



PG Day Toulouse 2017

La communauté des développeurs de PostgreSQL

Daniel Vérité, juin 2017



Les premiers développeurs : l'époque Berkeley

~ **1986**

Michael Stonebraker
et 5 étudiants de l'Université de
Berkeley créent **Postgres**.

1ère version = 80000 lignes
de code, 80% en C, 20 % en Lisp.
Langage de requêtes : PostQuel.





La période Berkeley

4 versions majeures développées, jusqu'en 1994, innovant notamment sur :

- Extensibilité des types, opérateurs, index.
- Héritage, orientation objet.
- Règles métiers.
- Versioning (time travel).



Première version avec SQL

1994-95

Jolly Chen et Andrew Yu remplacent PostQuel par SQL et sortent Postgres95.



La licence est de type BSD, ouverte aux dérivés commerciaux.



Première version « communautaire »

1996

Un groupe d'enthousiastes prend le relai pour développer Postgres en commun à travers Internet.

Le projet est renommé **PostgreSQL** et démarre en 6.0

(postgres95 est rétrospectivement baptisé 5.0)



PostgreSQL Global Development Team

1996

- Marc Fournier, Canada
- Thomas Lockhart, Etats-Unis
- Vadim Mikheev, Russie
- Bruce Momjian, Etats-Unis



2006 : PGCon Toronto Anniversaire des 10 ans





PostgreSQL Core Team

2017

- Peter Eisentraut, Etats-Unis
- Magnus Hagander, Suède
- Tom Lane, Etats-Unis
- Bruce Momjian, Etats-Unis
- Dave Page, Angleterre



La mailing-list

Les discussions sur le développement et la maintenance de PostgreSQL se font sur la mailing-list « officielle »: **pgsql-hackers**

En 2016 :

- 25500 messages (~70/jour).
- 560 expéditeurs différents.



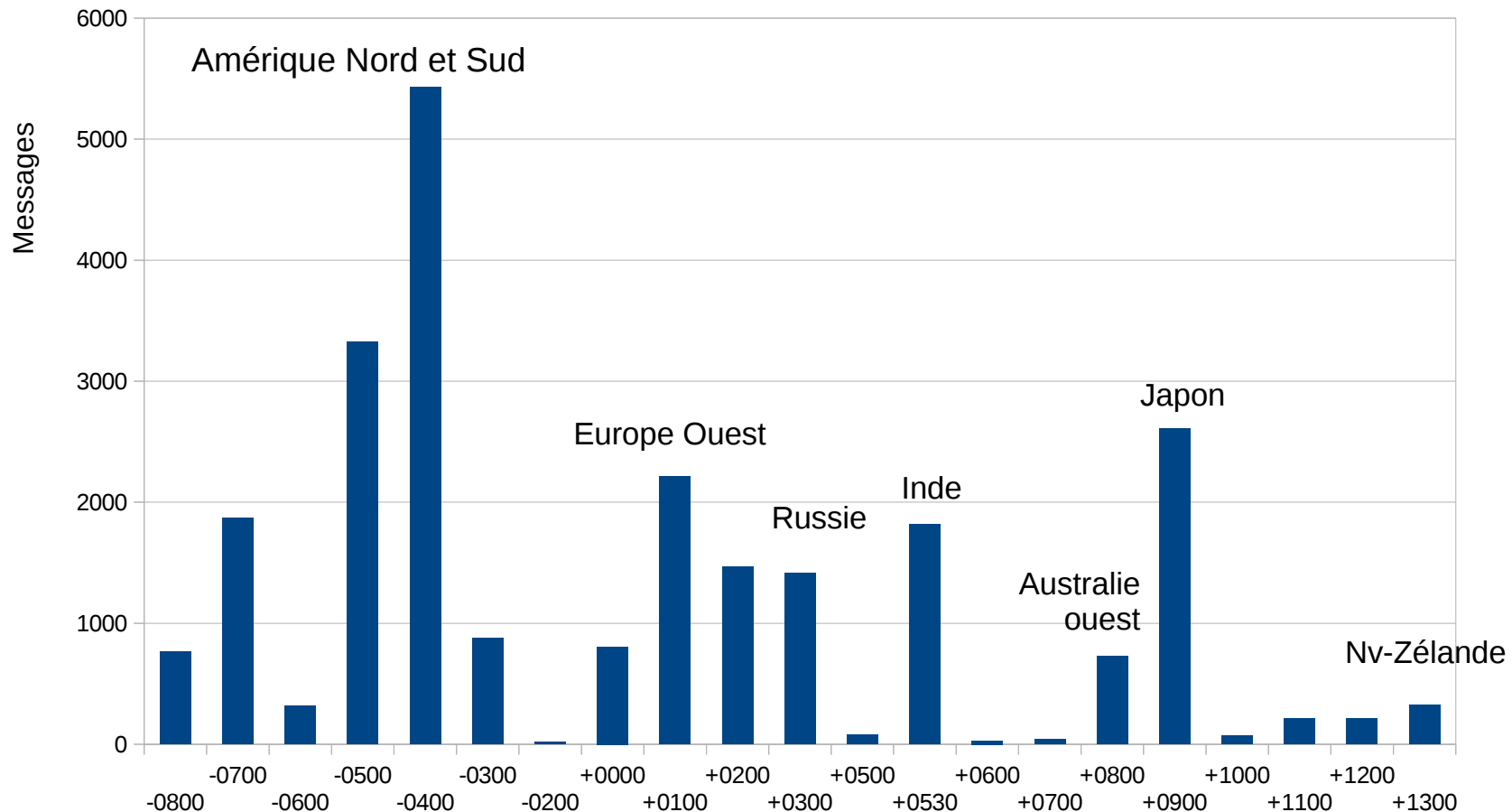
Le contenu des discussions

- Le détail des patches (⚠ discussions très techniques).
- La recherche de consensus sur des propositions (exemple : `xlog` → `wal`).
- Les outils de développement.
- L'organisation (planning de diffusion, Commit Fests).
- Les retours des utilisateurs (exemple : Uber).



La communauté dans le monde

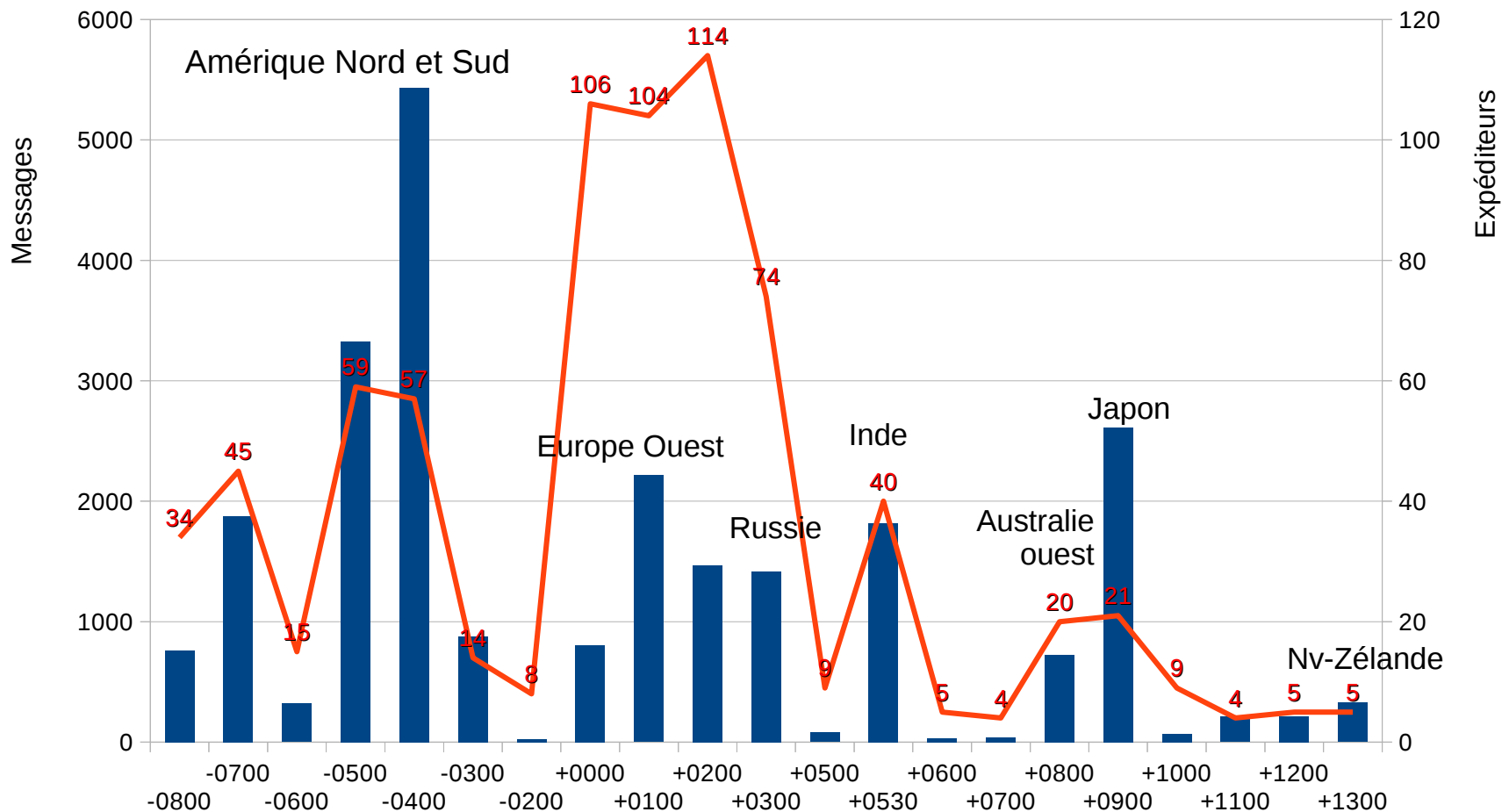
Participations à -hackers par fuseau horaire (2016)





La communauté dans le monde

Participants à -hackers par fuseau horaire (2016)





Le matériau : le code source

Le code source de la **version 10** comporte
~3,3 millions de lignes dont ~1,4 de code C :

- `bin/` : 110k lignes
- `contrib/` : 100k lignes
- `backend/` : 910k lignes
- `interfaces/` : 122k lignes
- `doc/` : 376k lignes
- messages en 12 langues: 490k lignes



Les patches

Patch = fichier de modifications faisant passer le code source d'un état à un autre. Pour :

- Corriger un bug.
- Corriger la doc.
- Améliorer une fonctionnalité.
- Ajouter une fonctionnalité.

~ 1900 patches committés entre 9.6.0 et 10-beta1



Proposer ses patches

Pourquoi ?

- Par altruisme et esprit de partage : efforts partagés, bénéfices partagés.
- Pour l'effet de levier de la communauté.
- Pour déléguer à la communauté la maintenance d'une fonctionnalité.



Les rôles des développeurs

- Auteur
- Examineur (reviewer)
- Committer

Le « **honor system** » :

Tout auteur devrait examiner des patches de même difficulté et même nombre (au moins) que ses propres contributions.



22 committers en 2017

- Bruce Momjian (momjian)
- Tom Lane (tgl)
- Michael Meskes (meskes)
- Tatsuo Ishii (ishii)
- Peter Eisentraut (petere)
- Teodor Sigaev (teodor)
- Joe Conway (joe)
- Alvaro Herrera (alvherre)
- Andrew Dunstan (adunstan)
- Magnus Hagander (mha)
- Heikki Linnakangas (heikki)
- Robert Haas (rhaas)
- Simon Riggs (sriggs)
- Greg Stark (stark)
- Kevin Grittner (kgrittn)
- Jeff Davis (jdavis)
- Stephen Frost (sfrost)
- Fujii Masao (fujii)
- Noah Misch (noah)
- Andres Freund (andres)
- Dean Rasheed (deanr)
- Andrew Gierth (rhodiumtoad)



42 committers depuis 1996

```
$ git shortlog -sn
```

```
13314 Bruce Momjian
12492 Tom Lane
3575 Peter Eisentraut
1806 Robert Haas
1491 Marc G. Fournier
1240 Heikki Linnakangas
1078 Thomas G. Lockhart
984 Alvaro Herrera
763 Magnus Hagander
725 Michael Meskes
616 Neil Conway
556 Andrew Dunstan
519 Vadim B. Mikheev
402 Simon Riggs
391 Tatsuo Ishii
386 Andres Freund
337 Teodor Sigaev
244 Fujii Masao
222 Noah Misch
220 Stephen Frost
176 Hiroshi Inoue
157 Bryan Henderson
152 Jan Wieck
143 Barry Lind
121 Kevin Grittner
107 Dave Cramer
92 Joe Conway
78 Itagaki Takahiro
69 Dennis Bjorklund
53 Philip Warner
53 PostgreSQL Daemon
52 Peter Mount
47 D'Arcy J.M. Cain
20 Byron Nikolaidis
20 Greg Stark
16 Edmund Mergl
15 Jeff Davis
9 Dean Rasheed
6 Julian Assange
5 Andrew Gierth
2 Vince Vielhaber
1 Kris Jurka
```

(comparaison : l'équivalent sur mysql@github sort 1375 committers)



Les contributeurs en 2016

Extrait de l'étude de Robert Haas :

<http://rhaas.blogspot.fr/2017/04/who-contributes-to-postgresql.html>

90 % du « nouveau code » a été contribué principalement par 37 auteurs
(sur 141 contributeurs au total)

#	author	lines	commits	#	author	lines	commits
1	Tom Lane	62077	637	20	Alvaro Herrera	2636	56
2	Amit Langote [*]	9889	30	21	Jeevan Chalke [*]	2454	2
3	Robert Haas	9685	108	22	Etsuro Fujita [*]	2378	21
4	Stephen Frost	9177	46	23	Kyotaro Horiguchi [*]	2171	19
5	Teodor Sigaev	8345	28	24	Masahiko Sawada [*]	2129	20
6	Michael Paquier [*]	7778	106	25	Peter Geoghegan [*]	2121	25
7	Andres Freund	5913	61	26	Tomas Vondra [*]	2084	10
8	David Rowley [*]	5582	26	27	Craig Ringer [*]	2063	17
9	Alexander Korotkov [*]	5174	11	28	Artur Zakirov [*]	1962	13
10	Peter Eisentraut	4877	161	29	Andrew Gierth [*]	1726	5
11	Heikki Linnakangas	4378	42	30	Dean Rasheed	1627	10
12	Thomas Munro [*]	3535	31	31	Daniel Vérité [*]	1530	4
13	Magnus Hagander	3494	26	32	Emre Hasegeli [*]	1497	5
14	Amit Kapila [*]	3480	35	33	Joe Conway	1471	10
15	Kevin Grittner	3103	23	34	Noah Misch	1430	31
16	Andreas Karlsson [*]	3062	33	35	Jim Nasby [*]	1404	11
17	Bruce Momjian	3049	27	36	Petr Jelinek [*]	1400	10
18	Fabien Coelho [*]	2768	22	37	Pavel Stehule [*]	1254	6
19	Shigeru Hanada [*]	2752	3				



Les CommitFests

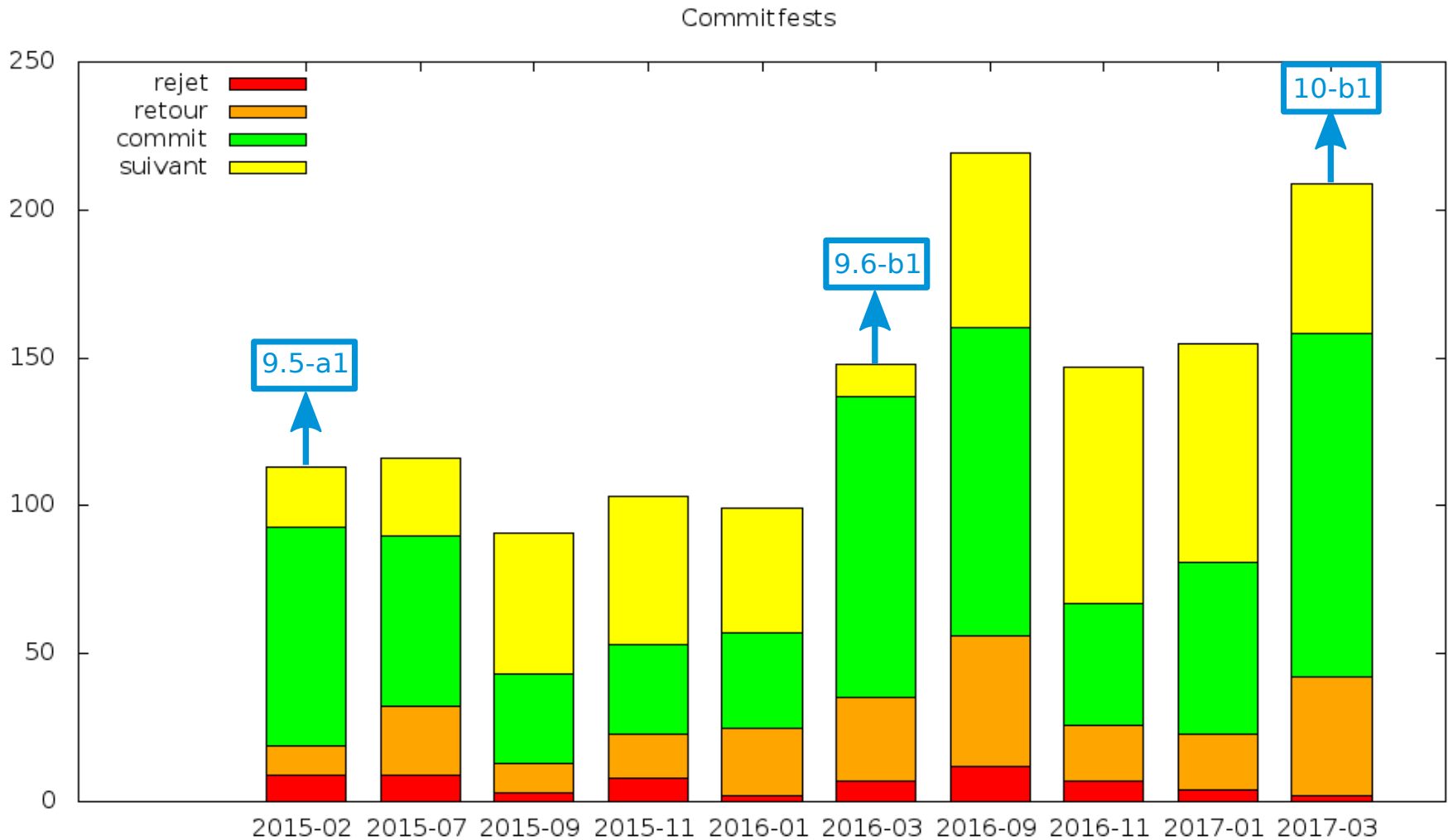
<https://wiki.postgresql.org/wiki/CommitFest>

"A CommitFest (CF) is a periodic break to PostgreSQL development that focuses on patch review and commit rather than new development."

- Calendrier : défini en début de cycle de version
- Durée : ~ 1 ou 2 mois
- Nombre : ~ 4 commitfests par an



Bilan des patches par CommitFest





Un patch : crosstabview 1/4

<https://wiki.postgresql.org/wiki/Crosstabview>

Pivot dynamique exécuté par psql

Exemple : requête sortant le nombre de messages par mois sur quelques mailing-lists

```
SELECT to_char(d, 'Mon') as month, name, cnt FROM (
SELECT date_trunc('month', msg_date) as d,
       t.name,
       count(*) as cnt
FROM mail JOIN mail_tags using(mail_id) JOIN tags t
on(t.tag_id=mail_tags.tag)
WHERE t.tag_id in (7,8,12,34,79)
AND msg_date>='2016-01-01'::date and msg_date<'2017-01-01'::date
GROUP BY 1,2
) s ORDER BY extract(month from d),3;
```



crosstabview 2/4

month	name	cnt
Jan	announce	24
Jan	performance	76
Jan	general	1284
Jan	hackers	2185
Feb	announce	17
Feb	performance	62
Feb	general	738
Feb	hackers	2091
Mar	interfaces	4
Mar	announce	14
Mar	performance	74
Mar	general	864
Mar	hackers	3451
Apr	interfaces	2
Apr	announce	19
Apr	performance	52
Apr	general	809
Apr	hackers	2274
May	interfaces	1
May	announce	9
May	performance	51
May	general	814
May	hackers	1663
Jun	interfaces	2
Jun	announce	14
Jun	performance	107
Jun	general	604
Jun	hackers	1573

Jul	interfaces	1
Jul	announce	12
Jul	performance	79
Jul	general	767
Jul	hackers	1432
Aug	interfaces	1
Aug	announce	18
Aug	performance	65
Aug	general	734
Aug	hackers	2402
Sep	interfaces	1
Sep	announce	22
Sep	performance	130
Sep	general	770
Sep	hackers	2435
Oct	interfaces	1
Oct	announce	16
Oct	performance	51
Oct	general	718
Oct	hackers	1937
Nov	announce	15
Nov	performance	88
Nov	general	688
Nov	hackers	1970
Dec	interfaces	4
Dec	announce	15
Dec	performance	48
Dec	general	853
Dec	hackers	2105

(57 rows)



crosstabview 3/4

```
=> \crosstabview month name cnt
```

month	announce	general	hackers	performance	interfaces
Jan	24	1284	2185	76	
Feb	17	738	2091	62	
Mar	14	864	3451	74	4
Apr	19	809	2274	52	2
May	9	814	1663	51	1
Jun	14	604	1573	107	2
Jul	12	767	1432	79	1
Aug	18	734	2402	65	1
Sep	22	770	2435	130	1
Oct	16	718	1937	51	1
Nov	15	688	1970	88	
Dec	15	853	2105	48	4

```
(12 rows)
```




crosstabview 4/4

```
=> \crosstabview name month cnt
```

name	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
announce	24	17	14	19	9	14	12	18	22	16	15	15
general	1284	738	864	809	814	604	767	734	770	718	688	853
hackers	2185	2091	3451	2274	1663	1573	1432	2402	2435	1937	1970	2105
performance	76	62	74	52	51	107	79	65	130	51	88	48
interfaces			4	2	1	2	1	1	1	1		4

(5 rows)



Bilan patch crosstabview

- Discussion : 185 mails (dont 30 post-commit), 19 participants, dont 8 committers.
- Itérations : 8 mois (16 versions) entre « proof of concept » (290 lignes) et commit (1500 lignes).
- ~ 75 % du temps passé à communiquer, 25 % à programmer.



Les outils : pas de bugtracker !

Formalisme minimal pour le suivi de bugs :

- Un formulaire de saisie web public.
- Un ID de suivi assigné automatiquement.
- Tout le reste via la liste « **pgsql-bugs** »

Pas de consensus sur l'utilisation d'un outil de type Bugzilla / Jira / Redmine / Trac / ...



Les outils : la buildfarm

<https://buildfarm.postgresql.org/>

Réseau communautaire de « créatures » qui compilent automatiquement PostgreSQL pour diverses architectures et configurations.

Comprend 94 machines en juin 2017.



Les outils : code coverage report

<https://coverage.postgresql.org/>

Révèle les portions de code non mises à l'oeuvre lors des tests de non-régression.

src/backend/parser		84.6 %	15921 / 18825	90.8 %	414 / 456
src/backend/port		58.7 %	131 / 223	85.0 %	17 / 20
src/backend/postmaster		66.4 %	3914 / 5898	82.1 %	234 / 285
src/backend/regex		71.0 %	3178 / 4478	90.1 %	172 / 191
src/backend/replication		73.4 %	3224 / 4390	75.5 %	173 / 229
src/backend/replication/libpqwalreceiver		81.4 %	262 / 322	100.0 %	17 / 17
src/backend/replication/logical		84.3 %	3744 / 4441	93.8 %	242 / 258
src/backend/replication/pgoutput		89.7 %	174 / 194	100.0 %	15 / 15
src/backend/rewrite		87.3 %	1728 / 1980	96.2 %	77 / 80
src/backend/snowball		71.2 %	47 / 66	100.0 %	6 / 6
src/backend/snowball/libstemmer		4.8 %	520 / 10919	8.2 %	37 / 449
src/backend/statistics		89.1 %	620 / 696	85.4 %	41 / 48



Roadmap ? Todo-List ?

- Roadmap : pas vraiment

<https://www.postgresql.org/developer/roadmap/>

« We really do follow the mantra of letting developers scratch their own itches. »

- Todo-List : oui mais

<https://wiki.postgresql.org/wiki/ToDo>

« Do not assume that you can select one [feature], code it and then expect it to be committed. Always discuss design on Hackers list before starting to code. »



END;

```
SELECT * FROM questions;
```