Debian Med sprints and mentoring

Andreas Tille

Debian

Saint Malo, 31. January 2015
Face to face meetings with users and upstream

Have a quick look at what happened since 2011 when we started doing yearly sprints by also inviting users and upstream developers.

Thanks to Debian for supporting sprints which we will prove to be successful.
Top 10 Uploaders of Debian Med team

Andreas T 1482
Charles P 522
Steffen M 342
Mathieu M 320
Olivier S 207
Steve M. R 131
Thorsten A 109
Laszlo K 98
Aaron M. U 94
Dominique B 44
Top 10 discussion on Debian Med user mailing list

- Andreas T: 4548
- Charles P: 1323
- Karsten H: 607
- Mathieu M: 446
- Steffen M: 430
- Olivier S: 391
- David P: 315
- Eric M: 306
- Yaroslav H: 259
- Sebastian H: 210

Years: 2002-2015

- 2002: 0
- 2003: 200
- 2004: 400
- 2005: 600
- 2006: 800
- 2007: 0
- 2008: 200
- 2009: 400
- 2010: 600
- 2011: 800
- 2012: 0
- 2013: 200
- 2014: 400
- 2015: 600

Debian Med sprints and mentoring
Saint Malo, 31. January 2015
Andreas Tille (Debian)
Debian Med sprints and mentoring
Saint Malo, 31. January 2015

Top 10 discussion Debian Med developer mailing list

2006 2007 2008 2009 2010 2011 2012 2013 2014
0 100 200 300 400 500 600
Andreas T 1948
Charles P 856
Mathieu M 408
Olivier S 180
Laszlo K 158
Alexandre R 152
Michael H 145
Shaun J 141
Steffen M 128
Matthias K 127
Top 10 bug hunters of Debian Med packages

Andreas Tille (Debian)
Debian Med sprints and mentoring
Saint Malo, 31. January 2015
Top 10 committers to Debian Med VCS

- Andreas T 8407
- Charles P 3901
- Mathieu M 2063
- Olivier S 1251
- Steffen M 984
- Laszlo K 594
- David P 536
- Steven M. R 516
- Aaron M. U 388
- Thorsten A 306
Maintainer per package relation in Debian Med
Number of Packages in selected tasks

Andreas Tille (Debian)
Debian Med sprints and mentoring
Saint Malo, 31. January 2015
In several talks at DebConf and in discussions before I have heard the argument that it is hard to find friends. But it is not. The answer is to create inviting teams.
Waking up in the morning and realising that somebody else has solved your problem from yesterday
Mentoring of the Month (MoM)

- mentor dedicates a part of his spare time to a newcomer
- guide the student kindly into all secrets of Debian packaging
- learning per example
- student is free to pick the package (mentor has a vetoing right)
Mentoring of the Month (MoM)

- mentor dedicates a part of his spare time to a newcomer
- guide the student kindly into all secrets of Debian packaging
- learning per example
- student is free to pick the package (mentor has a vetoing right)
mentor dedicates a part of his spare time to a newcomer
guide the student kindly into all secrets of Debian packaging
learning per example
student is free to pick the package (mentor has a vetoing right)
Mentoring of the Month (MoM)

- Mentor dedicates a part of his spare time to a newcomer.
- Guide the student kindly into all secrets of Debian packaging.
- Learning per example.
- Student is free to pick the package (mentor has a vetoing right).
MoM Goals

- The package in question is finished and uploaded.
- The student is able to do advanced packaging tasks and is introduced into the communication channels of Debian Med team.
MoM Goals

- The package in question is finished and uploaded.
- The student is able to do advanced packaging tasks and is introduced into the communication channels of Debian Med team.
 Summary of MoM formalism

- MoM is *work*
- hopefully it shows that this work is also *fun*
- mentor trades his spare time for the work of the student
- attempt to train "silent observers on the list"
- focus not only on technical details but also on learning to know relevant communication channels
Summary of MoM formalism

- MoM is *work*
- hopefully it shows that this work is also *fun*
- mentor trades his spare time for the work of the student
- attempt to train "silent observers on the list"
- focus not only on technical details but also on learning to know relevant communication channels
MoM is *work*

- hopefully it shows that this work is also *fun*
- mentor trades his spare time for the work of the student
- attempt to train "silent observers on the list"
- focus not only on technical details but also on learning to know relevant communication channels
Summary of MoM formalism

- MoM is *work*
- hopefully it shows that this work is also *fun*
- mentor trades his spare time for the work of the student
- attempt to train "silent observers on the list"
- focus not only on technical details but also on learning to know relevant communication channels
MoM is *work*

- hopefully it shows that this work is also *fun*
- mentor trades his spare time for the work of the student
- attempt to train "silent observers on the list"
- focus not only on technical details but also on learning to know relevant communication channels
<table>
<thead>
<tr>
<th>Month</th>
<th>Student</th>
<th>Package</th>
<th>Uploaded?</th>
</tr>
</thead>
<tbody>
<tr>
<td>02.2012</td>
<td>Luis Ibanez</td>
<td>fis-gtm</td>
<td>yes</td>
</tr>
<tr>
<td>03.2012</td>
<td>Scott Christley</td>
<td>libswarm</td>
<td>no</td>
</tr>
<tr>
<td>06.2012</td>
<td>Eric Maeker</td>
<td>libquazip</td>
<td>yes</td>
</tr>
<tr>
<td>11.2012</td>
<td>Tomás Di Domenico</td>
<td>python-csb</td>
<td>yes</td>
</tr>
<tr>
<td>02.2013</td>
<td>Sukhbir Singh</td>
<td>hunspell-en-med</td>
<td>yes</td>
</tr>
<tr>
<td>12.2013</td>
<td>Lennart C. Karssen</td>
<td>probabel</td>
<td>yes</td>
</tr>
<tr>
<td>01.2014</td>
<td>Jorge Sebastião Soares</td>
<td>snp-sites</td>
<td>yes</td>
</tr>
<tr>
<td>03.2014</td>
<td>Stephen Smith</td>
<td>phyutility</td>
<td>yes</td>
</tr>
<tr>
<td>06.2014</td>
<td>Ian Wallace</td>
<td>openemr</td>
<td></td>
</tr>
<tr>
<td>09.2014</td>
<td>Corentin Desfarges</td>
<td>camp</td>
<td>yes</td>
</tr>
<tr>
<td>02.2015</td>
<td>Paul Novotny</td>
<td>opensurgsim</td>
<td></td>
</tr>
</tbody>
</table>

Some people put their names in the waiting queue but did not answer a ping.
Debian Med has attracted one developer per year

According to a *questionnaire* in Wiki

- Debian Med has 25 DDs+DMs
- 12 DDs *because* Debian Med exists;
  4 DDs before Debian Med started
- 8 out of the 12 above extended their activity to other fields in Debian
- 10 out of the 12 above are active in Debian Med
Debian Med has attracted one developer per year

According to a *questionnaire* in Wiki

- Debian Med has 25 DDs+DMs
- 12 DDs *because* Debian Med exists;
  4 DDs before Debian Med started
- 8 out of the 12 above extended their activity to other fields in Debian
- 10 out of the 12 above are active in Debian Med
Debian Med has attracted one developer per year

According to a *questionnaire* in Wiki

- Debian Med has 25 DDs+DMs
- 12 DDs *because* Debian Med exists;
  4 DDs before Debian Med started
- 8 out of the 12 above extended their activity to other fields in Debian
- 10 out of the 12 above are active in Debian Med
Debian Med has attracted one developer per year

According to a *questionnaire* in Wiki

- Debian Med has 25 DDs+DMs
- 12 DDs *because* Debian Med exists;
  - 4 DDs before Debian Med started
- 8 out of the 12 above extended their activity to other fields in Debian
- 10 out of the 12 above are active in Debian Med
MoM Conclusions

- **Time spent into mentoring is worth the effort**
- No student for each month so the workload is bearable
- Students have just read recent documents which I did ten years ago → I can learn new stuff from them
- Major advantage: training upstream to pool their knowledge about the code with ours about packaging is **very efficient** for the hard packages
- About 50% of students had strong connection to upstream and requests for upstream changes went very smoothly
MoM Conclusions

- Time spent into mentoring is worth the effort
- No student for each month so the workload is bearable
- Students have just read recent documents which I did ten years ago → I can learn new stuff from them
- Major advantage: training upstream to pool their knowledge about the code with ours about packaging is very efficient for the hard packages
- About 50% of students had strong connection to upstream and requests for upstream changes went very smoothly
MoM Conclusions

- Time spent into mentoring is worth the effort
- No student for each month so the workload is bearable
- Students have just read recent documents which I did ten years ago → I can learn new stuff from them
- Major advantage: training upstream to pool their knowledge about the code with ours about packaging is very efficient for the hard packages
- About 50% of students had strong connection to upstream and requests for upstream changes went very smoothly
MoM Conclusions

- Time spent into mentoring is worth the effort
- No student for each month so the workload is bearable
- Students have just read recent documents which I did ten years ago → I can learn new stuff from them
- Major advantage: training upstream to pool their knowledge about the code with ours about packaging is very efficient for the hard packages
- About 50% of students had strong connection to upstream and requests for upstream changes went very smoothly
Time spent into mentoring is worth the effort
No student for each month so the workload is bearable
Students have just read recent documents which I did ten years ago → I can learn new stuff from them
Major advantage: training upstream to pool their knowledge about the code with ours about packaging is very efficient for the hard packages
About 50% of students had strong connection to upstream and requests for upstream changes went very smoothly
This talk is available at

http://people.debian.org/~tille/talks/

Andreas Tille <tille@debian.org>