Originally packages were literally *owned* by their maintainer

- At DebConf3 Teams were suggested, since DebConf5 Uploaders
- Both quickly adopted in Debian Med and other Blends
- At DebConf8 I realised: Hard to specify who is in the team
- Solved for DebConf11 with GSoC Teammetrics
- Originally packages were literally *owned* by their maintainer
- At DebConf3 Teams were suggested, since DebConf5 Uploaders
  - Both quickly adopted in Debian Med and other Blends
  - At DebConf8 I realised: Hard to specify who is in the team
  - Solved for DebConf11 with GSoC Teammetrics
Originally packages were literally *owned* by their maintainer. At DebConf3 Teams were suggested, since DebConf5 Uploaders. Both quickly adopted in Debian Med and other Blends. At DebConf8 I realised: Hard to specify who is in the team. Solved for DebConf11 with GSoC Teammetrics.
Originally packages were literally *owned* by their maintainer

At DebConf3 Teams were suggested, since DebConf5 Uploaders

Both quickly adopted in Debian Med and other Blends

At DebConf8 I realised: Hard to specify who is in the team

Solved for DebConf11 with GSoC Teammetrics
Originally packages were literally *owned* by their maintainer
At DebConf3 Teams were suggested, since DebConf5 Uploaders
Both quickly adopted in Debian Med and other Blends
At DebConf8 I realised: Hard to specify who is in the team
Solved for DebConf11 with GSoC Teammetrics
Types of teams

- Content driven teams - frequently Blends
- Infrastructure teams (installer, dpkg, apt)
- Language teams
Types of teams

- Content driven teams - frequently Blends
- Infrastructure teams (installer, dpkg, apt)
- Language teams
Types of teams

- Content driven teams - frequently Blends
- Infrastructure teams (installer, dpkg, apt)
- Language teams
Composition of teams

- Well staffed, more than one member can handle issues
- Multiple members but reaching limits
- Growing teams
- Shrinking teams
- (Basically) one member team
- Single maintainers
Composition of teams

- Well staffed, more than one member can handle issues
- Multiple members but reaching limits
- Growing teams
- Shrinking teams
- (Basically) one member team
- Single maintainers
Composition of teams

- Well staffed, more than one member can handle issues
- Multiple members but reaching limits
- Growing teams
- Shrinking teams
- (Basically) one member team
- Single maintainers
Composition of teams

- Well staffed, more than one member can handle issues
- Multiple members but reaching limits
- Growing teams
- Shrinking teams
  - (Basically) one member team
- Single maintainers
Composition of teams

- Well staffed, more than one member can handle issues
- Multiple members but reaching limits
- Growing teams
- Shrinking teams
- (Basically) one member team
- Single maintainers

Andreas Tille (Debian)
Composition of teams

- Well staffed, more than one member can handle issues
- Multiple members but reaching limits
- Growing teams
- Shrinking teams
- (Basically) one member team
- Single maintainers
Culture in teams

- Every member can and should upload every team package
- Few members care for everything, others only for specific packages
- Uploader cares for specific packages, others will contribute randomly
- Team wide automatic packaging upgrades in Git (Janitor)
Culture in teams

- Every member can and should upload every team package
- Few members care for everything, others only for specific packages
- Uploader cares for specific packages, others will contribute randomly
- Team wide automatic packaging upgrades in Git (Janitor)
Culture in teams

- Every member can and should upload every team package
- Few members care for everything, others only for specific packages
- Uploader cares for specific packages, others will contribute randomly
- Team wide automatic packaging upgrades in Git (Janitor)
Culture in teams

- Every member can and should upload every team package
- Few members care for everything, others only for specific packages
- Uploader cares for specific packages, others will contribute randomly
- Team wide automatic packaging upgrades in Git (Janitor)
Actively inviting and teaching

- Mentoring of the Month (teaching upstream and users)
- Sprints in common with upstream and users
- Participating in GSoC + Outreachy
- Sponsoring of Blends
Actively inviting and teaching

- Mentoring of the Month (teaching upstream and users)
- Sprints in common with upstream and users
  - Participating in GSoC + Outreachy
- Sponsoring of Blends
Actively inviting and teaching

- Mentoring of the Month (teaching upstream and users)
- Sprints in common with upstream and users
- Participating in GSoC + Outreachy
- Sponsoring of Blends
Actively inviting and teaching

- Mentoring of the Month (teaching upstream and users)
- Sprints in common with upstream and users
- Participating in GSoC + Outreachy
- Sponsoring of Blends
In case you want to run your own team stats

There is a download script

```
#!/bin/sh

rsync -a -P \\
blends.debian.net:/var/backups/teammetrics/teammetrics.0.xz .
dropdb teammetrics || true
createdb teammetrics
xzcat teammetrics.0.xz | psql teammetrics
```

The database enables you to do more fine grained investigation of your team