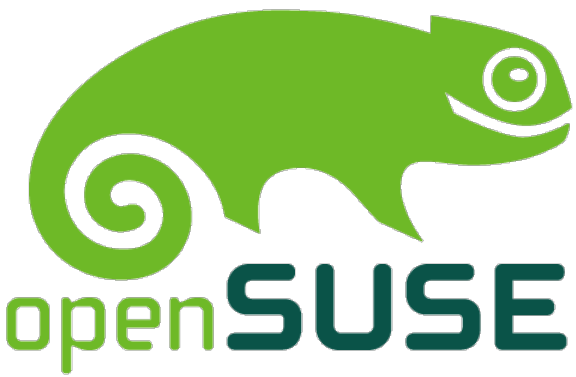
openSUSE in Numbers

Basic numbers in the openSUSE project



Alberto Planas
<aplanas@suse.de>
openSUSE Team

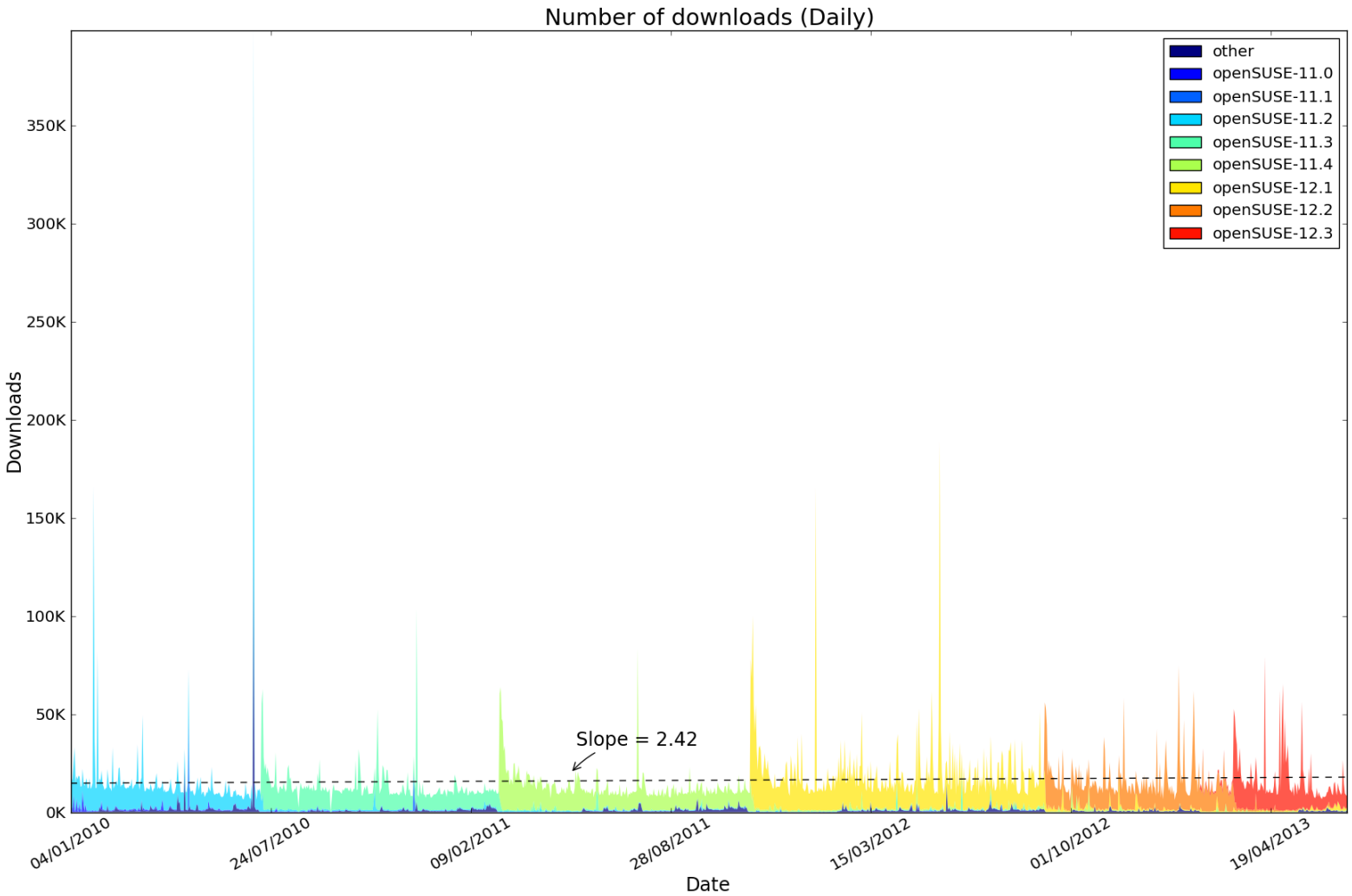
July 17, 2013

Downloads

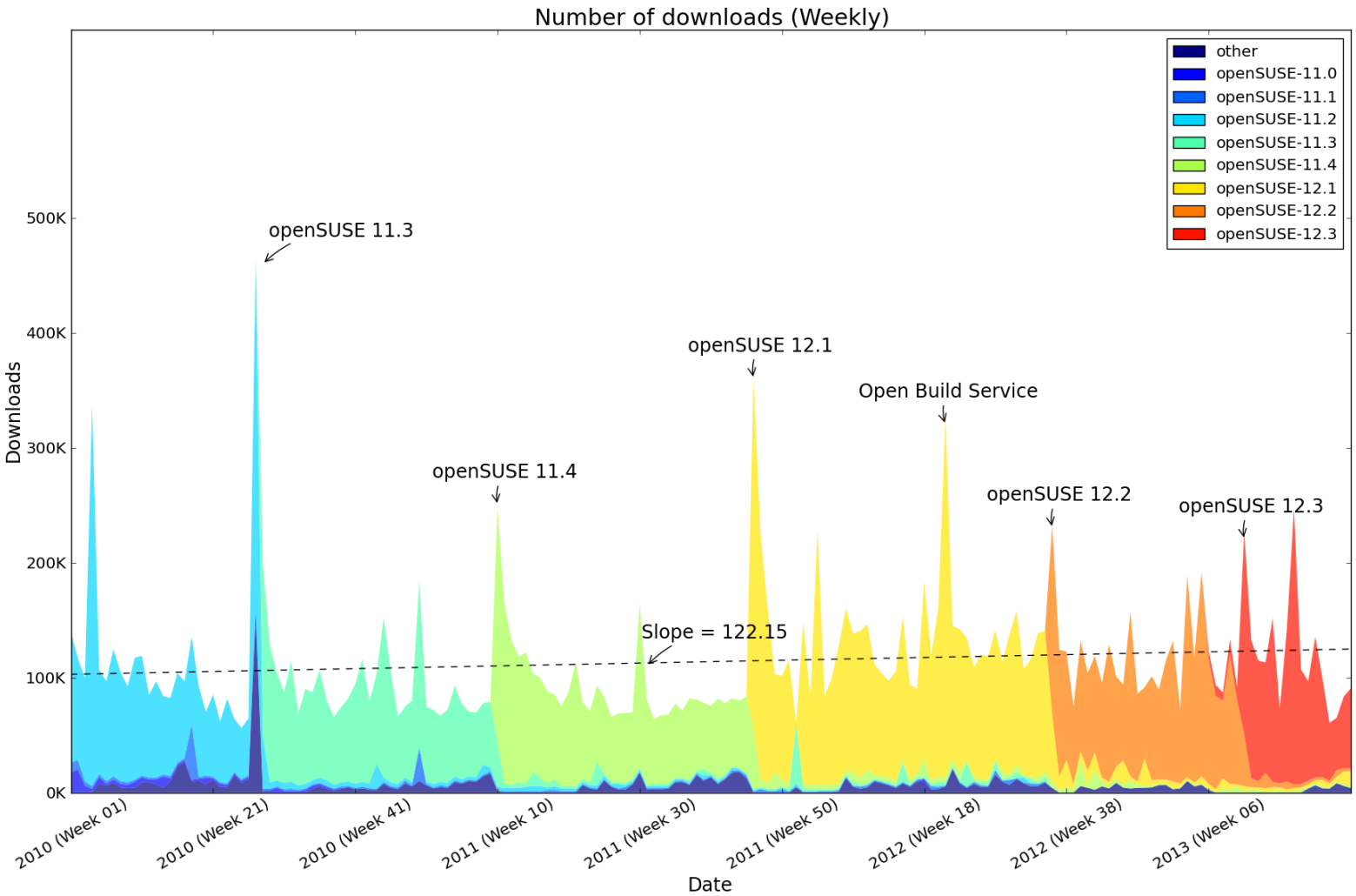
Methodology

- Take all ISO downloads
- Group ISOs into products (openSUSE versions)
- We do not collapse by IP, so:
  - every download from the same proxy, and
  - every different product downloaded by the same user
  - ... are counted independently
- Raw data is 2Gb per day, compressed!
- Now 1.5Tb in BerkeleyDB queues

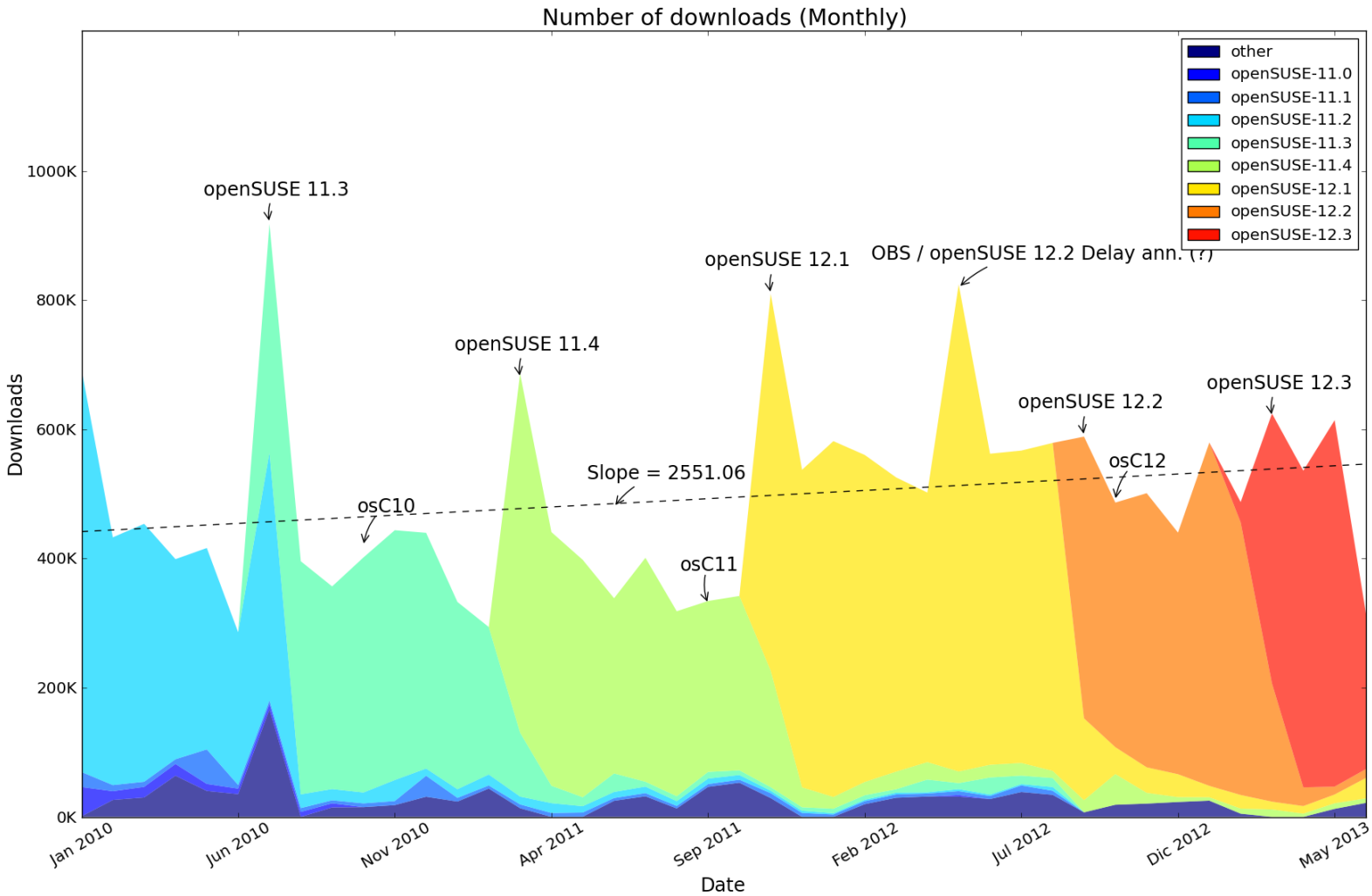
Daily Downloads



Weekly Downloads



Monthly Downloads



Download Model

Date	X	Y (downloads)
01/01/2011	12	472309
01/01/2012	24	502921
01/01/2013	36	533533
01/01/2014	48	564145

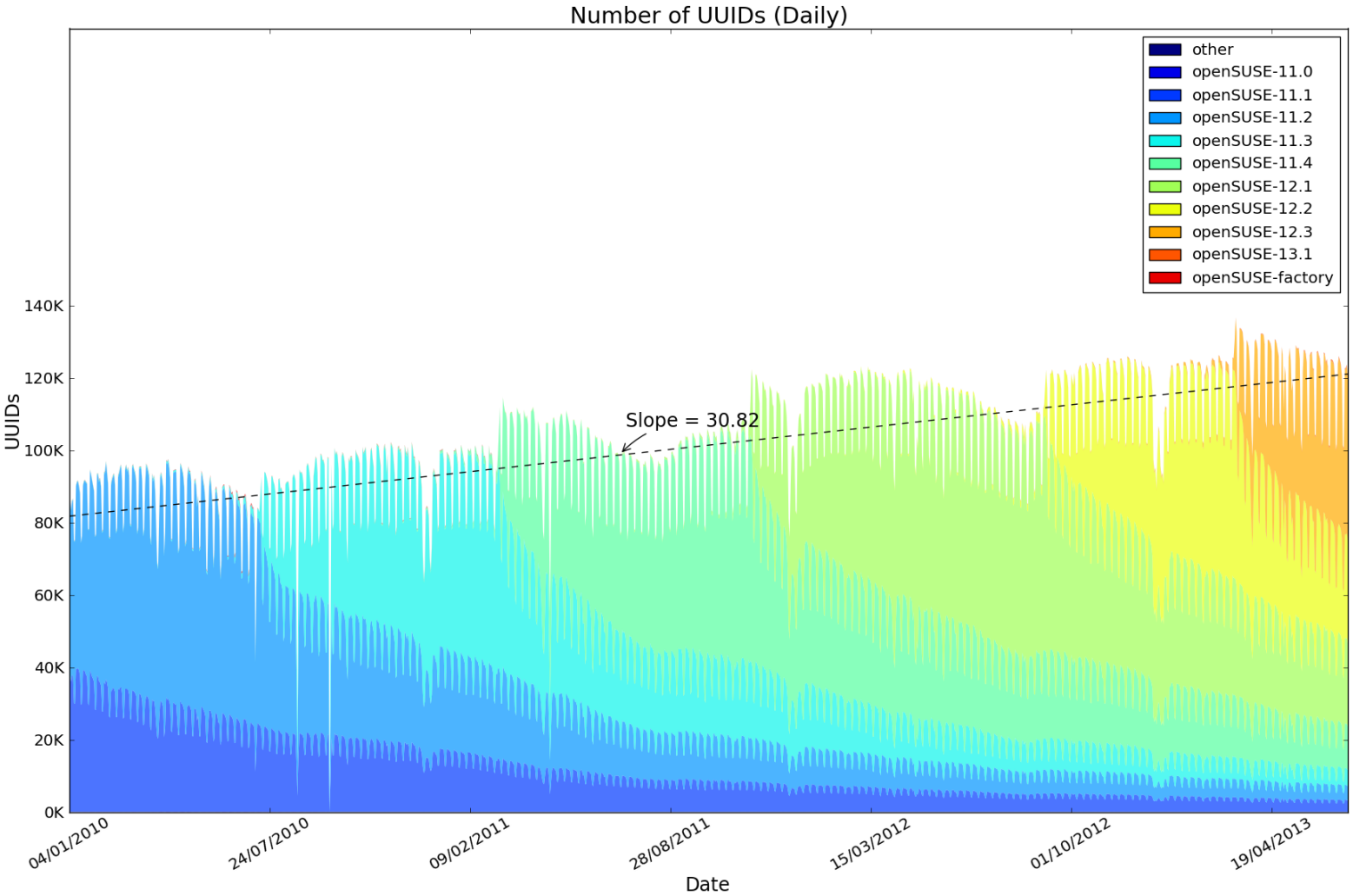
- Linear Model: $y = ax + b$
- Parameters: $\begin{cases} a(slope) = 2551 \\ b = 441697 \end{cases}$

Installations (UUID)

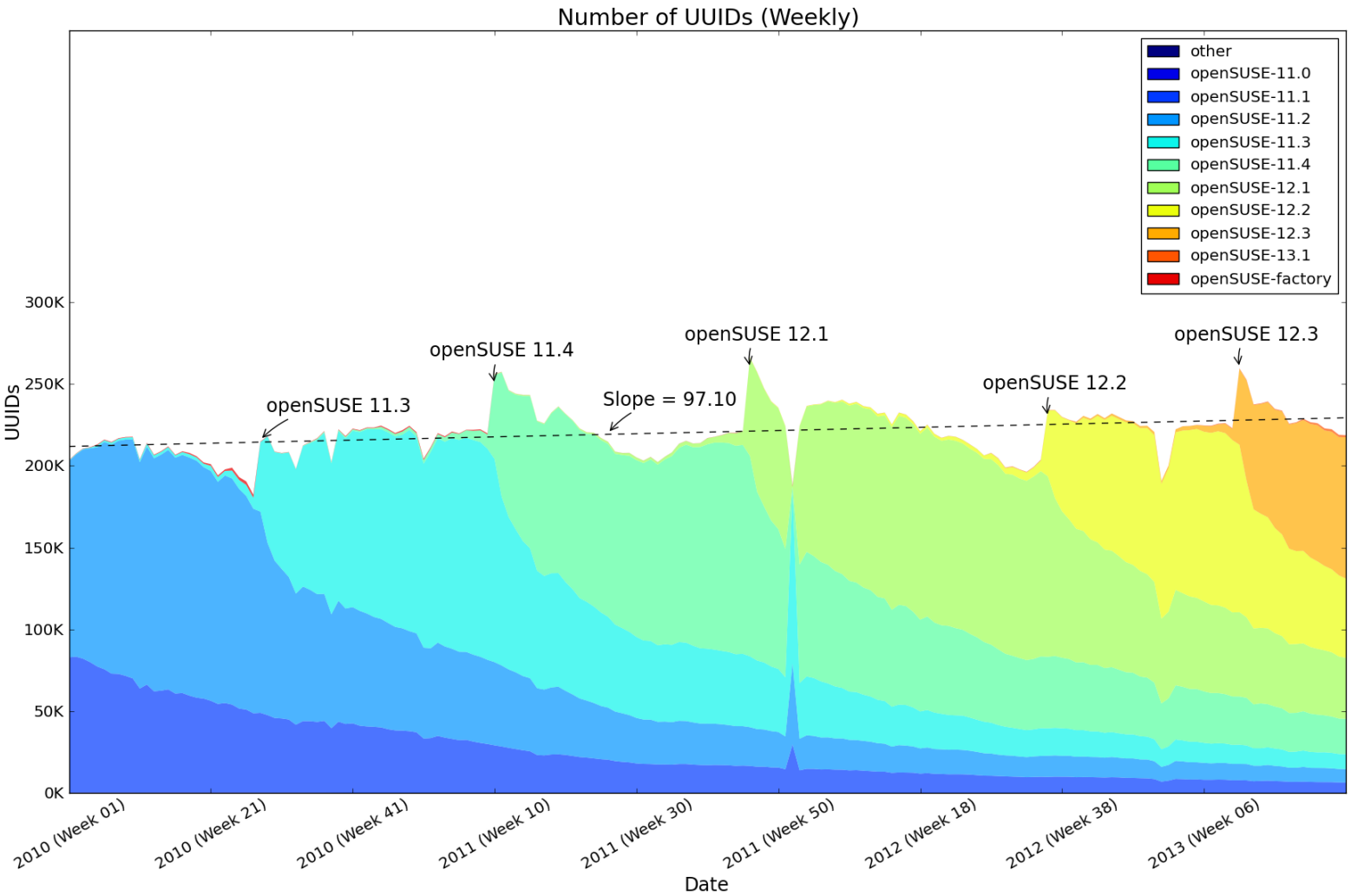
Methodology

- Take zypper signatures, and extract UUIDs
- An UUID represent an installation
- The UUID survives an update
- If we see an UUID in a time frame, is counted once
- We do collapse by UUID, so:
  - every installation is counted once in the time frame
- We use the same data store from downloads

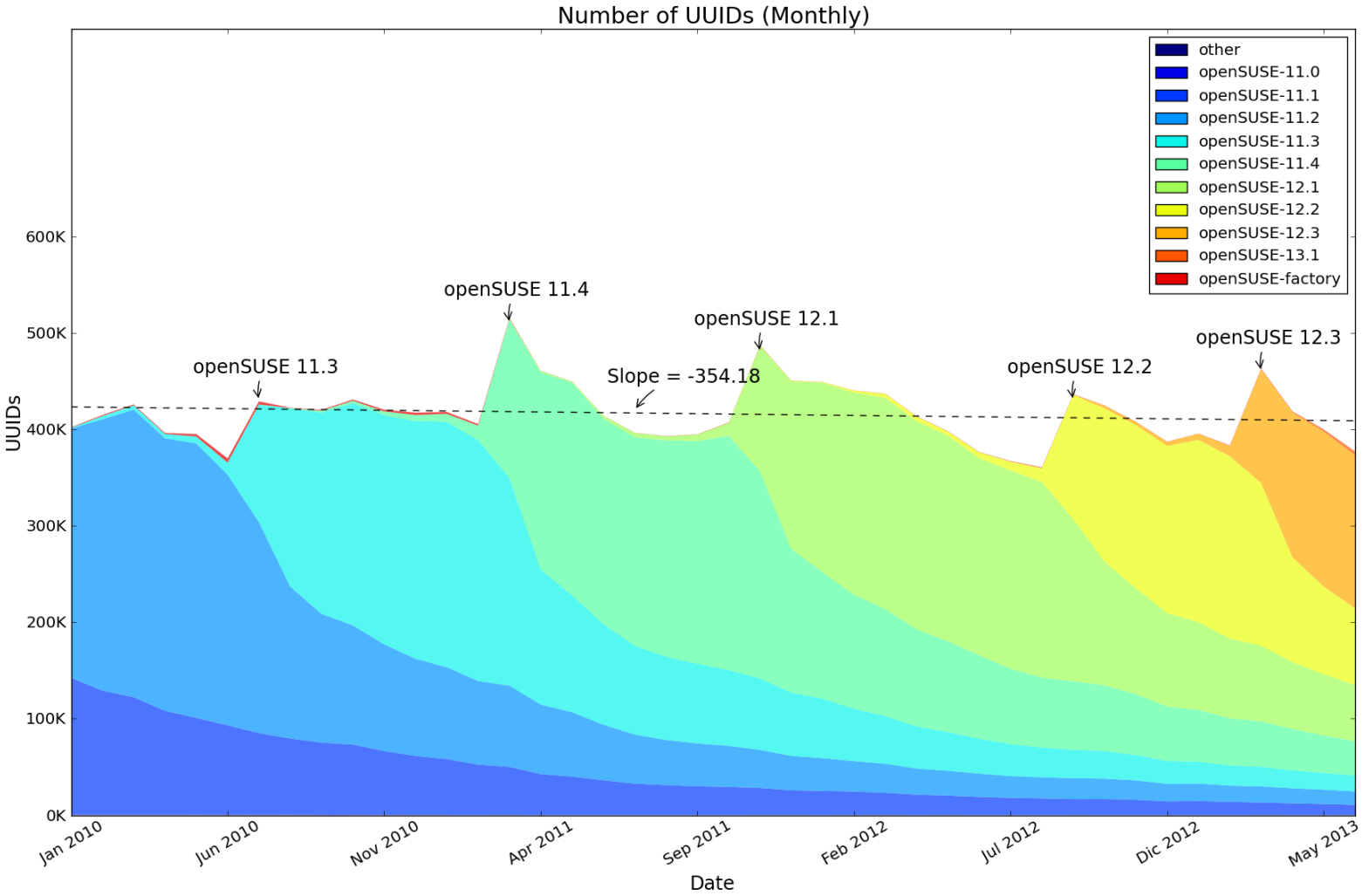
Daily UUIDs


Number of UUIDs (Daily)
other
openSUSE-11.0
openSUSE-11.1
openSUSE-11.2
openSUSE-11.3
openSUSE-11.4
openSUSE-12.1
openSUSE-12.2
openSUSE-12.3
openSUSE-13.1
openSUSE-factory
Slope = 30.82
UUIDs
0K
20K
40K
60K
80K
100K
120K
140K
04/01/2010
24/07/2010
09/02/2011
28/08/2011
15/03/2012
01/10/2012
19/04/2013
Date

Weekly UUIDs



Monthly UUIDs



Users Model

Date	X	Y (Users)
01/01/2011	12	419021
01/01/2012	24	414771
01/01/2013	36	410521
01/01/2014	48	406270

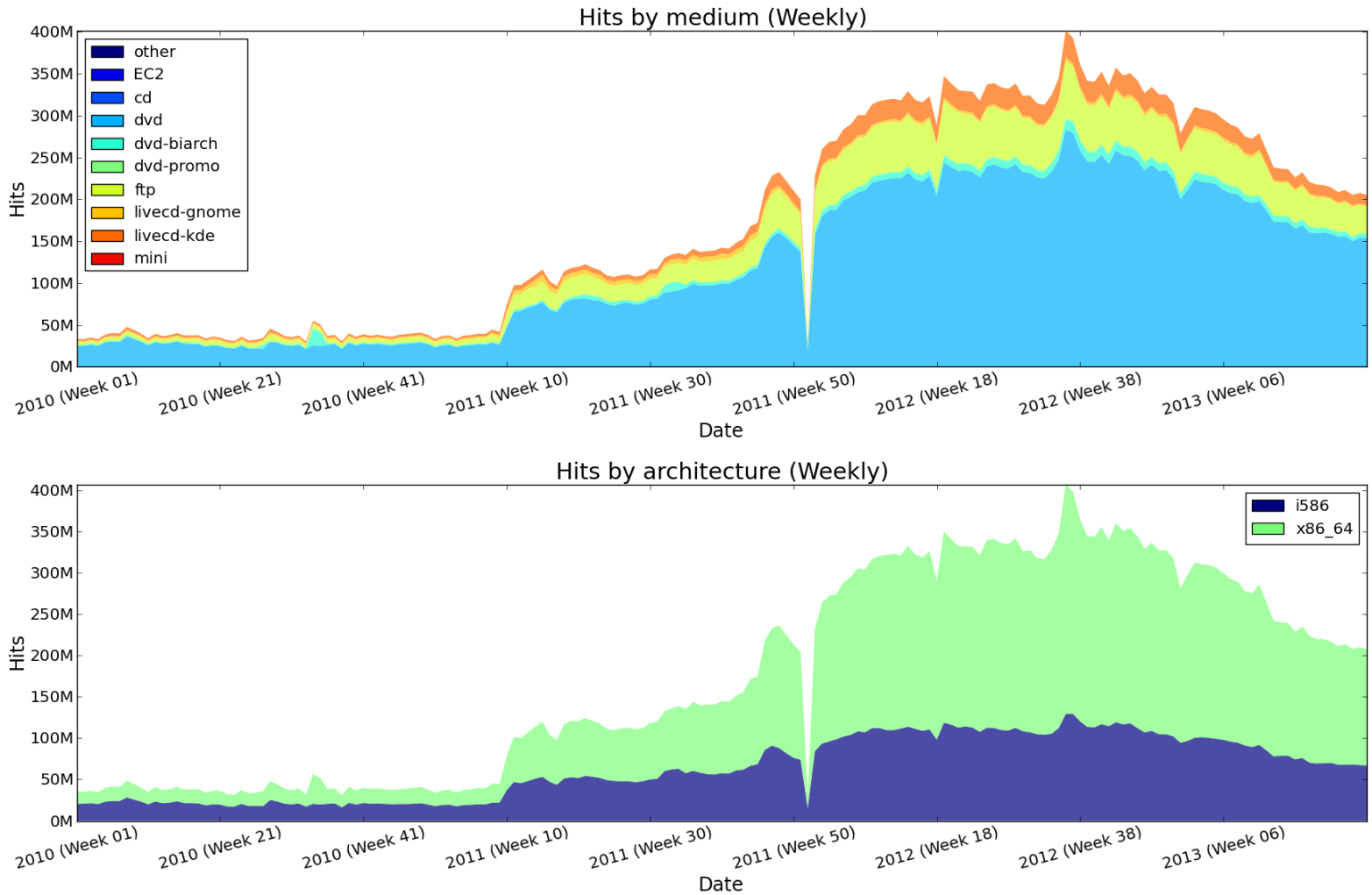
- Linear Model: $y = ax + b$
- Parameters: $\begin{cases} a(slope) = -354.18 \\ b = 423271 \end{cases}$

Medium and Architecture

Methodology

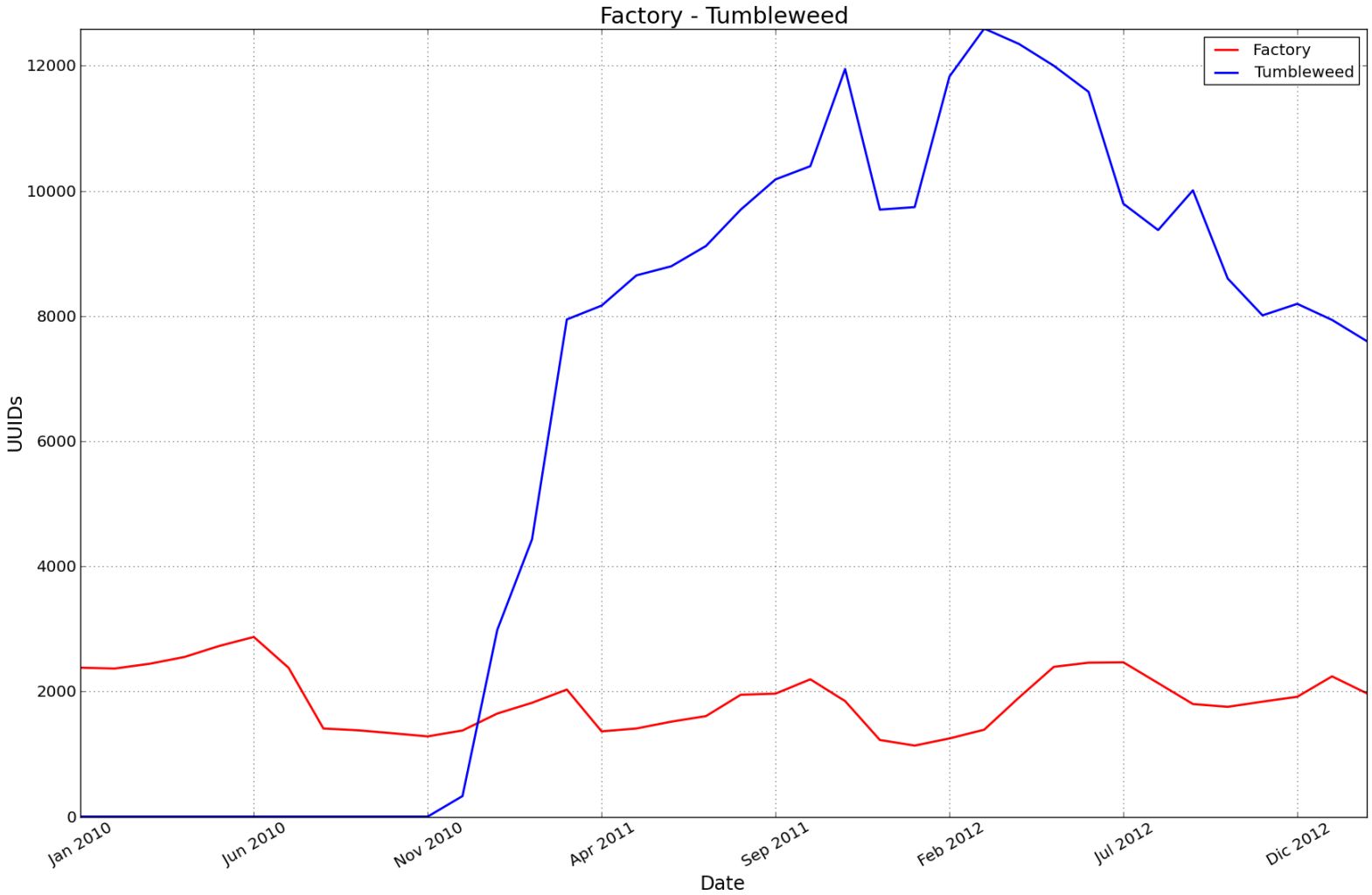
- Information extracted from zypper signature
- Very straightforward approach: every hit is a count
- Caution here, they are not users, only hits

Medium and Arch



Factory and Tumbleweed

Factory and Tumbleweed - UUID count

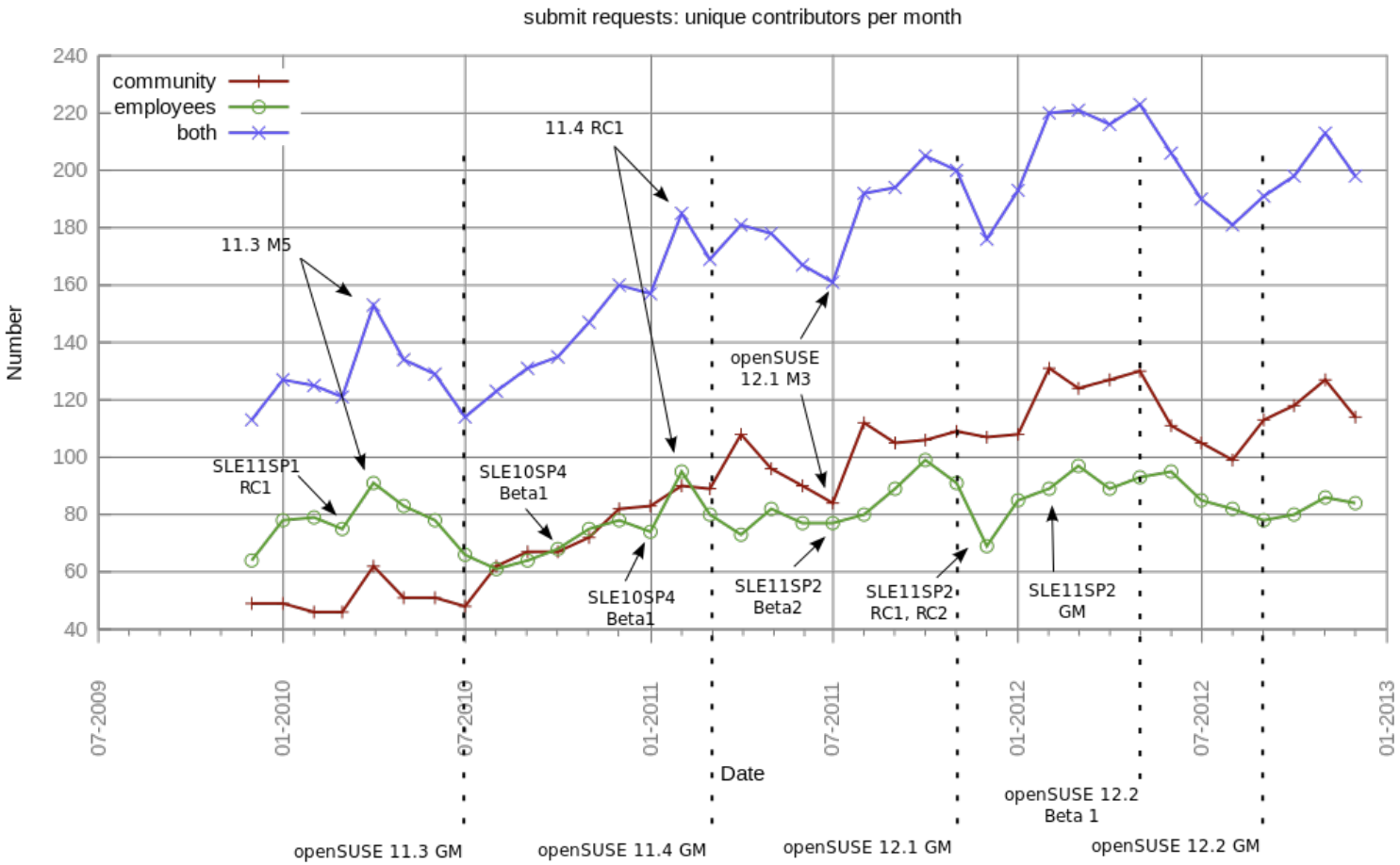


OBS

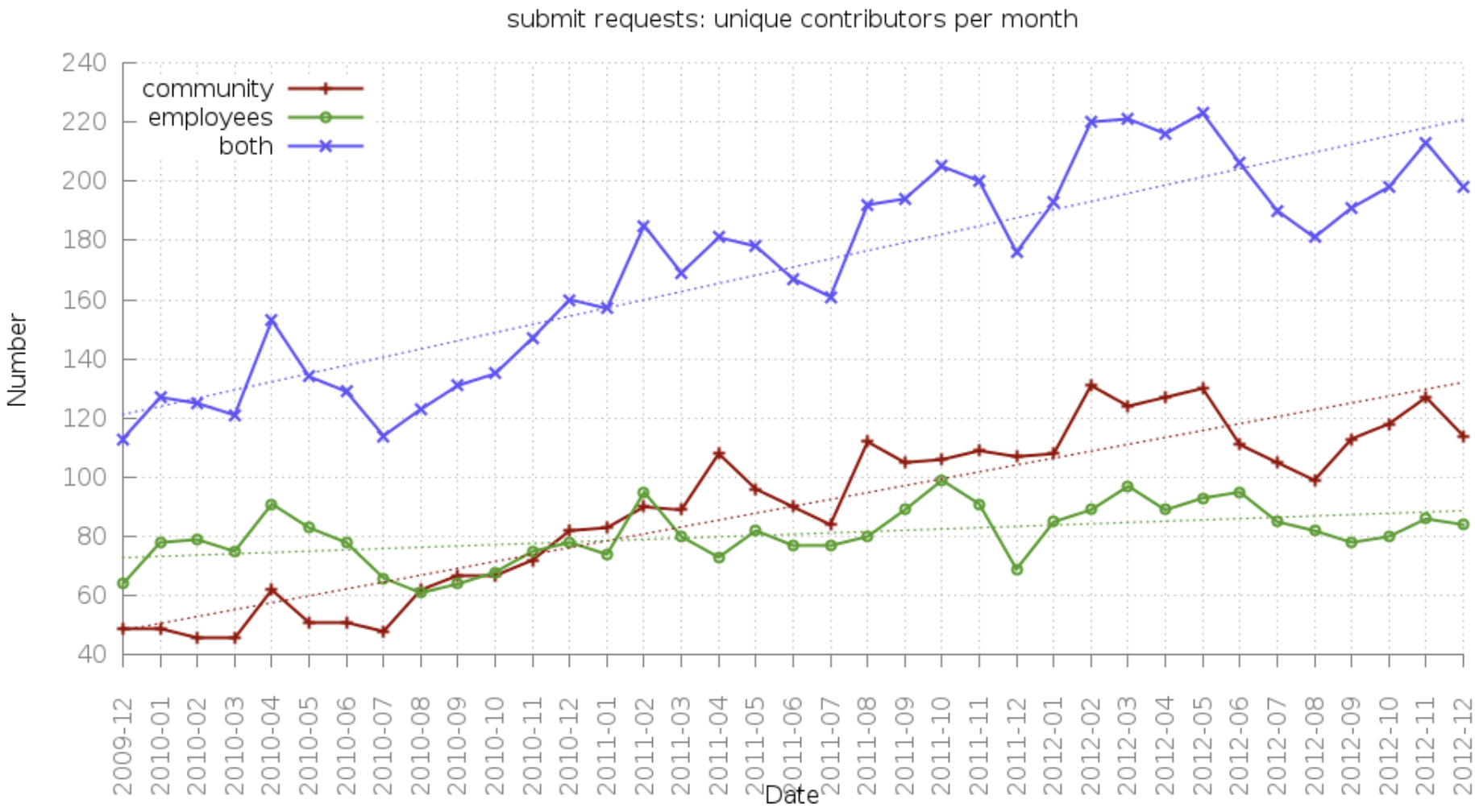
Methodology

- Use the OBS API to get Submit Request (SR) list
- SR are to Factory and to Devel projects
- We are counting users in the time frame (month)
- Community members are identified by email
- Use a linear model to analyze the evolution

Submit Request (Factory and Devel)

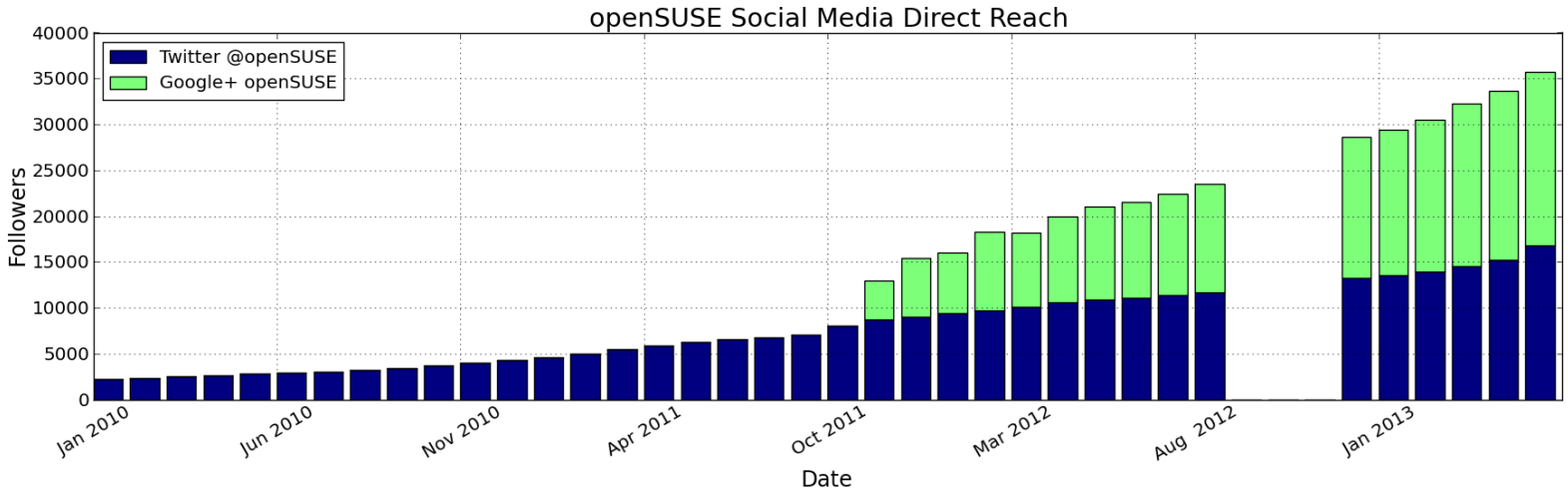


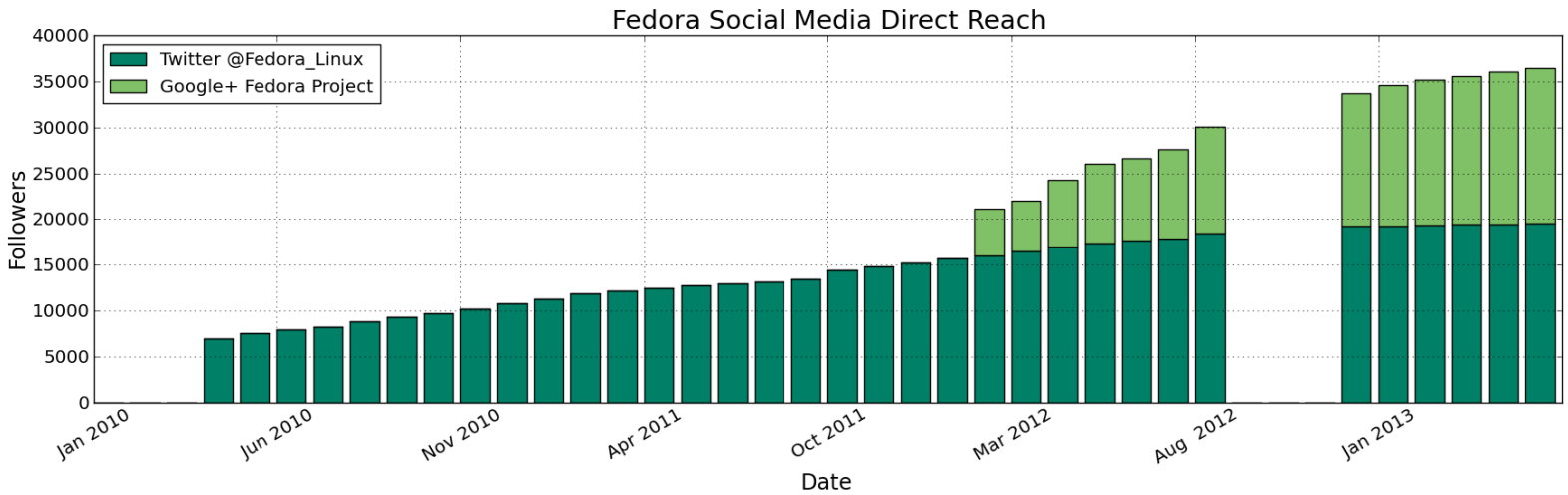
Submit Request model



Social Media Data

Social Media Direct Reach



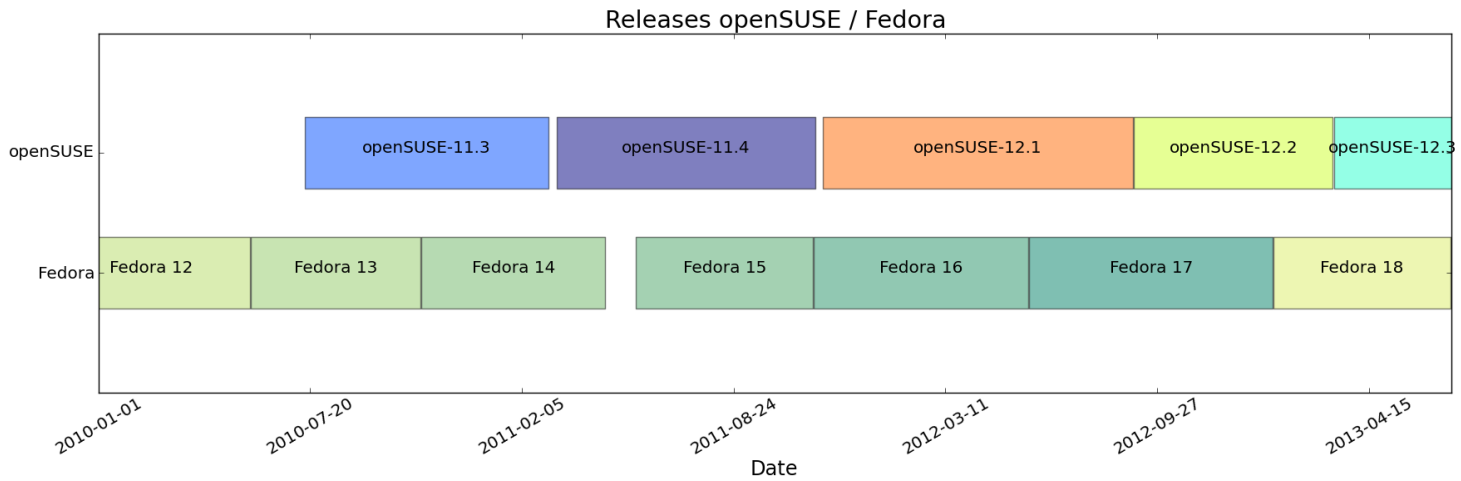


openSUSE vs. Fedora

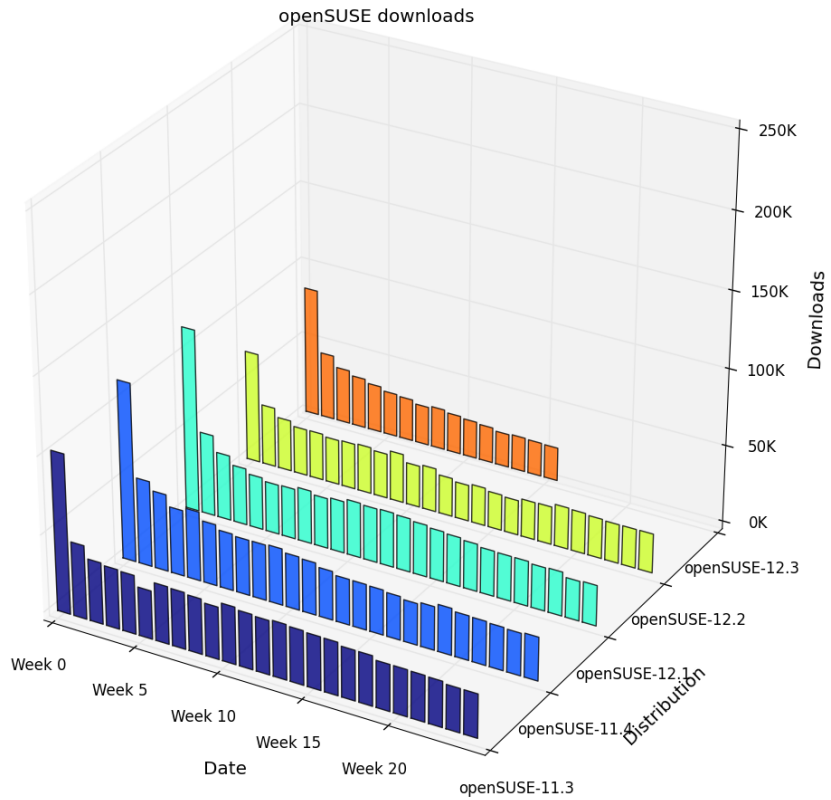
Methodology

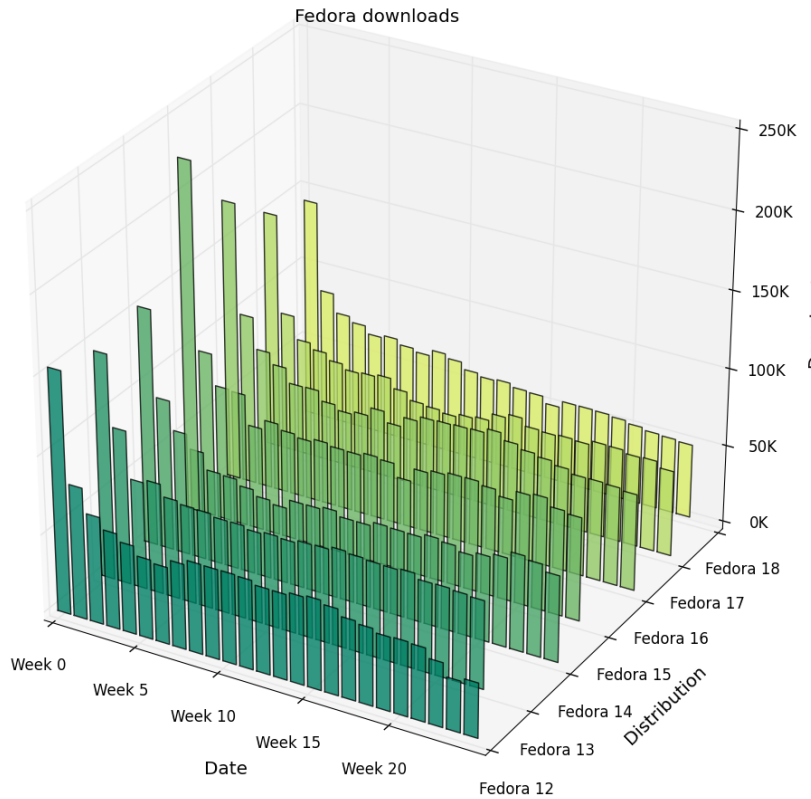
- Use the same methodology that Fedora uses
- Scripts, data and description found in Fedora Statistics wiki http://fedoraproject.org/wiki/Statistics
- Kudos for Fedora!
- Use new IPs never seen for this project as a basis to count users
- Use new IPs per day to count downloads
- Mangled our data to follow both rules

Reference timeline

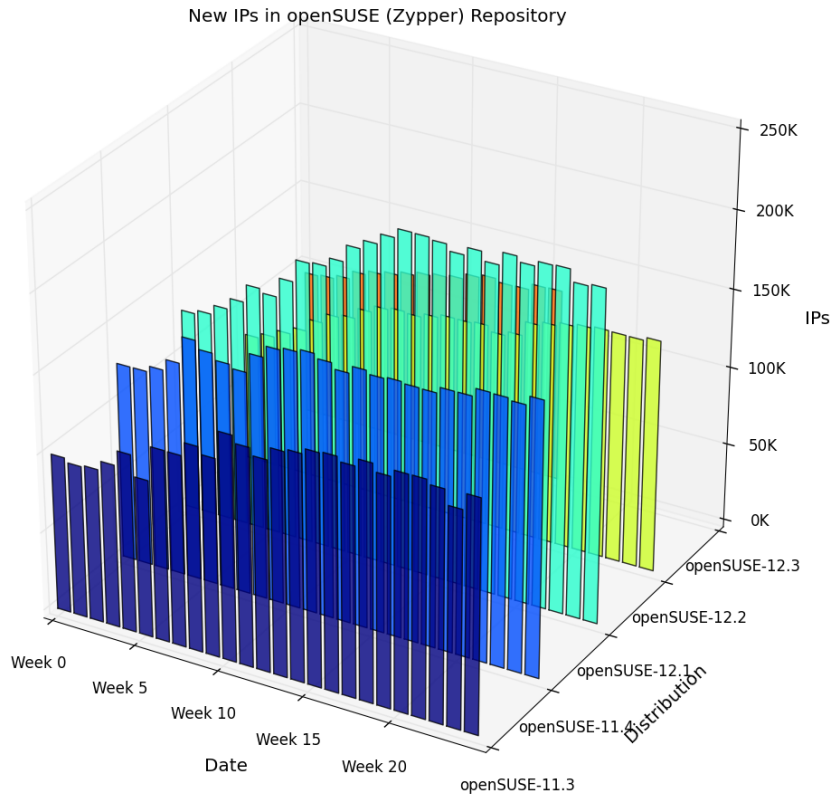


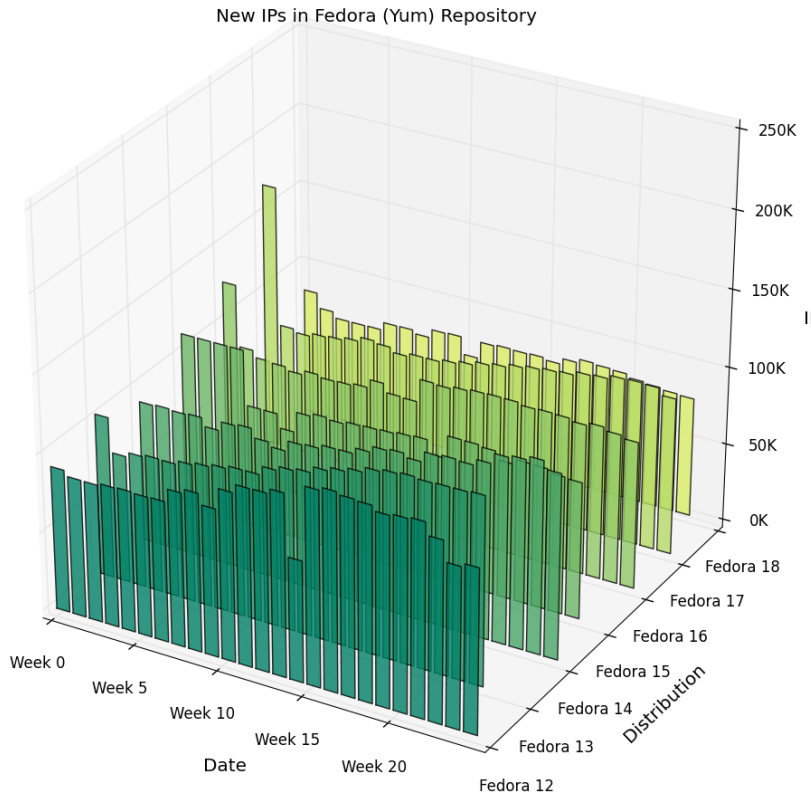
Comparing downloads





Comparing users





Conclusion

We have less downloads, but more users!

(if I didn't make a mistake)

Endnote

Suse is hiring

Join the Geeko!

Learn more about SUSE's openings, globally:

1. Talk to our colleagues at the booth
2. Check out our careers page www.suse.com/careers
3. Contact our recruiting team at jobs@suse.com

Thanks

Thank you for your attention.