusa per un comportamento e per modi scorretti. Potremmo tutti vivere qualche frustrazione talvolta, ma non potremmo mai permettere che tale frustrazione si trasformi in un attacco personale. È importante ricordare che una comunità dove le persone si sentono a disagio non è una comunità produttiva. Ci aspettiamo che i membri della comunità Ubuntu siano rispettosi sia quando hanno a che fare con altri collaboratori, sia con persone al di fuori del progetto Ubuntu, sia con gli utenti.

Siate collaborativi. Ubuntu e Free Software collaborano e lavorano insieme. La collaborazione riduce la ridondanza del lavoro compiuto del mondo Free Software e migliora la qualità del software prodotto. Dovreste tendere a collaborare con altri maintainers Ubuntu, così come con la comunità a monte che è interessata al vostro lavoro. Il vostro lavoro dovrà essere eseguito con trasparenza e le patch per Ubuntu devono essere consegnate alla comunità quando si rendono disponibili, non al rilascio dell'edizione. Se volete lavorare a nuovo codice per progetti esistenti, almeno mantenete informati delle vostre idee e progressi i responsabili di quei progetti. Potrebbe non essere possibile ottenere il consenso circa la corretta implementazione di un'idea, così non sentitevi obbligati ad ottenere un accordo prima di iniziare, ma almeno mantenete informato del vostro lavoro il mondo esterno, e pubblicatelo in modo tale da consentire altri di svolgere prove, discussioni e contribuire ai vostri sforzi.

Quando non siete d'accordo, consultate gli altri. Disaccordi, sia politici che tecnici, avvengono ogni giorno e la comunità Ubuntu non ne è esente. L'obiettivo importante non è evitare i disaccordi o le diverse vedute, ma di risolverli costruttivamente. Dovreste sempre tornare alla comunità ed ai suoi processi per cercare consigli e risolvere disaccordi. Ci sono sia il Technical Board che il Community Council che vi aiuteranno a decidere il giusto corso di Ubuntu. Ci sono inoltre diversi Project Teams e Team Leaders, che vi aiuteranno a capire quale direzione potrebbe essere la più accettabile. Se alla fine volete comunque prendere una strada diversa, vi invitiamo a fornire una diversa distribuzione o un set di pacchetti alternativo usando la struttura dell'Ubuntu Package Management, affinché la comunità possa comunque provare i vostri cambiamenti e le vostre idee, e contribuire alla discussione.

Quando non siete sicuri, chiedete. Nessuno sa tutto, e nessuno si aspetta che l'altro sia perfetto nella comunità Ubuntu. Rivolgere domande evita molti problemi lungo il percorso, e quindi le domande sono incoraggiate. Coloro che devono rispondere, dovranno essere reattivi e di grande aiuto. Comunque, nel porre una domanda, occorre avere cura nel rivolgersi al forum appropriato. Domande fuori-tema, come ad esempio una richiesta di supporto in una mailing list di sviluppo distoglie da una discussione produttiva.

Lasciate con considerazione. Gli sviluppatori di ogni progetto vanno e vengono, e per Ubuntu non è diverso. Quando lasciate un progetto, del tutto o in parte, fatelo cercando di minimizzare le ripercussioni sul progetto stesso. Ciò significa che dovrete avvisare prima di lasciare e intraprendere le opportune azioni per assicurare che gli altri possano riprendere dal punto da voi lasciato.

Questo è il codice di condotta di Ubuntu, come vedete è un po' lunghetto.

CODICE DI CONDOTTA

Siate premurosi. Il vostro lavoro sarà usato da altre persone, e voi a vostra volta dipenderete dal lavoro degli altri. Ogni decisione presa coinvolgerà utenti e colleghi, e ci aspettiamo che prendiate in considerazione le conseguenze di ogni decisione. Ad esempio, quando siamo in uno stato di "freeze", non fate drammatici upload di nuove versioni di software per sistemi critici, in quanto altre persone sono in fase di test dei sistemi "congelati" e non sono in grado di assorbire grandi variazioni.

Siate rispettosi. La comunità Ubuntu ed i suoi membri si rivolgono l'un l'altro con grande rispetto. Ciascuno può realizzare un valido contributo ad Ubuntu. Non possiamo sempre essere d'accordo, ma il disaccordo non è una scusa per un comportamento e per modi scorretti. Potremmo tutti vivere qualche frustrazione talvolta, ma non potremmo mai permettere che tale frustrazione si trasformi in un attacco personale. È importante ricordare che una comunità dove le persone si sentono a disagio non è una comunità produttiva. Ci aspettiamo che i membri della comunità Ubuntu siano rispettosi sia quando hanno a che fare con altri collaboratori, sia con persone al di fuori del progetto Ubuntu, sia con gli utenti.

Siate collaborativi. Ubuntu e Free Software collaborano e lavorano insieme. La collaborazione riduce la ridondanza del lavoro compiuto del mondo Free Software e migliora la qualità del software prodotto. Dovreste tendere a collaborare con altri maintainers Ubuntu, così come con la comunità a monte che è interessata al vostro lavoro. Il vostro lavoro dovrà essere eseguito con trasparenza e le patch per Ubuntu devono essere consegnate alla comunità quando si rendono disponibili, non al rilascio dell'edizione. Se volete lavorare a nuovo codice per progetti esistenti, almeno mantenete informati delle vostre idee e progressi i responsabili di quei progetti. Potrebbe non essere possibile ottenere il consenso circa la corretta implementazione di un'idea, così non sentitevi obbligati ad ottenere un accordo prima di iniziare, ma almeno mantenete informato del vostro lavoro il mondo esterno, e pubblicatelo in modo tale da consentire altri di svolgere prove, discussioni e contribuire ai vostri sforzi.

Quando non siete d'accordo, **consultate gli altri.** Disaccordi, sia politici che tecnici, avvengono ogni giorno e la comunità Ubuntu non ne è esente. L'obiettivo importante non è evitare i disaccordi o le diverse vedute, ma di risolverli costruttivamente. Dovreste sempre tornare alla comunità ed ai suoi processi per cercare consigli e risolvere disaccordi. Ci sono sia il Technical Board che il Community Council che vi aiuteranno a decidere il giusto corso di Ubuntu. Ci sono inoltre diversi Project Teams e Team Leaders, che vi aiuteranno a capire quale direzione potrebbe essere la più accettabile. Se alla fine volete comunque prendere una strada diversa, vi invitiamo a fornire una diversa distribuzione o un set di pacchetti alternativo usando la struttura dell'Ubuntu Package Management, affinché la comunità possa comunque provare i vostri cambiamenti e le vostre idee, e contribuire alla discussione.

Quando non siete sicuri, **chiedete.** Nessuno sa tutto, e nessuno si aspetta che l'altro sia perfetto nella comunità Ubuntu. Rivolgere domande evita molti problemi lungo il percorso, e quindi le domande sono incoraggiate. Coloro che devono rispondere, dovranno essere reattivi e di grande aiuto. Comunque, nel porre una domanda, occorre avere cura nel rivolgersi al forum appropriato. Domande fuori-tema, come ad esempio una richiesta di supporto in una mailing list di sviluppo distoglie da una discussione produttiva.

Lasciate con considerazione. Gli sviluppatori di ogni progetto vanno e vengono, e per Ubuntu non è diverso. Quando lasciate un progetto, del tutto o in parte, fatelo cercando di minimizzare le ripercussioni sul progetto stesso. Ciò significa che dovrete avvisare prima di lasciare e intraprendere le opportune azioni per assicurare che gli altri possano riprendere dal punto da voi lasciato.

Non vi fate spaventare dalla lunghezza, il codice di condotta è abbastanza semplice e può essere sintetizzato con alcune parole chiave.

CODICE DI CONDOTTA

Siate premurosi.

**Siate
rispettosi.**

Siate collaborativi.

consultate gli altri.

chiedete.

Lasciate con considerazione.

Questi sono gli inviti che il codice di condotta fa a chi è membro della comunità Ubuntu. Come vedete sono principi semplici e condivisibili ma contraddistinguono lo stile con cui la comunità Ubuntu si pone alle cose.

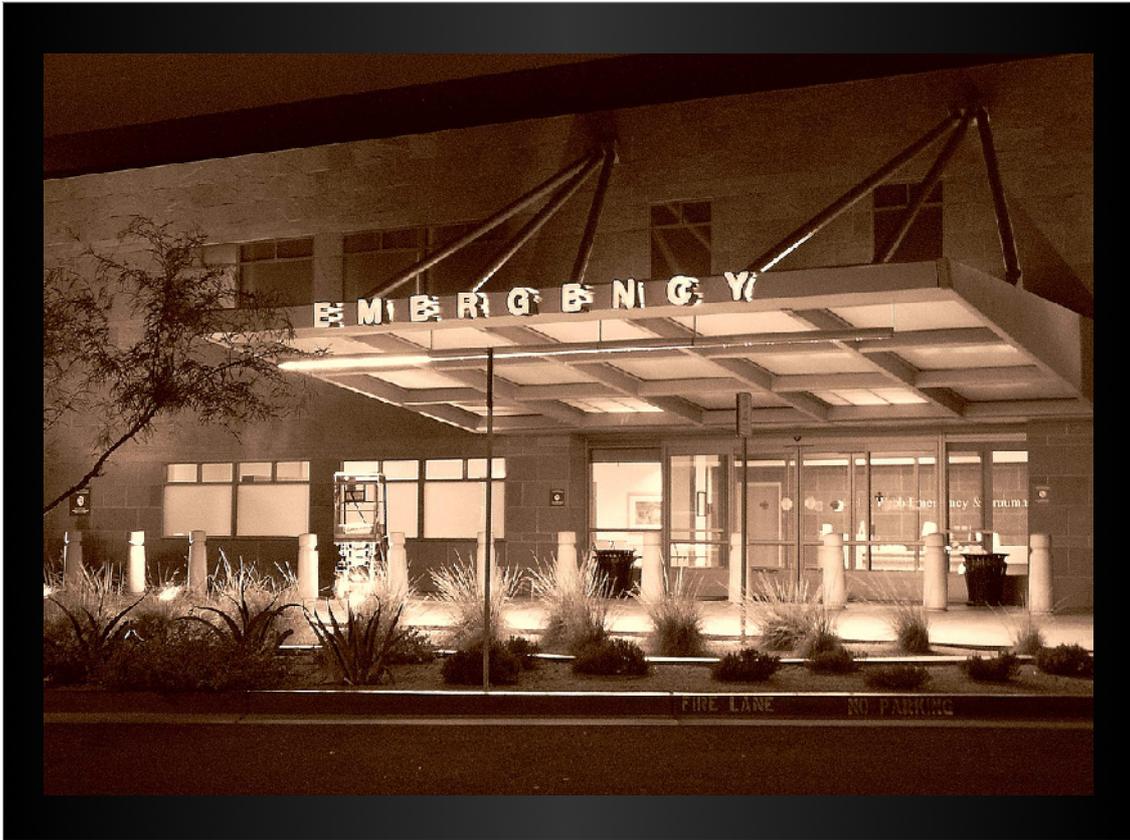
Una comunità serena ed in armonia è anche una comunità produttiva.

Il codice di condotta vuole mantenere un bel clima di rispetto all'interno della comunità.



Ieri abbiamo visto come segnalare un BUG. Vediamo adesso il processo che porta un bug dalla sua segnalazione fino alla sua approvazione per essere sottoposto ai gruppi di sviluppo.

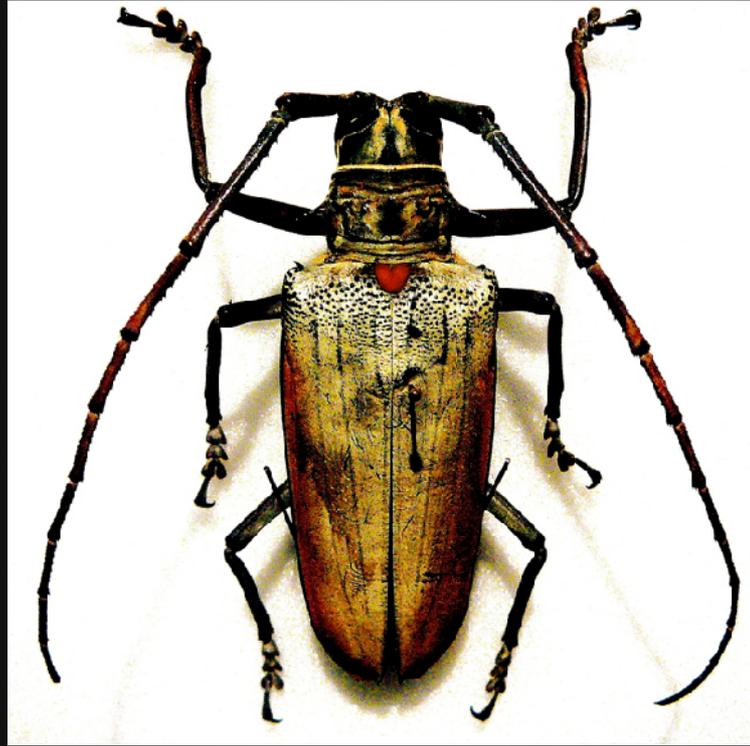
Cosa significa, quindi, TRIAGE?



Più o meno tutti abbiamo avuto la sfortuna di andare in ospedale, per noi stessi o per accompagnare qualcuno. Se non si trattava di una visita programmata siamo dovuti passare dal Pronto Soccorso.



Al Pronto Soccorso viene svolta una procedura chiamata TRIAGE. In pratica si tratta di identificare i pazienti in arrivo, decidere dove devono andare e fornirgli una priorità in base all'urgenza del loro trattamento.



La stessa cosa la facciamo con gli odiati BUG.

Il Triage dei Bug viene condotto per buona parte dai volontari della comunità Internazionale. Anche voi potete partecipare al triage dei bug sveltendo così la loro correzione e migliorando Ubuntu.



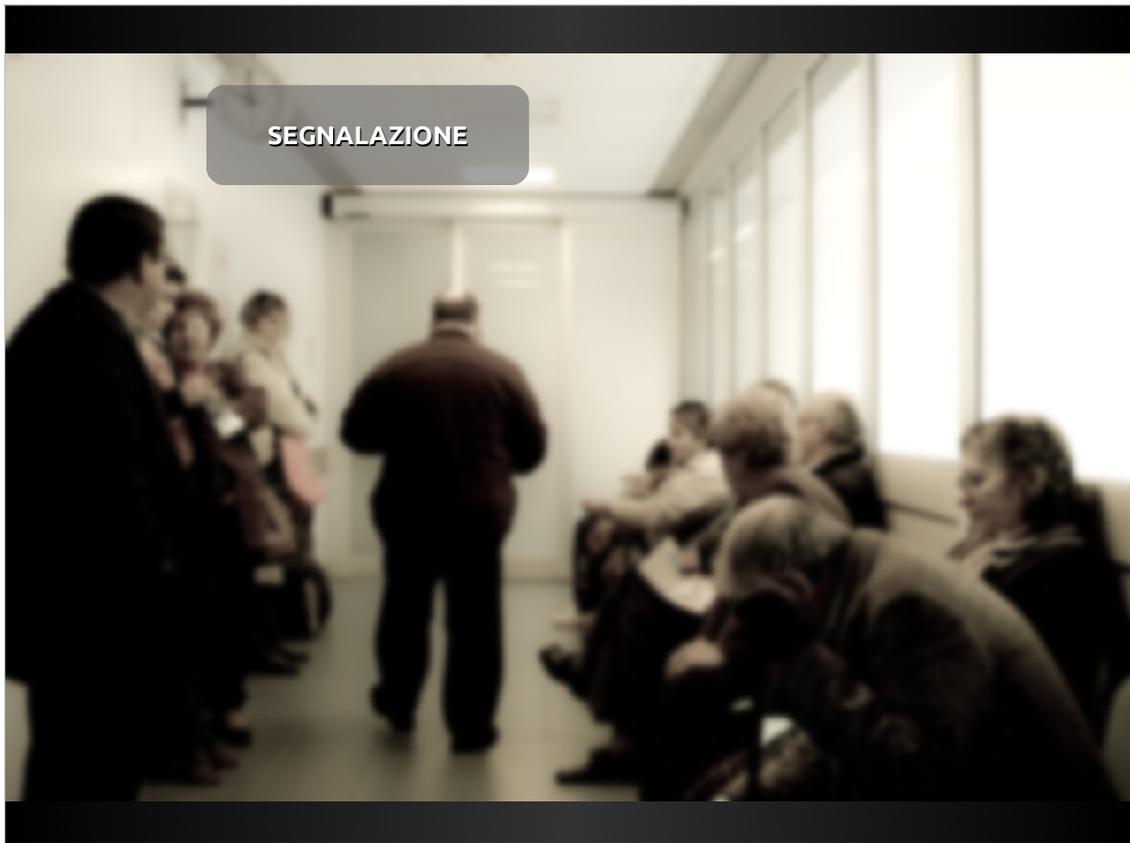
Perché è necessario eseguire un Triage di un Bug?

Come avviene per le prestazioni sanitarie non è possibile aggredire e risolvere tutti i problemi quando si presentano. Si forma in maniera naturale una sorta di fila di attesa.



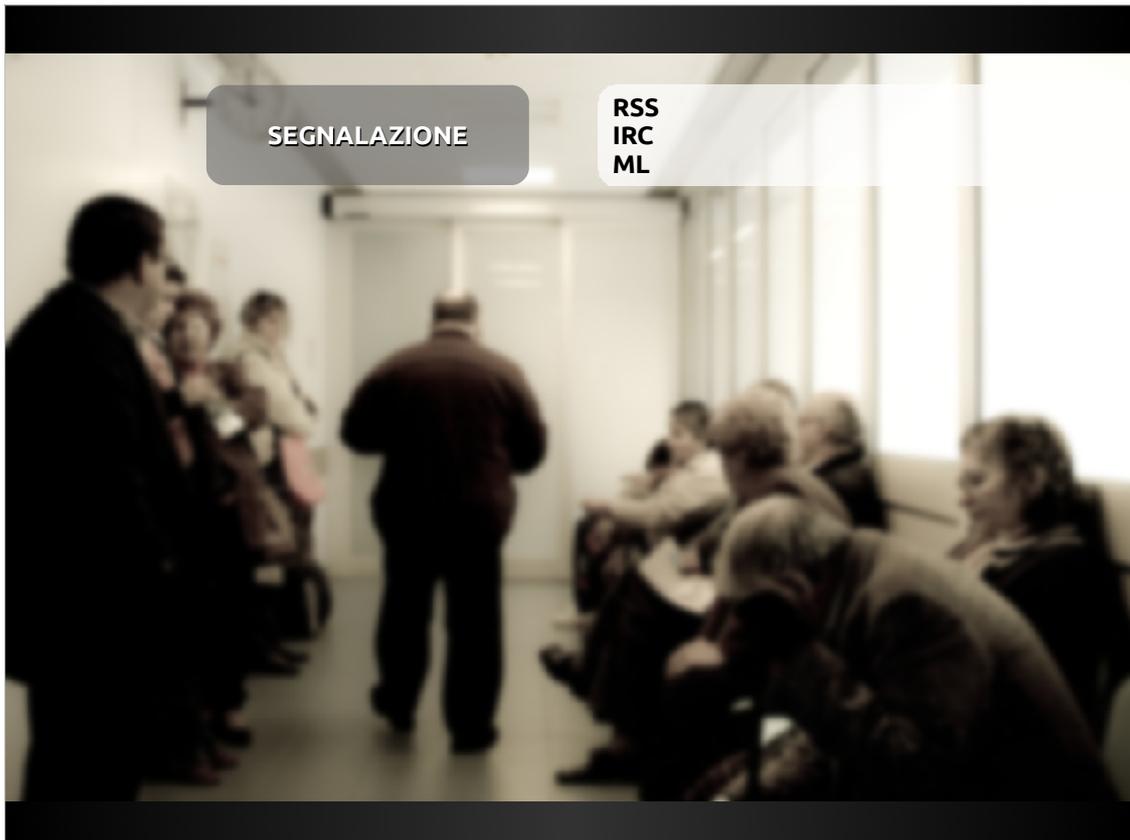
Il triage permette di catalogare i pazienti in attesa in modo che i casi più urgenti vengano serviti prima.

Vediamo i passi che si deve svolgere per fare il triage dei Bug in Ubuntu.



Innanzitutto, per fare il Triage vorremmo vedere l'elenco dei bug che vengono segnalati.

Essi si trovano su Launchpad



e le informazioni circa le nuove segnalazioni vengono propagate anche in altri canali, come i feed rss, il canale IRC e una mailing list apposita.



Iniziando a trattare un Bug, per prima cosa è utile determinare se la segnalazione è riferita ad un bug già segnalato precedentemente.



Questo per evitare di fare un lavoro doppio.



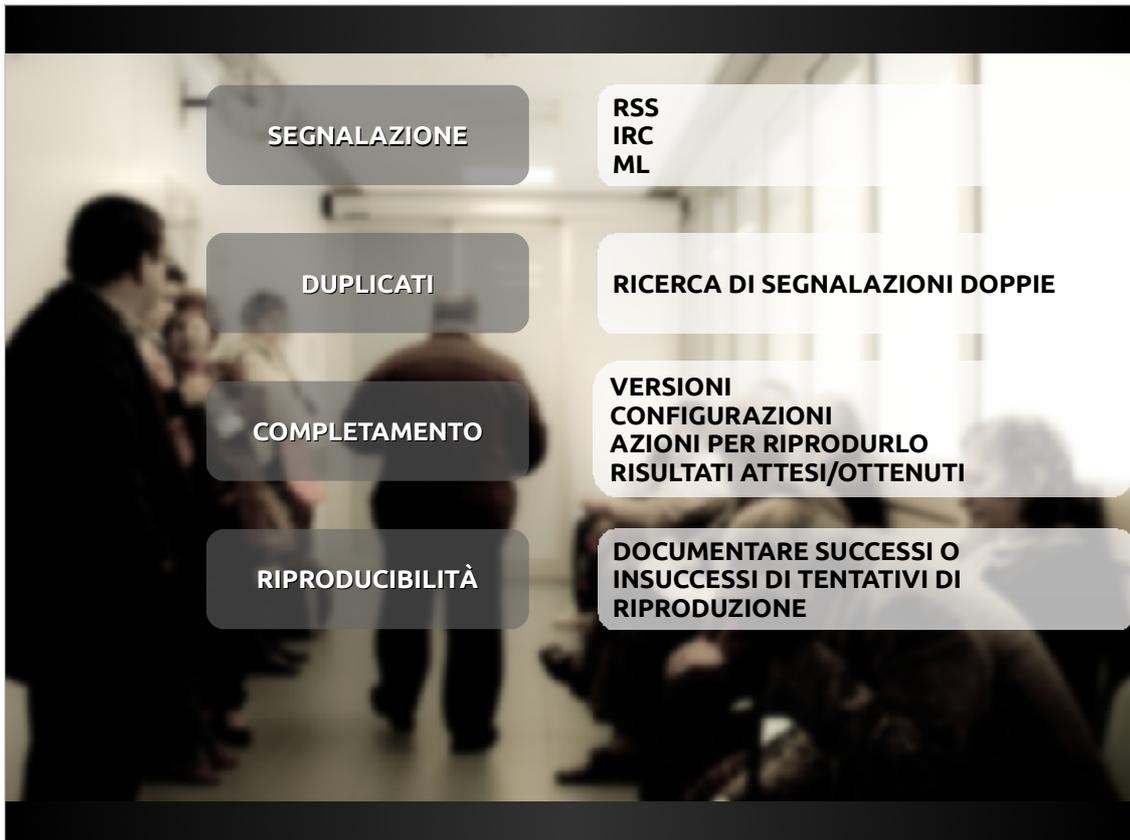
Dopodiché dobbiamo aiutare l'utente, tramite una sorta di dialogo che avviene nei commenti del bug, a completare la segnalazione, in modo che tutte le informazioni necessarie siano state scritte nella segnalazione di Bug.



Ad esempio è importante avere informazioni sulle versioni, sulle configurazioni e sui passi da compiere per riprodurlo.

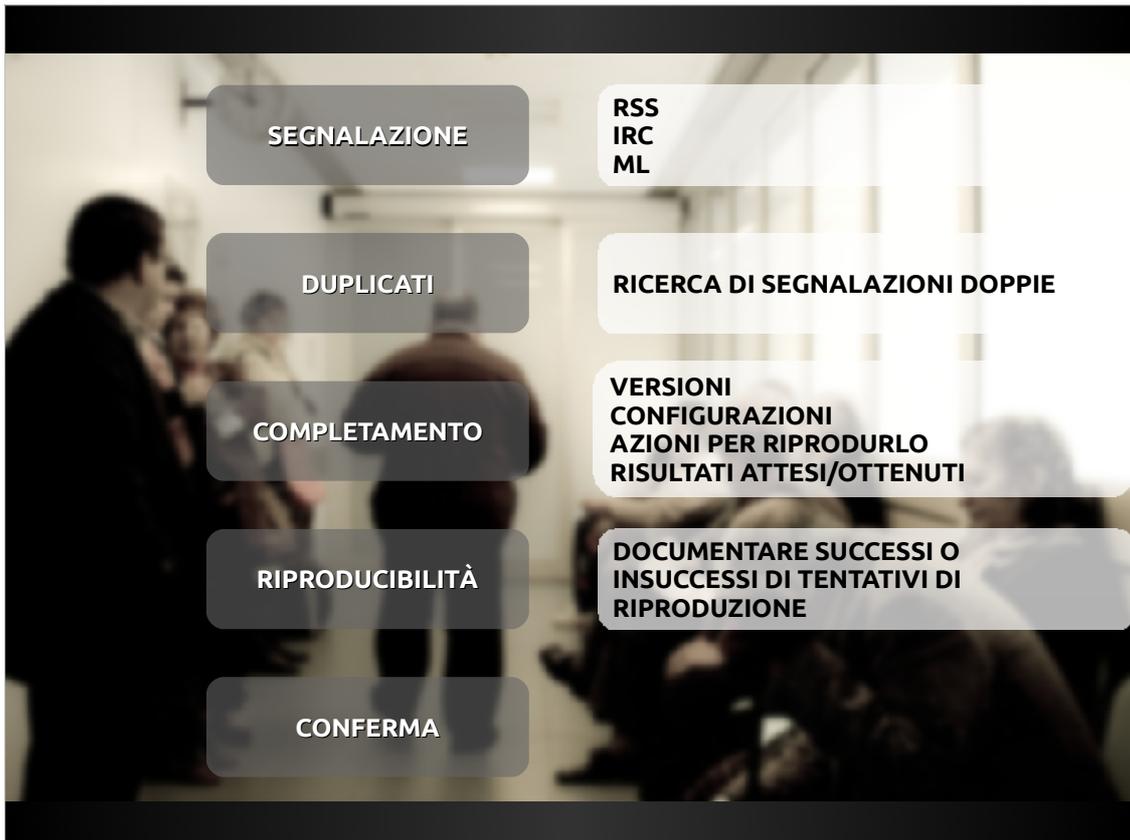


Quindi, coloro che effettuano il Triage di un Bug proveranno a riprodurlo.



Documentando i loro risultati. Sono utili anche risultati di insuccesso nella riproduzione in quanto possono indicare informazioni ulteriori.

Ad esempio, se il bug è dipendente da un particolare componente hardware.



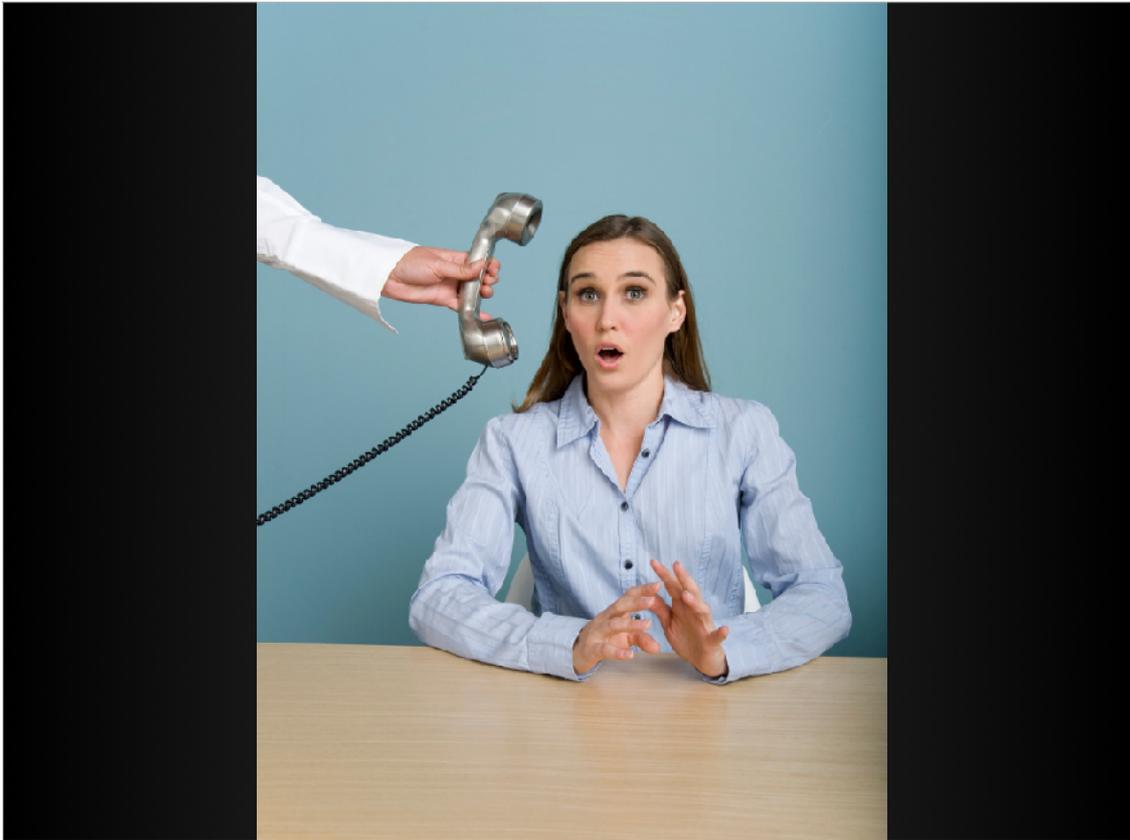
Se tutte le informazioni sono state inserite sul bug possiamo quindi confermarlo.



Con la conferma del bug si inizia a collegarlo con il resto del mondo. Ad esempio è importante, sui bug confermati, verificare se non sono già stati segnalati upstream e nel caso fossero di pertinenza è utile aprire la segnalazione anche là.

Launchpad permette di inserire link ad altri gestori di Bug dei progetti da cui Ubuntu deriva (Debian, Mozilla, GNOME, KDE, ecc).

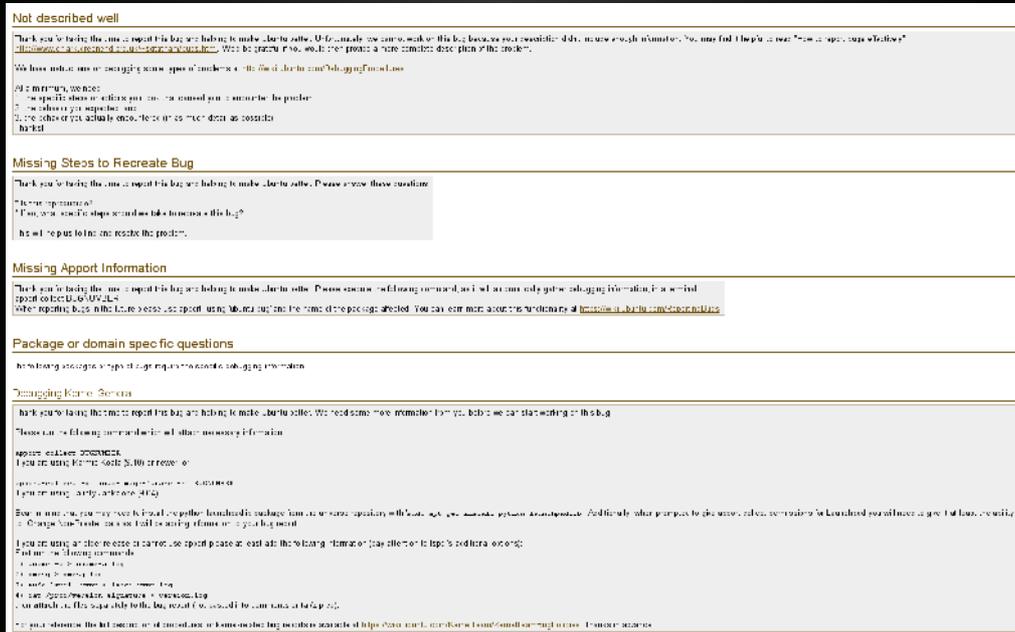
Inoltre, specie per i Bug di sicurezza, è importante fare ricerche su altre distribuzioni, anche se non facenti parti della catena di produzione di Ubuntu, in modo da avere una collaborazione più efficace.



Quando si inizia ad imparare a fare il Triage una delle prime difficoltà è che non si sa cosa rispondere a chi ha aperto il Bug.

Come ricordate il Codice di Condotta ci chiede di essere gentili, e la forma è importante nel comunicare con gli utenti che ci segnalano bug.

<https://wiki.ubuntu.com/Bugs/Responses>

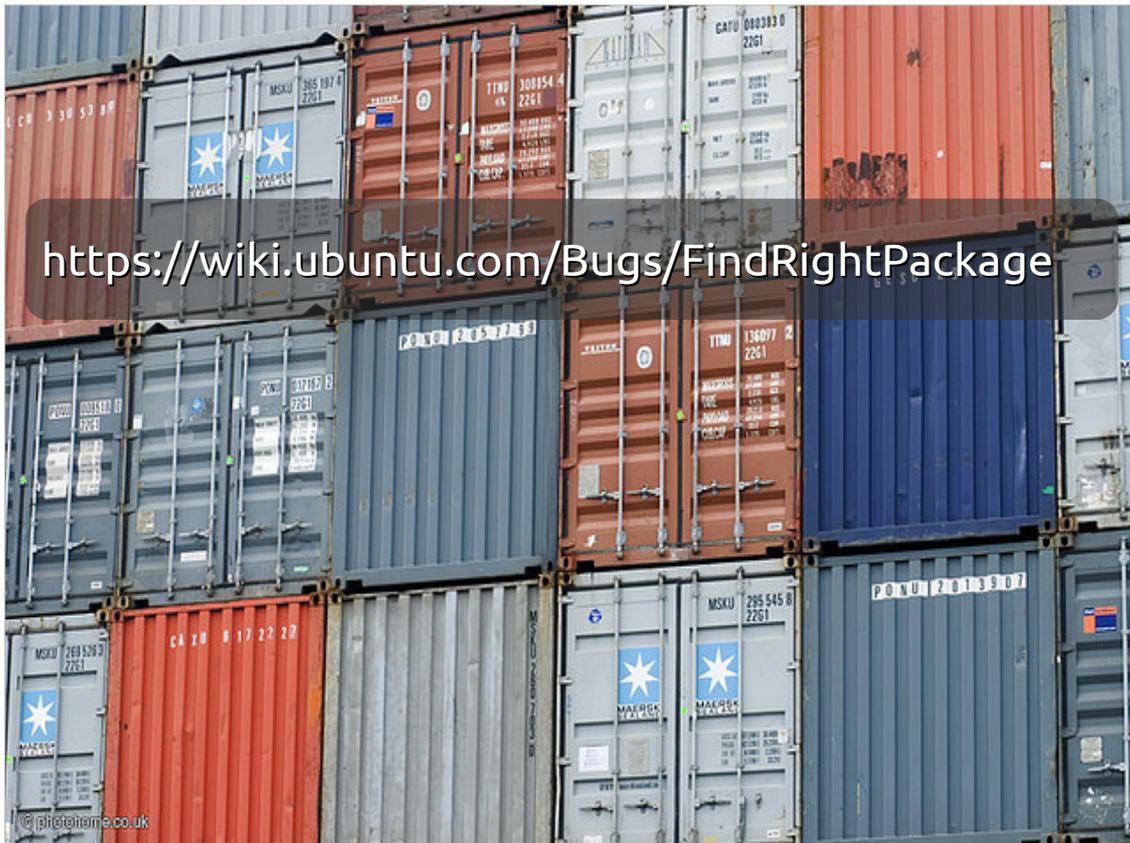


Ma per fortuna abbiamo una risorsa molto utile, le risposte tipo catalogate per categoria:

<https://wiki.ubuntu.com/Bugs/Responses>

È una pagina molto lunga e viene aggiornata anche abbastanza spesso.

Non esitate a consultarla ogni volta che dovete fare il Triage di un BUG.



Altro problema è individuare il pacchetto giusto a cui associare il BUG. Anche qui c'è una risorsa molto importante:

<https://wiki.ubuntu.com/Bugs/FindRightPackage>

Come la pagina precedente è molto lunga, non si presta ad essere imparata a memoria. Va seguita ogni volta alla ricerca delle informazioni corrette. Inoltre varia frequentemente, in funzione dei cambiamenti che si presentano nella distribuzione stessa. Chi è interessato al Triage troverà utile sottoscrivere quella pagina in modo da ricevere mail sulle modifiche che vengono apportate.

The screenshot shows a bug report on the Ubuntu Launchpad website. The title is "Toshiba laptop battery is drained while shut down". The bug is reported by Deema on 2007-04-28. It affects 17 people. The bug is linked to several other bugs and milestones.

Affects	Status	Importance	Assigned to	Milestone
Linux	Invalid	Unknown	linux-kernel-bugs # 0797	
linux-2.6.26 (Debian)	Fix Released	Unknown	debbugs #481493	
linux (Ubuntu)	Confirmed	High	Stefan Bader	

The bug description states: "After shutting down the laptop, the battery still slowly drains, going from 100% to 70% overnight. And, no, it's not sleep mode. This has been confirmed by other people with Toshiba models such as M115, M100 and many others. No such problem in Windows. This is in Eddy, Feisty, and Hardy as well as Jaunty."

Tags: cherry-pick cft-2.6.27

Dave Gilbert wrote on 2007-04-28:

```
I was about to report the same thing: mine is a Toshiba Equium A100-306. It does appear to switch off fully (screen fully black, no disc sounds, onlight goes off), but there is obviously something still eating power (I think about a 50% drain over night; totally flat left in hibernate for 5 days). I've included a section of syslog that looks suspicious to me: note the bit about USB (I've not got any USB devices plugged in): (Is it normal there there is the remnants of the logs from the shutdown (Apr 22) finding their way into the logs on the 26th when it woke up?)
Apr 22 20:37:33 davcos kernel: [30193.408129] CPU 1 is now offline
```

E, come vi dicevo, i Bug in Launchpad possono essere collegati a Bug in altri sistemi di tracciatura in modo da avere un collegamento, ed una sincronizzazione, con Upstream e le altre distribuzioni.

Qui vedete un esempio di un bug del Kernel collegato con Debian e con il gruppo di lavoro del Kernel stesso.



Comunque sia, capiterà di trovarvi nel dubbio di cosa fare. Fare il Triage dei BUG è un tema molto vasto e non si può sapere tutto di tutto.

Però il Codice di Condotta ci dice che “Quando non siete sicuri, chiedete”.

Non c'è niente di male a chiedere aiuto o un parere agli altri.



Bugsquad MailingList

Per coloro che svolgono il Triage c'è la mailing list della Bugsquad:

<https://lists.ubuntu.com/mailman/listinfo/ubuntu-bugsquad>



E, per risposte rapide, un canale IRC sulla rete
freenode.net:

#ubuntu-bugs



Vediamo adesso come funziona il packaging dei programmi in Ubuntu e come possiamo correggere un bug applicando una patch al pacchetto individuato durante il processo di Triage.

Perchè il packaging?



A differenza di altri sistemi operativi, in Ubuntu è possibile installare software in maniera semplice prelevandolo da un archivio centrale gestito dagli Ubuntu Developer.

Questo consente una grande quantità di vantaggi:

- facilità di installazione
- upgrade facile
- riduzione del rischio di infettare il sistema con virus, malware e altro
- gestione automatica delle dipendenze
- rimozione semplice senza lasciare tracce in giro.

Cosa c'è in
un pacchetto?



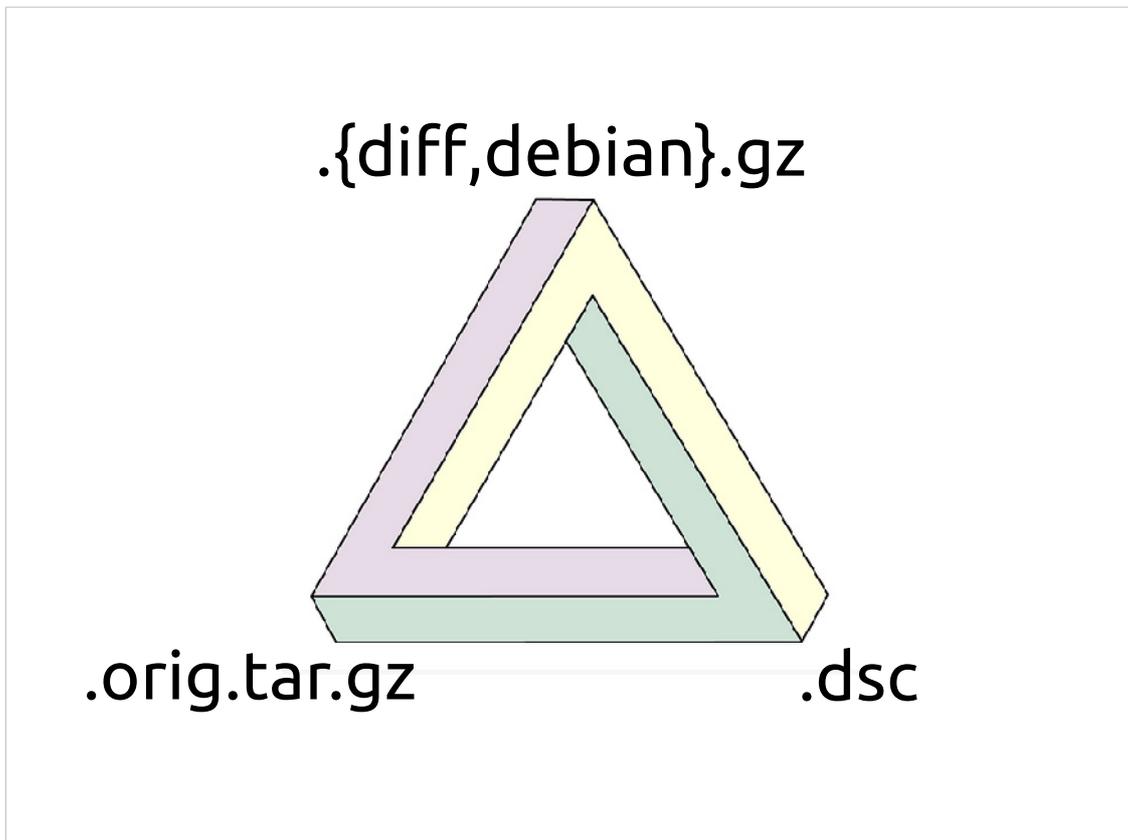
Un pacchetto binario, con estensione `.deb`, contiene i binari compilati e gli altri file del programma in una gerarchia di directory analoga a quella del sistema su cui il programma sarà installato. `Dpkg` non fa altro che estrarre i file da lì e copiarli nel sistema nella posizione giusta.

Inoltre, ogni pacchetto contiene una serie di informazioni, note come meta-dati, necessarie per gestire le dipendenze, le versioni, gli upgrade, etc.



Il source package,
questo sconosciuto

Il lavoro del maintainer di pacchetti si concentra in realtà nel 99% dei casi sul source package, che è il pacchetto contenente il software del programma e le modifiche che il packager ha apportato per consentire la pacchettizzazione stessa. Pertanto, è indispensabile capire com'è fatto un source package per poterci lavorare.



Un pacchetto sorgente è in realtà composto da 3 file distinti:

- un file `.orig.tar.gz`, che è la tarball del software originale (tipicamente senza alcuna modifica)
- un file `.dsc` che contiene metadati necessari alla corretta gestione del source package, inclusi i checksum sha1 e sha256 di tutti e tre i file e la firma GPG dell'ultima persona che ha lavorato sul pacchetto
- un file `.diff.gz` (o `.debian.gz` nel formato più recente dei source package, il 3.0) che è una patch applicata al file `.orig.tar.gz` contenente tutte le modifiche necessarie per il packaging. Tipicamente queste modifiche sono tutte raccolte in una serie di file inseriti all'interno di una cartella di nome `debian/` che è inserita nel source tree del codice sorgente.



All'interno del file diff ci sono quindi i file necessari al packaging dell'applicazione, inseriti all'interno della cartella debian/. Il numero di questi file dipende dalla complessità del packaging stesso.

Tuttavia, 4 di questi file sono obbligatori e devono essere sempre presenti. Vediamoli rapidamente.

```
control (/tmp/unity-3.8.8/debian) - gedit
1 Source: unity
2 Section: gnome
3 Priority: optional
4 Maintainer: Ubuntu Core Developers <ubuntu-devel-discuss@lists.ubuntu.com>
5 Build-Depends: cmake, debhelper (>= 7.0.56~), dh-translations (>= 94), pkg-config, quilt, valac (>= 0.7.8), intltool (>=
0.35.0), libgee-dev (>= 0.5.0), libunique-dev, libbamf-dev (>= 0.2.76), gsettings-desktop-schemas-dev, libgconf2-dev,
libglib2.0-dev (>= 2.22.2-0ubuntu1wncksync3), libdbusmenu-glib-dev (>= 0.3.91), libgnome-desktop-dev, libgtk2.0-dev (>=
2.18.0), libpangol.0 dev, libgnomecui dev, libdee dev (>= 0.5.4), libindicator dev (>= 0.3.19), libunity-misc dev (>= 0.2.1),
libutouch-grail-dev (>= 1.0.20), libxcb-iccml-dev, libxkb-0.9-dev (>= 0.9.42), compiz-dev (>= 1:0.9.2.1
+glibmainloop4-0ubuntu4), libcompizconfig-dev (>= 0.9.2.1git101125), xsitproc, libboost1.42-dev, libboost-serialization1.42-
dev, libstartup-notification-dev, liba1k1.0-dev, libutouch-geis-dev (>= 2.0.10)
6 Standards-Version: 3.9.1
7 Homepage: https://launchpad.net/unity
8 Vcs-Bzr: https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
9
10 Package: unity
11 Architecture: any
12 Depends: ${shlibs:Depends}, ${misc:Depends}, unity-common (= ${source:Version}), compiz-core, compiz-core-obiversion-
${coreabiversion}, compiz-plugins-main, libglib2.0-bin, python, nux-tools, unity-asset-pool (>= 0.8.18)
13 Provides: indicator-renderer, netbook-launcher
14 Conflicts: netbook-launcher (<< 1:2.1.18-0ubuntu2)
15 Replaces: netbook-launcher (<< 1:2.1.10-0ubuntu2)
16 Recommends: unity-place-applications, unity-place-files, indicator-appmenu, indicator-application, indicator-sound, indicator-
datetime, indicator-messages, indicator-me, indicator-session, ubuntu-core-launcher, ubuntu-core-launcher
17 Breaks: compiz-core (<< 1:0.9.2.1+glibmainloop3), bamf (<< 0.2.76),
18 Description: Interface designed for efficiency of space and interaction.
19 Unity is a desktop experience that sings. Designed by Canonical and the Ayatana
20 community, Unity is all about the combination of familiarity and the future. We
21 bring together visual design, analysis of user experience testing, modern
22 graphics technologies and a deep understanding of the free software landscape
23 to produce what we hope will be the lightest, most elegant and most delightful
24 way to use your PC.
25
26 Package: unity-common
27 Architecture: all
28 Depends: ${misc:Depends}, python, Replaces: unity (<< 3.2.2-0ubuntu3) Conflicts: unity (<< 3.2.2-0ubuntu3)
29 Description: Common files for the Unity interface.
30 Unity is a desktop experience that sings. Designed by Canonical and the Ayatana
31 community, Unity is all about the combination of familiarity and the future. We
```

Il file “control” è probabilmente il più importante dei 5. Contiene tutte le metainformazioni necessarie al build del pacchetto e alla sua catalogazione in archivio, oltre alla descrizione che sarà mostrata all'utente in synaptic o nell'Ubuntu Software Center. Contiene una stanza per il source package, e una stanza per ogni binary package che viene generato durante il build. Campi tipici includono il nome del pacchetto, l'elenco delle dipendenze di build, il nome del Maintainer, la homepage del software upstream, etc.

```
1 unity (3.8.8-6ubuntu2) natty; urgency=low
2
3 * Cherry pick a lazy activation case not starting the places on demand
4 (LP: #761225)
5
6 -- Didier Roche <didrocks@ubuntu.com> Fri, 15 Apr 2011 12:15:40 +0200
7
8 unity (3.8.8-6ubuntu1) natty; urgency=low
9
10 * New upstream release.
11 - Empty desktop after login (LP: #687660)
12 - compiz crashed with SIGSEGV in nux::Area::InitiateResizeLayout()
13 (LP: #757709)
14 - compiz crashed on initial load of Java Citrix Client/Inkscape in
15 PrivateWindow::processMap() (LP: #750307)
16 - Sometimes dragging down from the panel just stops working (LP: #750155)
17 - Ensure Unity prioritises icon loading from Unity-icon-theme
18 (LP: #750471)
19 - text truncated in Dash. Empty search from Applications or Files lens
20 returns "Your search did not" due to cut off label (LP: #757362)
21 - should list PlacesGroup.cpp in P01FILES.in (LP: #757663)
22 - Launcher intelhide stuck after edge reveal (LP: #757810)
23 - unity-preferences doesn't load translations (LP: #759682)
24 - dynamic quicklists are not working (LP: #720674)
25 - with second monitor, first monitor does not show notifications
26 application/global menu when displaying desktop (LP: #741449)
27 - launcher - The Workspace, File Lens, App Lens and Trash Launcher icons
28 need to be rendered correctly (LP: #745555)
29 - the launcher background gets dimmed after icons dnd (LP: #747304)
30 - Numpad 'Enter' does not work on unity launcher when navigating through
31 keyboard (LP: #766003)
32 - Launcher shouldn't hover when being in the dash (LP: #760770)
33 - Launcher displays key shortcuts while holding Super while the dash is
34 open but the shortcuts are unfunctional (LP: #760728)
35 - Running/active indicators point to space between launcher icons for
36 folded icons (LP: #783057)
37 - user-trash.svg and user-trash-full.svg not lens grayscale versions
38 (LP: #741894)
39 - code-cleanup: unregistered any established ubus-interests (LP: #757588)
```

Il file “changelog” contiene l'elenco delle modifiche che sono apportate al packaging nel corso del tempo. Il numero di versione del pacchetto è registrato in questo file.



Il file “compat” contiene solo un numero, che rappresenta il livello di compatibilità con debhelper, una suite di tool utilizzati per il build del pacchetto. In effetti, questo file non è strettamente obbligatorio, ma è necessario se ci si avvale delle potenzialità offerte da debhelper. La grande maggioranza dei pacchetti che compongono l'archivio utilizza effettivamente debhelper



```
1 This package was debianized by Didier Roche <didrocks@ubuntu.com> on
2 Tue, 09 Nov 2010 13:19:35 +0200.
3
4 Upstream Authors:
5
6 Neil Jagdish Patel <neil.patel@canonical.com>
7 Gordon Allott <gord.allott@canonical.com>
8 Mirco "MacSlow" Müller <mirco.mueller@canonical.com>
9 Jay Taoko <jay.taoko@canonical.com>
10 Jason Smith <jason.smith@canonical.com>
11 Sam Spillsbury <smspillar@gmail.com>
12
13 Copyright:
14
15 Copyright (C) 2009-2010 Canonical Ltd
16
17 License:
18
19 * (if not otherwise specified):
20
21 This program is free software: you can redistribute it and/or modify
22 it under the terms of the GNU General Public License version 3 as
23 published by the Free Software Foundation.
24
25 This program is distributed in the hope that it will be useful
26 but WITHOUT ANY WARRANTY; without even the implied warranty of
27 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
28 GNU General Public License for more details.
29
30 On Debian/Ubuntu systems, the full text of the GPL v3 can be found in
31 '/usr/share/common-licenses/GPL-3'
32
33 unity/*:
34
35 This library is free software; you can redistribute it and/or
36 modify it under the terms of the GNU Lesser General Public
37 License Version 3.0 as published by the Free Software Foundation.
38
39 This library is distributed in the hope that it will be useful,
```

Il file “copyright” è un file particolarmente delicato che deve essere compilato con grande cura. Contiene tutte le informazioni legate al copyright e alla licenza del software originale.

Un “copyright” compilato in maniera errata può impedire al pacchetto di entrare in archivio (o causarne la eliminazione) per motivi legali.

```
1#!/usr/bin/make -f
2#-*- makefile -*
3
4# Uncomment this to turn on verbose mode.
5#export DH_VERBOSE=1
6
7DEB_VERSION := $(shell dpkg-parsechangelog | egrep '^Version:' | cut -f 2 -d ' ')
8
9CFLAGS=$(shell echo $CFLAGS | sed -e 's/\-Wall//')
10
11CORE_ABIVERSION := $(shell sed -rn 's/^\#define[[:space:]]+CORE_ABIVERSION[[:space:]]+//p' /usr/include/compiz/core/core.h )
12
13override dh_auto_configure:
14    dh_auto_configure -- DCOMPIZ_BUILD_WITH_RPATH=FALSE -DCOMPIZ_PACKAGING_ENABLED=TRUE -
15    DCOMPIZ_PLUGIN_INSTALL_TYPE=package
16
17override dh_install:
18    find debian/tmp/usr/lib -name \*.a -exec rm {} \;
19    dh_install --fail-missing
20
21override dh_gencontrol:
22    dh_gencontrol -- Vcoreabiversion=$(CORE_ABIVERSION)
23    # override netbook-launcher on 1386 to get an epoch
24    rm -f debian/netbook-launcher/DEBIAN/control
25    dh_gencontrol -pnetbook-launcher -- -Vcoreabiversion=$(CORE_ABIVERSION) y2:$(DEB_V)
26
27override dh_auto_test:
28    echo "not working right now"
29%:
30    dh s@ --with translations --with quilt
```

Infine, il file “rules” è un banale Makefile che contiene le istruzioni per il build del pacchetto. Può essere estremamente breve o molto lungo in base alla complessità del pacchetto stesso.



E ora diamoci da fare!

A questo possiamo metterci al lavoro e risolvere il nostro primo bug!



Scegliere il bug giusto

Non tutti i bug sono creati uguali. Scegliere il bug giusto è una decisione che combina gusti personali, conoscenze tecniche, difficoltà e altro ancora.


Ubuntu
warp10 [Log Out](#)

[Overview](#)
[Code](#)
[Bugs](#)
[Blueprints](#)
[Translations](#)
[Answers](#)

System freezes using Unity on IBM Thinkpad Z60m 🔒

Ubuntu » "unity" package » Bug » **Bug #743861**
 Reported by [Cristian Mammoli](#) on 2011-03-27

This bug affects 4 people. [Does this bug affect you?](#) 👍👎👏

Affects	Status	Importance	Assigned to	Milestone
unity	Incomplete	Critical	Unassigned	
linux (Ubuntu)	New	Undecided	Unassigned	Target to milestone
unity (Ubuntu)	Incomplete	Critical	Unassigned	Target to milestone

[Also affects project](#)
[Also affects distribution](#)
[Nominate for series](#)

Bug Description

Binary package hint: unity

Description: Ubuntu Natty (development branch)
Release: 11.04

Unity:
 Installed: 3.6.8-0ubuntu3
 Candidate: 3.6.8-0ubuntu3
 Version table:
 *** 3.6.8-0ubuntu3 0
 500 <http://archive.ubuntu.com/ubuntu/natty/main> 1386 Packages
 100 /var/lib/dpkg/status

After a random time (30 minutes - 1 hour) the system hard freezes:
 can't move mouse, can't switch to tty, can't send magic sysrq, can't ping the computer from network

This only happens when using unity, with a "ubuntu classic" session with compiz everything runs fine for many hours

ProblemType: Bug
 DistroRelease: Ubuntu 11.04
 Package: unity 3.6.8-0ubuntu3
 ProcVersionSignature: Ubuntu 2.6.36-7.36-generic 2.6.38

This report is public

[Mark as duplicate](#)
[Convert to a question](#)
[Link a related branch](#)
[Link to CVE](#)

[Subscribe](#)
[Subscribe someone else](#)

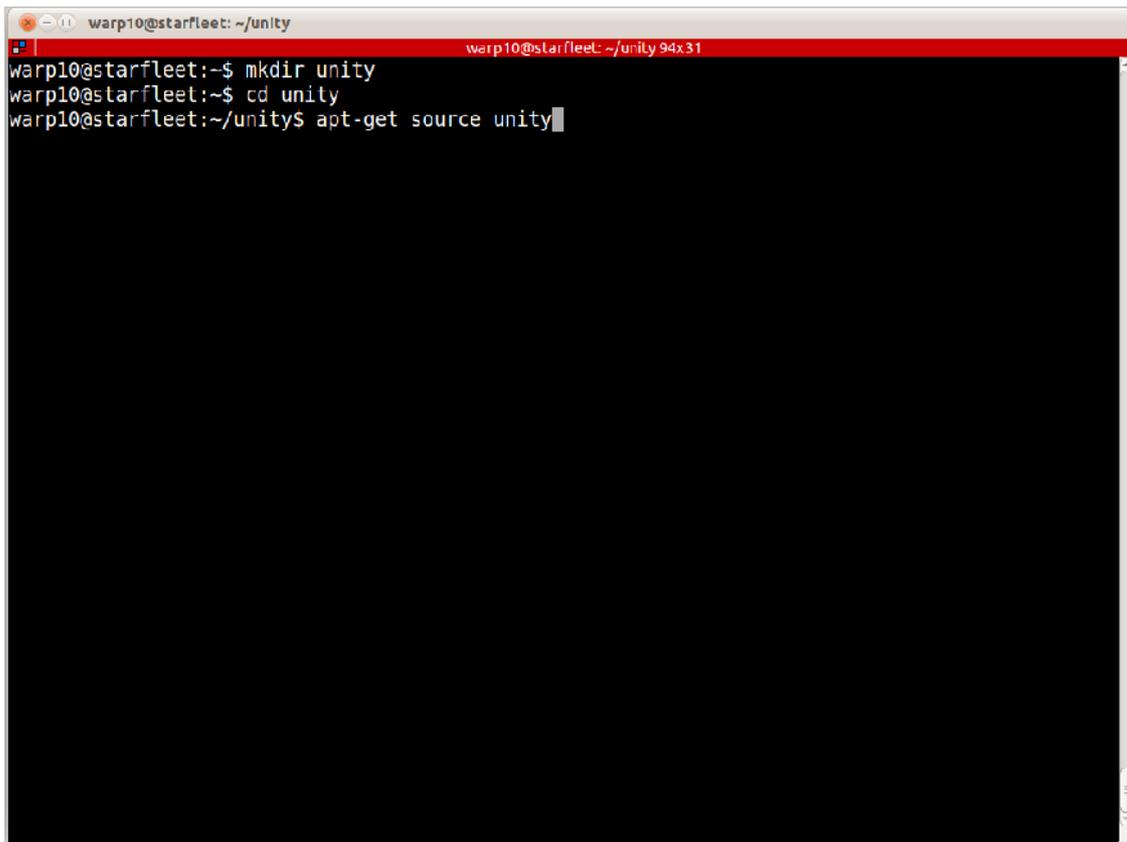
Subscribers

- [Cristian Mammoli](#)
- [Samantha April Davis](#)

Also notified

- [Adriano Pangione](#)
- [Akshey](#)
- [Alejandro J. Alvarez S](#)
- [Ain Andrei](#)
- [Andy Hornq](#)
- [Ashani Holland](#)
- [Bilal Akhtar](#)
- [Bishwendu Kundu](#)
- [Bruno Garcia](#)
- [CRC](#)
- [Carles](#)
- [Charlie_Smoltherman \(porthose\)](#)
- [Chen Li](#)
- [Christian Rücer](#)
- [Cold](#)
- [Conrad J. Sabatier](#)
- [Constantinos Koniaris](#)
- [David Calle](#)
- [Dawning](#)
- [Debian PTS](#)

Per questa dimostrazione scegliamo un bug particolarmente grave che coinvolge Unity, l'interfaccia grafica di Ubuntu

A terminal window titled 'warp10@starfleet: ~/unity' with a red title bar. The terminal shows the following commands and their outputs:

```
warp10@starfleet:~$ mkdir unity
warp10@starfleet:~$ cd unity
warp10@starfleet:~/unity$ apt-get source unity
```

Prendiamo il terminale, creiamo una cartella in cui lavorare e scarichiamo il source package di unity con apt-get

```
warp10@starfleet: ~/unity
warp10@starfleet:~$ mkdir unity
warp10@starfleet:~/unity$ cd unity
warp10@starfleet:~/unity$ apt-get source unity
Reading package lists... Done
Building dependency tree
Reading state information... Done
NOTICE: 'unity' packaging is maintained in the 'Bzr' version control system at:
https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
Please use:
bzip2 -d unity_3.8.8-0ubuntu2.dsc
bzr get https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
to retrieve the latest (possibly unreleased) updates to the package.
Need to get 550 kB of source archives.
Get:1 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (dsc) [1,883 B]
Get:2 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (tar) [517 kB]
Get:3 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (diff) [30.8 kB]
Fetched 550 kB in 2s (210 kB/s)
gpgv: Signature made Fri 15 Apr 2011 12:18:46 PM CEST using DSA key ID E4AC208E
gpgv: Can't check signature: public key not found
dpkg-source: warning: failed to verify signature on ./unity_3.8.8-0ubuntu2.dsc
dpkg-source: info: extracting unity in unity-3.8.8
dpkg-source: info: unpacking unity_3.8.8.orig.tar.gz
dpkg-source: info: applying unity_3.8.8-0ubuntu2.diff.gz
dpkg-source: info: upstream files that have been modified:
 unity-3.8.8/src/PlacesView.cpp
warp10@starfleet:~/unity$
```

Dopo breve tempo, apt-get scaricherà i sorgenti e preparerà un source tree su cui lavorare. Vediamo in dettaglio che cos'ha fatto.

```
warp10@starfleet: ~/unity
warp10@starfleet:~$ mkdir unity
warp10@starfleet:~/unity$ cd unity
warp10@starfleet:~/unity$ apt-get source unity
Reading package lists... Done
Building dependency tree
Reading state information... Done
NOTICE: 'unity' packaging is maintained in the 'Bzr' version control system at:
https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
Please use:
bzip2 -d unity_3.8.8-0ubuntu2.dsc
bzr get https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
to retrieve the latest (possibly unreleased) updates to the package.
Need to get 550 kB of source archives.
Get:1 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (dsc) [1,883 B]
Get:2 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (tar) [517 kB]
Get:3 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (diff) [30.8 kB]
Fetched 550 kB in 2s (210 kB/s)
gpgv: Signature made Fri 15 Apr 2011 12:18:46 PM CEST using DSA key ID E4AC20BE
gpgv: Can't check signature: public key not found
dpkg-source: warning: failed to verify signature on ./unity_3.8.8-0ubuntu2.dsc
dpkg-source: info: extracting unity in unity-3.8.8
dpkg-source: info: unpacking unity_3.8.8.orig.tar.gz
dpkg-source: info: applying unity_3.8.8-0ubuntu2.diff.gz
dpkg-source: info: upstream files that have been modified:
unity-3.8.8/src/PlacesView.cpp
warp10@starfleet:~/unity$
```

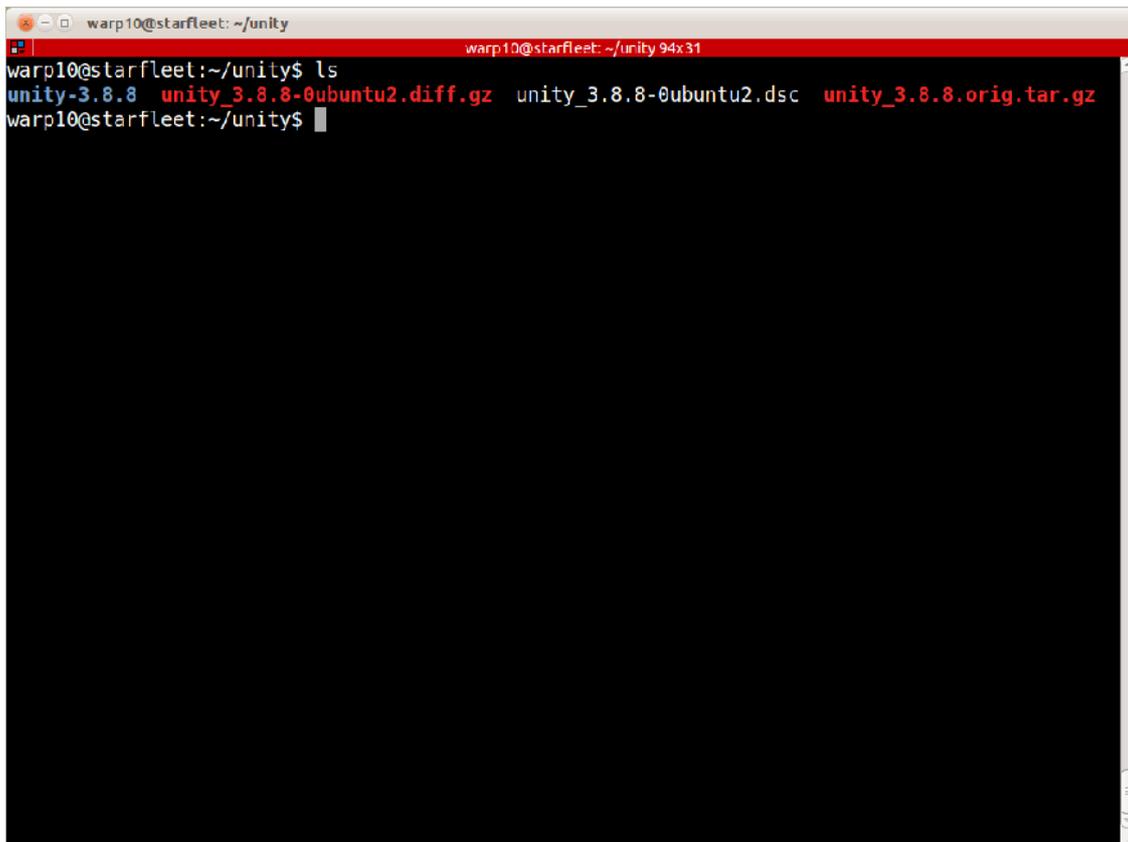
Prima di tutto, apt-get ci avvisa che il packaging di unity è gestito anche tramite bzr, un sistema di controllo delle versioni che sarà sempre più usato in Ubuntu perchè consente una gestione migliore del codice.

```
warp10@starfleet: ~/unity
warp10@starfleet:~$ mkdir unity
warp10@starfleet:~/unity$ cd unity
warp10@starfleet:~/unity$ apt-get source unity
Reading package lists... Done
Building dependency tree
Reading state information... Done
NOTICE: 'unity' packaging is maintained in the 'Bzr' version control system at:
https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
Please use:
bzip2 -d unity_3.8.8-0ubuntu2.tar.gz
to retrieve the latest (possibly unreleased) updates to the package.
Need to get 550 kB of source archives.
Get:1 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (dsc) [1,883 B]
Get:2 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (tar) [517 kB]
Get:3 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (diff) [30.8 kB]
Fetched 550 kB in 2s (210 kB/s)
gpgv: Signature made Fri 15 Apr 2011 12:18:46 PM CEST using DSA key ID E4AC208E
gpgv: Can't check signature: public key not found
dpkg-source: warning: failed to verify signature on ./unity_3.8.8-0ubuntu2.dsc
dpkg-source: info: extracting unity in unity-3.8.8
dpkg-source: info: unpacking unity_3.8.8.orig.tar.gz
dpkg-source: info: applying unity_3.8.8-0ubuntu2.diff.gz
dpkg-source: info: upstream files that have been modified:
unity-3.8.8/src/PlacesView.cpp
warp10@starfleet:~/unity$
```

Successivamente, apt-get scarica i 3 file che compongono il source package.

```
warp10@starfleet: ~/unity
warp10@starfleet:~$ mkdir unity
warp10@starfleet:~/unity$ cd unity
warp10@starfleet:~/unity$ apt-get source unity
Reading package lists... Done
Building dependency tree
Reading state information... Done
NOTICE: 'unity' packaging is maintained in the 'Bzr' version control system at:
https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
Please use:
bzip2 -d unity_3.8.8-0ubuntu2.dsc
bzr get https://code.launchpad.net/~ubuntu-desktop/unity/ubuntu
to retrieve the latest (possibly unreleased) updates to the package.
Need to get 550 kB of source archives.
Get:1 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (dsc) [1,883 B]
Get:2 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (tar) [517 kB]
Get:3 http://archive.ubuntu.com/ubuntu/ natty/main unity 3.8.8-0ubuntu2 (diff) [30.8 kB]
Fetched 550 kB in 2s (210 kB/s)
gpgv: Signature made Fri 15 Apr 2011 12:18:46 PM CEST using DSA key ID E4AC20BE
gpgv: Can't check signature: public key not found
dpkg-source: warning: failed to verify signature on ./unity_3.8.8-0ubuntu2.dsc
dpkg-source: info: extracting unity in unity-3.8.8
dpkg-source: info: unpacking unity_3.8.8.orig.tar.gz
dpkg-source: info: applying unity_3.8.8-0ubuntu2.diff.gz
dpkg-source: info: upstream files that have been modified:
unity-3.8.8/src/PlacesView.cpp
warp10@starfleet:~/unity$
```

Infine, apt-get decomprime il file .orig.tar.gz e applica la patch contenuta nel file .diff.gz



```
warp10@starfleet: ~/unity
warp10@starfleet: ~/unity 94x31
warp10@starfleet:~/unity$ ls
unity-3.8.8  unity_3.8.8-0ubuntu2.diff.gz  unity_3.8.8-0ubuntu2.dsc  unity_3.8.8.orig.tar.gz
warp10@starfleet:~/unity$
```

Alla fine delle operazioni vediamo che apt-get ha salvato nella cartella i 3 file del source package e ha preparato una cartella contenente il source tree

```
warp10@starfleet: ~/unity/unity-3.8.8
warp10@starfleet: ~/unity/unity-3.8.8 94x31
warp10@starfleet:~/unity$ ls
unity-3.8.8  unity_3.8.8-0ubuntu2.diff.gz  unity_3.8.8-0ubuntu2.dsc  unity_3.8.8.orig.tar.gz
warp10@starfleet:~/unity$ cd unity-3.8.8/
warp10@starfleet:~/unity/unity-3.8.8$ ls
AUTHORS          COPYING          po               tools
ChangeLog        COPYING.LGPL     README          unity-private
cmake            debian          resources       unityshell.xml.in
CMakeLists.txt   doc             services        utouch
com.canonical.Unity.gschema.xml  INSTALL         src             vapi
config.h.cmake   plugin-unityshell.png  tests
warp10@starfleet:~/unity/unity-3.8.8$
```

Entriamo ora nella directory che apt-get ha creato per noi: il contenuto è quello del file .orig.tar.gz (ovvero della tarball originale), con l'aggiunta della cartella debian/ e dei file in essa contenuti, come indicato nel file .diff.gz

```
warp10@starfleet: ~/unity/unity-3.8.8
warp10@starfleet: ~/unity/unity-3.8.8 94x31
warp10@starfleet:~/unity$ ls
unity-3.8.8  unity_3.8.8-0ubuntu2.diff.gz  unity_3.8.8-0ubuntu2.dsc  unity_3.8.8.orig.tar.gz
warp10@starfleet:~/unity$ cd unity-3.8.8/
warp10@starfleet:~/unity/unity-3.8.8$ ls
AUTHORS          COPYING          po              tools
ChangeLog        COPYING.LGPL     README         unity-private
cmake            debian          resources      unityshell.xml.in
CMakeLists.txt   doc             services       utouch
com.canonical.Unity.gschema.xml  INSTALL         src            vapi
config.h.cmake   plugin-unityshell.png  tests
warp10@starfleet:~/unity/unity-3.8.8$ ls debian
changelog  control  netbook-launcher.preinst  source_unity.py  unity.install  watch
compat     copyright  rules                     unity-common.install  unity.preinst
warp10@starfleet:~/unity/unity-3.8.8$
```

La cartella debian/ contiene i 5 file obbligatori di cui abbiamo parlato e altri che sono necessari per questo particolare pacchetto.

E ora che si fa?



A questo punto siamo pronti per cominciare.



Risolvere il bug è la parte più amata ed odiata dell'intero processo: amata perchè risolvere bug è un'attività stimolante e divertente, odiata perchè non sempre le cose vanno come speriamo e magari bisognerà fare diversi tentativi prima di arrivare al risultato desiderato.

In generale, i bug possono riguardare il packaging o il codice sorgente del programma. In entrambi i casi, strategie differenti dovranno essere utilizzate a seconda dei casi. L'esperienza e le conoscenze tecniche sapranno dirimenti.

In realtà, un buon maintainer deve essere anche un buon comunicatore, destreggiandosi tra le richieste e le esigenze (spesso contrapposte) dell'utente, dei colleghi MOTU, dei Debian Developer e dello sviluppatore upstream.

```
changelog.dch (~/.unity/unity-3.8.8/debian) - VIM
changelog.dch (~/.unity/Unity-3.8.8/debian) - VIM 94x31
unity (3.8.8-0ubuntu3) natty; urgency=low

* Just a fake fix to astonish people at Ubuntu Party in Schio (LP: #743861)
* So long, and thanks for all the fish!

-- Andrea Colangelo <warpl0@ubuntu.com> Tue, 19 Apr 2011 11:37:16 +0200

unity (3.8.8-0ubuntu2) natty; urgency=low

* Cherry pick a lazy activation case not starting the places on demand
(LP: #761225)

-- Didier Roche <didrocks@ubuntu.com> Fri, 15 Apr 2011 12:15:40 +0200

unity (3.8.8-0ubuntu1) natty; urgency=low

* New upstream release.
- Empty desktop after login (LP: #687660)
- compiz crashed with SIGSEGV in nux::Area::InitiateResizeLayout()
(LP: #757709)
- compiz crashed on initial load of Java Citrix Client/Inkscape in
PrivateWindow::processMap() (LP: #758307)
- Sometimes dragging down from the panel just stops working (LP: #750155)
- Ensure Unity prioritises icon loading from Unity-icon-theme
(LP: #750471)
- text truncated in Dash. Empty search from Applications or Files lense
returns "Your search did not" due to cut off label (LP: #757362)
- should list PlacesGroup.cpp in POTFILES.in (LP: #757663)
- Launcher intelhide stuck after edge reveal (LP: #757810)
- unity-preferences doesn't load translations (LP: #759682)

"debian/changelog.dch" 2051L, 89809C written
7,0-1 Top
```

Dopo aver trovato il problema e fatto il nostro fix, scriviamo una breve nota nel file changelog per documentare il lavoro fatto. In questo caso specifico, trascuriamo volutamente la parte relativa al fix in sé, che fingiamo sia data per scontata, e ci concentriamo in maniera specifica sulle procedure legate al packaging.

```
changelog.dch (~/.unity/unity-3.8.8/debian) - VIM
changelog.dch (~/.unity/Unity-3.8.8/debian) - VIM 94x31
unity (3.8.8-0ubuntu3) natty; urgency=low
* Just a fake fix to appease people at Ubuntu Party in Schio (LP: #743861)
* So long, and thanks for all the fish!
-- Andrea Colangelo <wasteb@ubuntu.com> Tue, 19 Apr 2011 11:37:16 +0200
unity (3.8.8-0ubuntu2) natty; urgency=low
* Cherry pick a lazy activation case not starting the places on demand
(LP: #761225)
-- Didier Roche <didrocks@ubuntu.com> Fri, 15 Apr 2011 12:15:40 +0200
unity (3.8.8-0ubuntu1) natty; urgency=low
* New upstream release.
- Empty desktop after login (LP: #687660)
- compiz crashed with SIGSEGV in nux::Area::InitiateResizeLayout()
(LP: #757709)
- compiz crashed on initial load of Java Citrix Client/Inkscape in
PrivateWindow::processMap() (LP: #758307)
- Sometimes dragging down from the panel just stops working (LP: #750155)
- Ensure Unity prioritises icon loading from Unity-icon-theme
(LP: #750471)
- text truncated in Dash. Empty search from Applications or Files lense
returns "Your search did not" due to cut off label (LP: #757362)
- should list PlacesGroup.cpp in POTFILES.in (LP: #757663)
- Launcher intelhide stuck after edge reveal (LP: #757810)
- unity-preferences doesn't load translations (LP: #759682)
"debian/changelog.dch" 2051L, 89809C written
7,0-1 Top
```

Il numero di versione viene ovviamente aumentato...

```
changelog.dch (~/.unity/unity-3.8.8/debian) - VIM
changelog.dch (~/.unity/Unity-3.8.8/debian) - VIM 94x31
unity (3.8.8-0ubuntu3) natty; urgency=low

* Just a fake fix to astonish people at Ubuntu Party in Schio (LP: #743861)
* So long, and thanks for all the fish!

-- Andrea Colangelo <warpl0@ubuntu.com> Tue, 19 Apr 2011 11:37:16 +0200

unity (3.8.8-0ubuntu2) natty; urgency=low

* Cherry pick a lazy activation case not starting the places on demand
(LP: #761225)

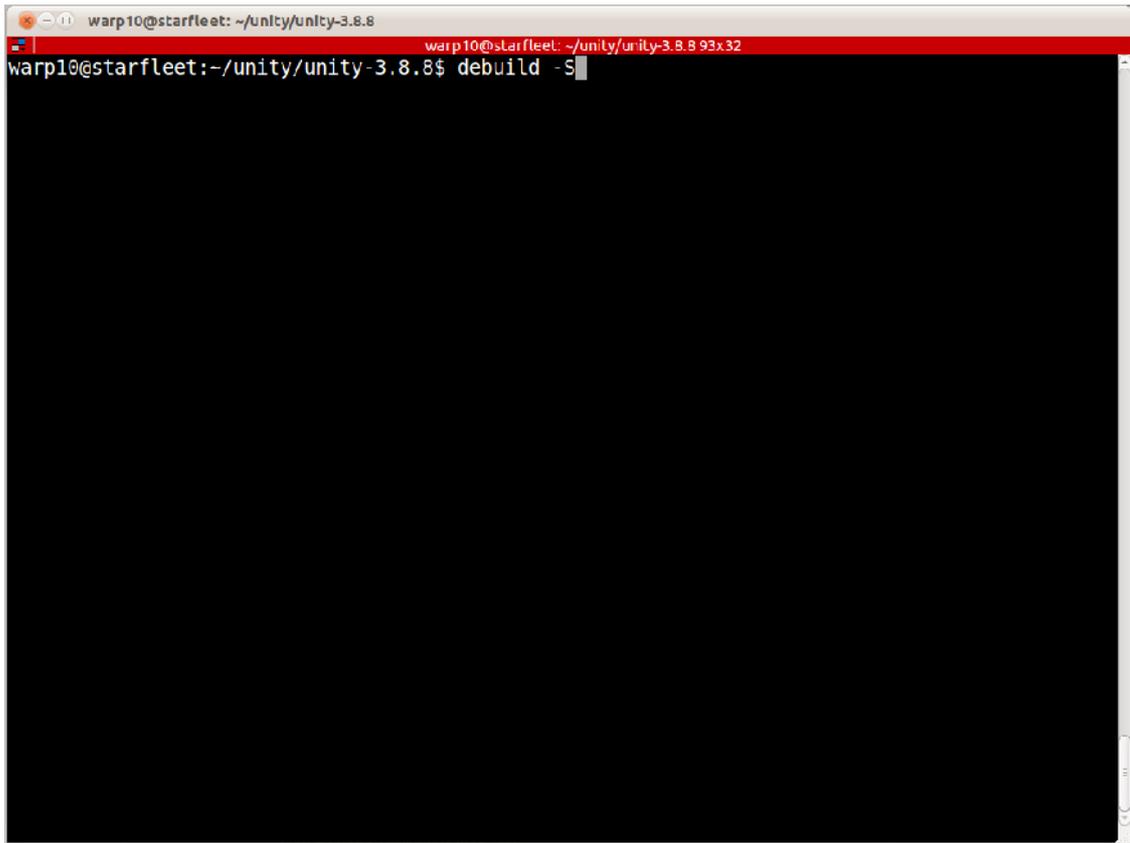
-- Didier Roche <didrocks@ubuntu.com> Fri, 15 Apr 2011 12:15:40 +0200

unity (3.8.8-0ubuntu1) natty; urgency=low

* New upstream release.
- Empty desktop after login (LP: #687660)
- compiz crashed with SIGSEGV in nux::Area::InitiateResizeLayout()
(LP: #757709)
- compiz crashed on initial load of Java Citrix Client/Inkscape in
PrivateWindow::processMap() (LP: #758307)
- Sometimes dragging down from the panel just stops working (LP: #750155)
- Ensure Unity prioritises icon loading from Unity-icon-theme
(LP: #750471)
- text truncated in Dash. Empty search from Applications or Files lense
returns "Your search did not" due to cut off label (LP: #757362)
- should list PlacesGroup.cpp in POTFILES.in (LP: #757663)
- Launcher intelhide stuck after edge reveal (LP: #757810)
- unity-preferences doesn't load translations (LP: #759682)

"debian/changelog.dch" 2051L, 89809C written
7,0-1 Top
```

...e si aggiunge questo tag magico che chiuderà automaticamente il bug quando il pacchetto sarà stato compilato e pubblicato in archivio

A terminal window with a red title bar. The title bar contains the text 'warp10@starfleet: ~/unity/unity-3.8.8' on the left and 'warp10@starfleet: ~/unity/unity-3.8.8 93x32' on the right. The terminal content shows the prompt 'warp10@starfleet:~/unity/unity-3.8.8\$' followed by the command 'debuild -S' and a cursor. The rest of the terminal area is black.

```
warp10@starfleet: ~/unity/unity-3.8.8
warp10@starfleet: ~/unity/unity-3.8.8 93x32
warp10@starfleet:~/unity/unity-3.8.8$ debuild -S
```

A questo punto possiamo creare il source package contenente il nostro fix.

```
warp10@starfleet: ~/unity/unity-3.8.8
warp10@starfleet: ~/unity/Unity-3.8.8 93x37
dpkg-source: info: use the '3.0 (quilt)' format to have separate and documented changes to up
stream files, see dpkg-source(1)
dpkg-source: info: building unity in unity_3.8.8-0ubuntu3.dsc
dpkg-genchanges -S >../unity_3.8.8-0ubuntu3_source.changes
dpkg-genchanges: not including original source code in upload
dpkg-source --after-build unity-3.8.8
dpkg-buildpackage: source only, diff-only upload (original source NOT included)
Now running lintian...
W: unity source: quilt-build-dep-but-no-series-file
W: unity source: patch-system-but-no-source-readme
W: unity source: patch-system-but-direct-changes-in-diff src/PlacesView.cpp
E: unity source: missing-build-dependency quilt (>= 0.46-7-)
Finished running lintian.
Now signing changes and any dsc files...
  signfile unity_3.8.8-0ubuntu3.dsc Andrea Colangelo <warp10@ubuntu.com>

You need a passphrase to unlock the secret key for
user: "Andrea Colangelo <warp10@ubuntu.com>"
2048-bit RSA key, ID CE5C8446, created 2011-04-19

gpg: gpg-agent is not available in this session

  signfile unity_3.8.8-0ubuntu3_source.changes Andrea Colangelo <warp10@ubuntu.com>

You need a passphrase to unlock the secret key for
user: "Andrea Colangelo <warp10@ubuntu.com>"
2048-bit RSA key, ID CE5C8446, created 2011-04-19

Enter passphrase: gpg: gpg-agent is not available in this session

Successfully signed dsc and changes files
warp10@starfleet:~/unity/unity-3.8.8$
```

Dopo pochi secondi, debuild restituirà il source package, debitamente firmato con la chiave GPG dello sviluppatore.

```
warp10@starfleet: ~/unity/unity-3.8.8
warp10@starfleet: ~/unity/unity-3.8.8 93x32
dpkg-source --after-build unity-3.8.8
dpkg-buildpackage: source only, diff-only upload (original source NOT included)
Now running lintian...
W: unity source: quilt-build-dep-but-no-series-file
W: unity source: patch-system-but-no-source-readme
W: unity source: patch-system-but-direct-changes-in-diff src/PlacesView.cpp
E: unity source: missing-build-dependency quilt (>= 0.46-7-)
Finished running lintian.
Now signing changes and any dsc files...
signfile unity_3.8.8-0ubuntu3.dsc Andrea Colangelo <warp10@ubuntu.com>

You need a passphrase to unlock the secret key for
user: "Andrea Colangelo <warp10@ubuntu.com>"
2048-bit RSA key, ID CE5C8446, created 2011-04-19

gpg: gpg-agent is not available in this session

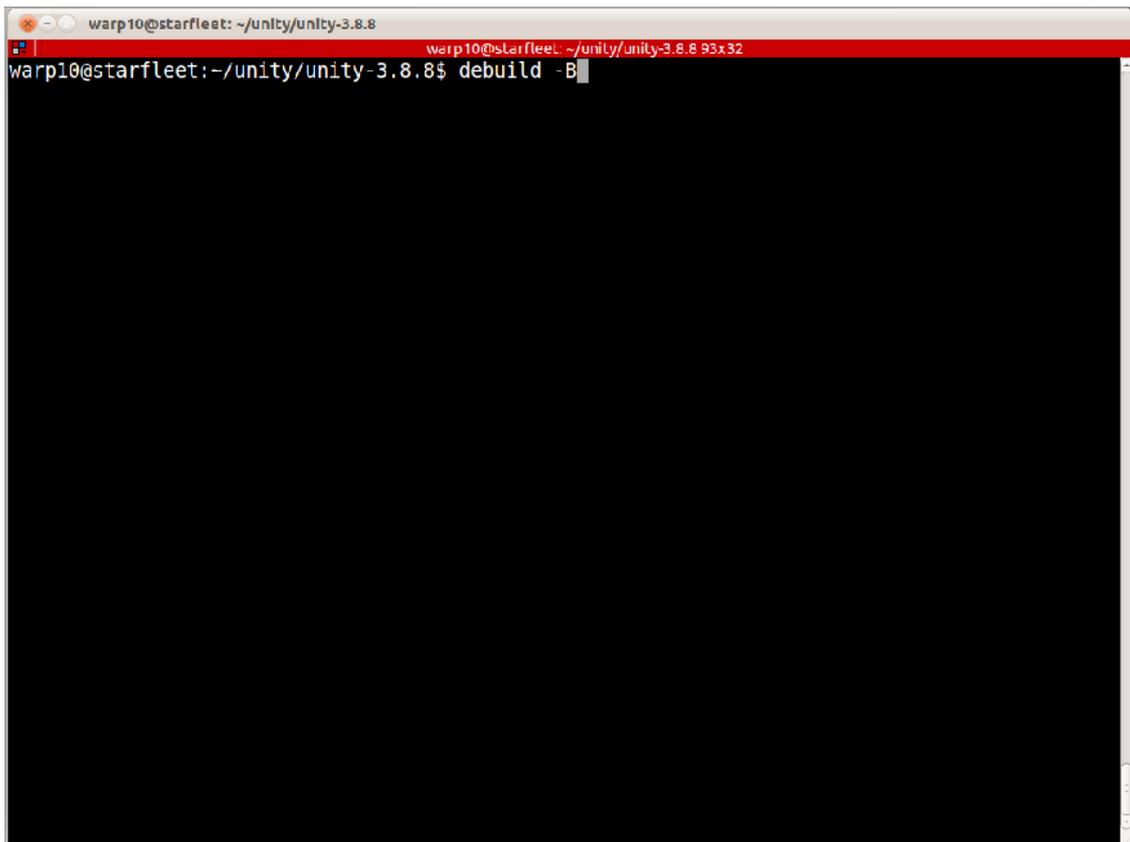
signfile unity_3.8.8-0ubuntu3_source.changes Andrea Colangelo <warp10@ubuntu.com>

You need a passphrase to unlock the secret key for
user: "Andrea Colangelo <warp10@ubuntu.com>"
2048-bit RSA key, ID CE5C8446, created 2011-04-19

gpg: gpg-agent is not available in this session

Successfully signed dsc and changes files
warp10@starfleet:~/unity/unity-3.8.8$ ls ..
unity-3.8.8      unity_3.8.8-0ubuntu3.dsc
unity_3.8.8-0ubuntu2.diff.gz  unity_3.8.8-0ubuntu3_source.build
unity_3.8.8-0ubuntu2.dsc      unity_3.8.8-0ubuntu3_source.changes
unity_3.8.8-0ubuntu3.diff.gz  unity_3.8.8.orig.tar.gz
warp10@starfleet:~/unity/unity-3.8.8$
```

Ora nella cartella esterna abbiamo un file .dsc e un file .diff.gz del nostro nuovo source package. Poichè il file .orig.tar.gz non è stato modificato, non c'è bisogno di ricrearlo, ed è quello che apt-get aveva scaricato per noi.

A terminal window with a red title bar. The title bar contains the text 'warp10@starfleet: ~/unity/unity-3.8.8' on the left and 'warp10@starfleet: ~/unity/unity-3.8.8 93x32' on the right. The terminal content shows the prompt 'warp10@starfleet:~/unity/unity-3.8.8\$' followed by the command 'debuild -B' and a cursor. The rest of the terminal area is black.

```
warp10@starfleet: ~/unity/unity-3.8.8
warp10@starfleet: ~/unity/unity-3.8.8 93x32
warp10@starfleet:~/unity/unity-3.8.8$ debuild -B
```

A questo punto usiamo di nuovo `debuild` per creare il pacchetto binario.

In realtà, un buon maintainer preferirebbe usare un programma come `pbuilder` che consente di creare una `chroot` pulita e minimale dentro la quale compilare il pacchetto, per una serie di ragioni tecniche. Per ora, ignoriamo questa buona pratica.

```
warp10@starfleet: ~/unity
warp10@starfleet:~/unity/unity-3.8.8$ cd ..
warp10@starfleet:~/unity$ ls
unity-3.8.8                                unity_3.8.8-0ubuntu3.diff.gz
unity_3.8.8-0ubuntu2.diff.gz              unity_3.8.8-0ubuntu3.dsc
unity_3.8.8-0ubuntu2.dsc                  unity_3.8.8-0ubuntu3_source.build
unity_3.8.8-0ubuntu3_amd64.build           unity_3.8.8-0ubuntu3_source.changes
unity_3.8.8-0ubuntu3_amd64.changes        unity_3.8.8.orig.tar.gz
unity_3.8.8-0ubuntu3_amd64.deb
warp10@starfleet:~/unity$
```



Dopo qualche minuto (o perfino qualche ora!) debuild completa la sua esecuzione ed ecco che un file .deb appare nella cartella esterna. Ora possiamo installare questo file sul sistema e assicurarci che il nostro fix risolva il problema (senza introdurne di nuovi!)

```
warp10@starfleet: ~/unity
warp10@starfleet: ~/unity/unity-3.8.8$ cd ..
warp10@starfleet: ~/unity$ ls
unity-3.8.8
unity_3.8.8-0ubuntu2.diff.gz      unity_3.8.8-0ubuntu3.diff.gz
unity_3.8.8-0ubuntu2.dsc         unity_3.8.8-0ubuntu3.dsc
unity_3.8.8-0ubuntu3_amd64.build unity_3.8.8-0ubuntu3_source.build
unity_3.8.8-0ubuntu3_amd64.changes unity_3.8.8-0ubuntu3_source.changes
unity_3.8.8-0ubuntu3_amd64.deb  unity_3.8.8.orig.tar.gz
warp10@starfleet: ~/unity$ dput ubuntu unity_3.8.8-0ubuntu3_source.changes
```

L'ultimo passaggio consiste nell'utilizzare dput per caricare il source package nell'archivio. Il pacchetto sarà compilato, pubblicato e il bug report di LP sarà chiuso automaticamente.



Missione compiuta!



Chiunque può partecipare allo sviluppo di Ubuntu. Questi tre link sono tre ottimi punti di partenza per approfondire le tematiche tecniche o per scoprire come cominciare a contribuire nel packaging.



Ubuntu Party 2011
Schio, Palazzo Toaldi Capra

1 Maggio 2011