NTT Open Source Software Center

pg_statsinfo
- More useful information for DBAs -

Tatsuhito Kasahara
NTT OSS Center

PGCon 2010 Ottawa

Agenda

NTT Open Source Software Center

- Background
- What's the pg_statsinfo ?
  - Goal
  - Overview
  - Output and input
  - How to work
  - Demo
- Future work
  - pg_statsinfo own challenges
  - Cooperate with PostgreSQL
- Conclusion

Background
- Troubles for novice DBAs -

What information do DBAs watch ?

NTT Open Source Software Center

- Information of DB status is important for DBAs
  - They have to check status of DB
- Now, PostgreSQL has many monitoring features
- PostgreSQL provides two kinds of information about it
  - Statistics views
  - Log messages

Statistics views and logs of PostgreSQL

NTT Open Source Software Center

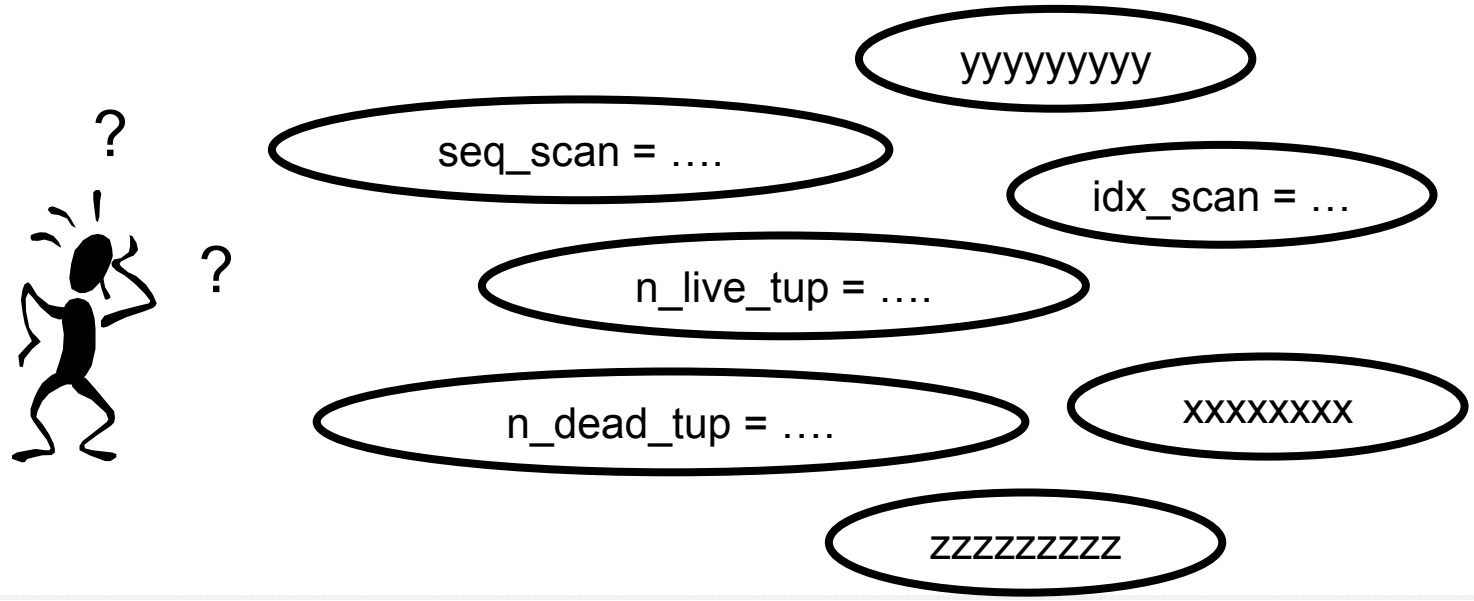
- They are very useful for experts DBAs☺
  - It is convenient to solve problems and detect some dangers
  - Experts can figure out the activity of PostgreSQL

seq_scan = …..
n_dead_tup= …..
n_live_tup= …..
“query” : calls …. : total_time : ….
statement :

Troubles for DBAs to see stats of PostgreSQL

NTT Open Source Software Center

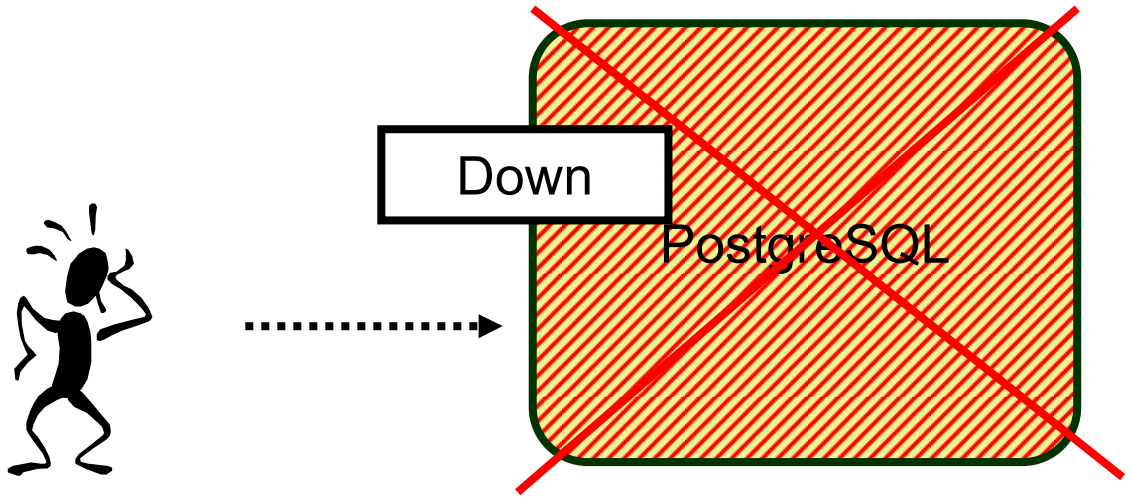
- Statistics are complex information for novice DBAs ☹
  - How do we translate these statistics ?
  - What values/status should we consider it to be healthy or not ?
  - What to do next ?



Troubles for DBAs to see stats of PostgreSQL

NTT Open Source Software Center

- It is too late to get statistics in case of troubles happened
  - They can not gather the statistics to resolve troubles



Troubles for DBAs to see logs of PostgreSQL

NTT Open Source Software Center

- It is difficult for DBAs to find which message is important
  - Because log file has various kinds of messages
    - Messages for DBAs,
    - Messages for developers
    - Performance information


```
……
[2010-05-01]  [postgres] WARNING:  nonstandard use of escape ...
[2010-05-01]  [postgres] LOG: archive command ….. failed
[2010-05-01]  [postgres] LOG: checkpoint starting: xlog
[2010-05-01]  [postgres] ERROR: syntax error at or near …
……
```

Troubles for DBAs to see logs of PostgreSQL

NTT Open Source Software Center

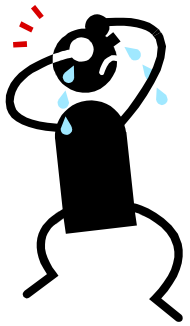
- It is too bother to pick up and aggregate log-messages which writes at problematic period
  - DBAs want to know <u>what happened at problematic period</u>
    - e.g.
    - Retrieve summary autovacuum information for a table when the table has troubles

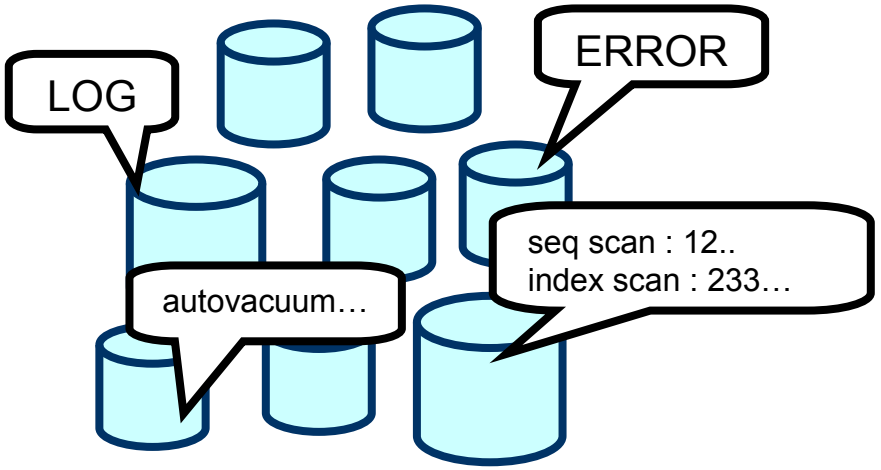

```
……
[2010-05-01]  [postgres] WARNING:  nonstandard use of escape ...
[2010-05-01]  [postgres] LOG: autovacuum …
[2010-05-01]  [postgres] LOG: autovacuum
[2010-05-01]  [postgres] ERROR: syntax error at or near
……
```

Troubles with many DBs

NTT Open Source Software Center

- DBAs have to manage many Machines and DBs recently.
  - It's too bothering to gather statistics from many DBs
  - They want to see a summarized and graphical information
    - Rather than enormous quantity of statistics and log-messages !





True necessity for DBAs

NTT Open Source Software Center

- DBAs don't see statistics and log messages
  - They needs to more simple information

  - "In Trouble or Safe ?"
  - "What's the reason ?"
  - "How should I do ?"

What's the pg_statsinfo ?
- Goal -

Goal of pg_statsinfo

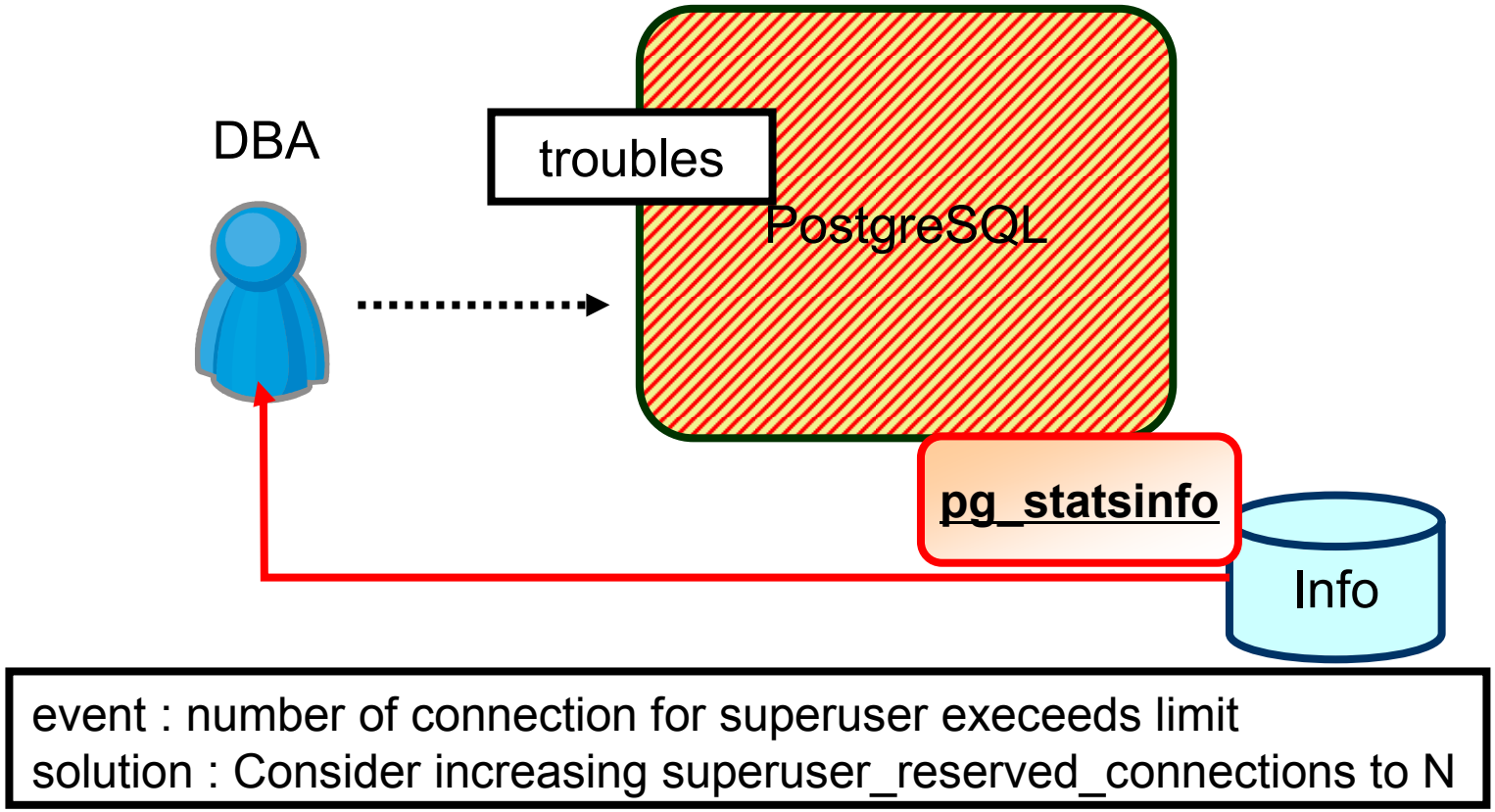
- Goals of pg_statsinfo are following points

- Alerts DBAs to occurrence and cause of troubles
  - Also alert contains ways of typical solution to troubles
- Shows the report to check and see the database activities and status at problematic

Alerts and reports

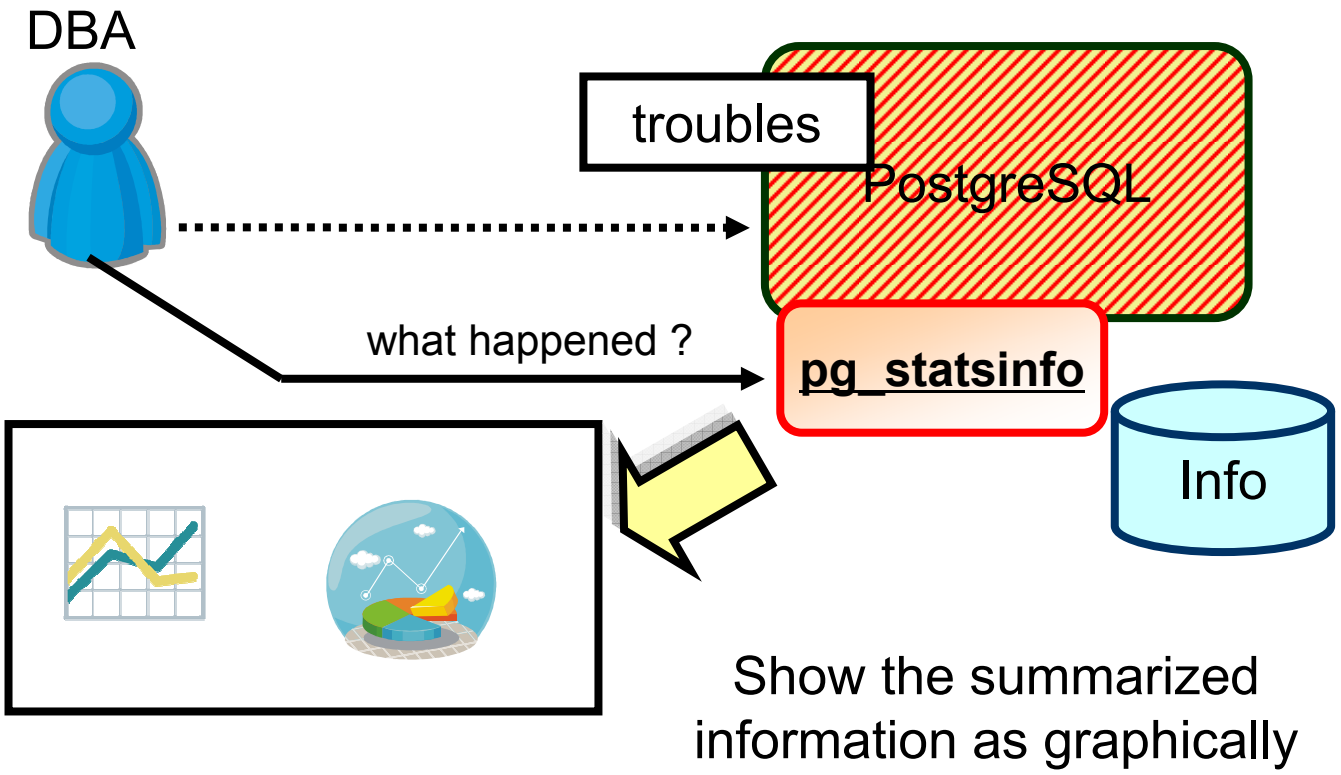
Alerts for DBAs

- DBAs only have to receive alerts generated by pg_statsinfo
  - They don't have to need to search and analysis



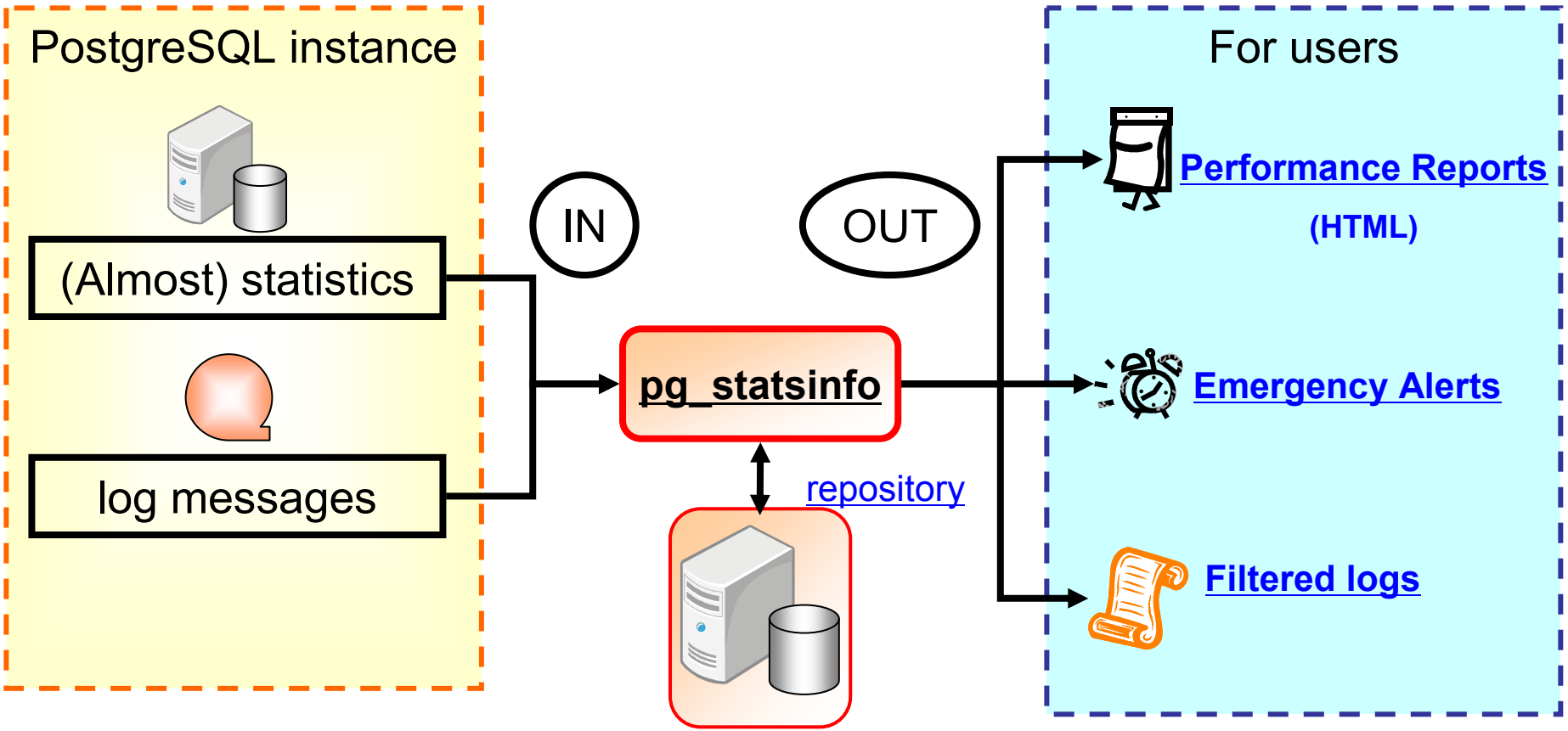
Showing report to check cause of troubles

- DBAs can see the graphs and summary info to check cause of troubles
  - DBAs can also see their DB is healthy or not

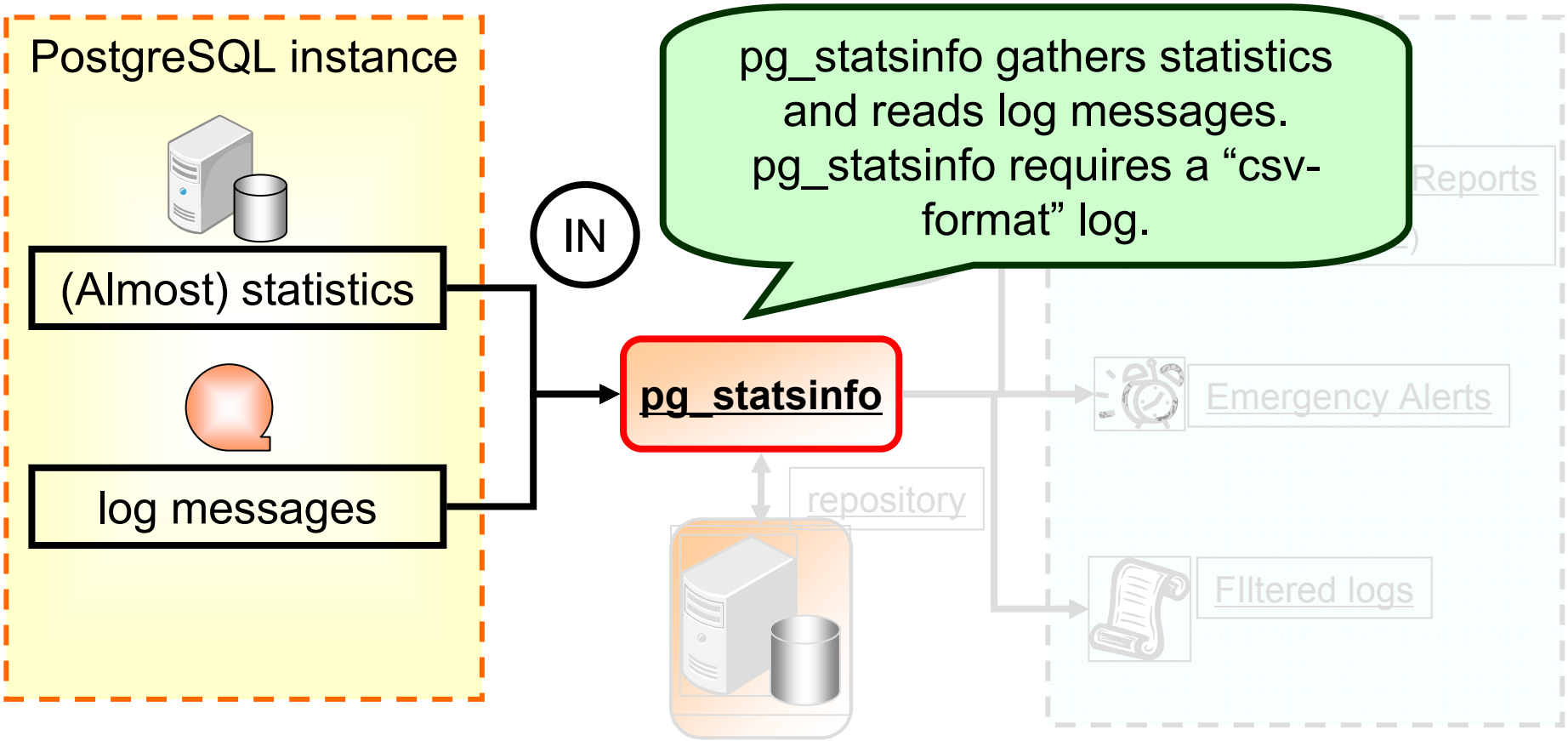


What's the pg_statsinfo ?
- Over view -

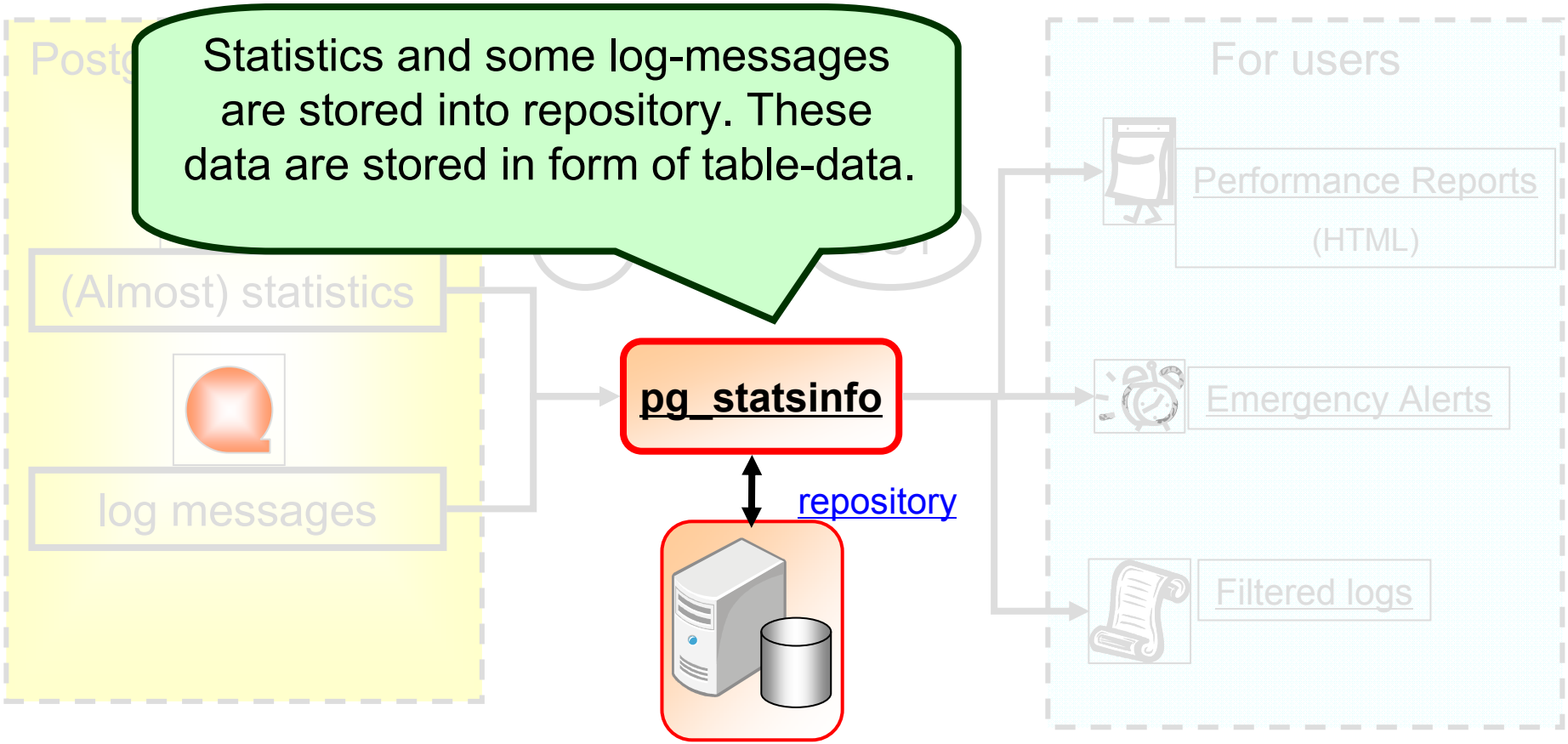
overall view



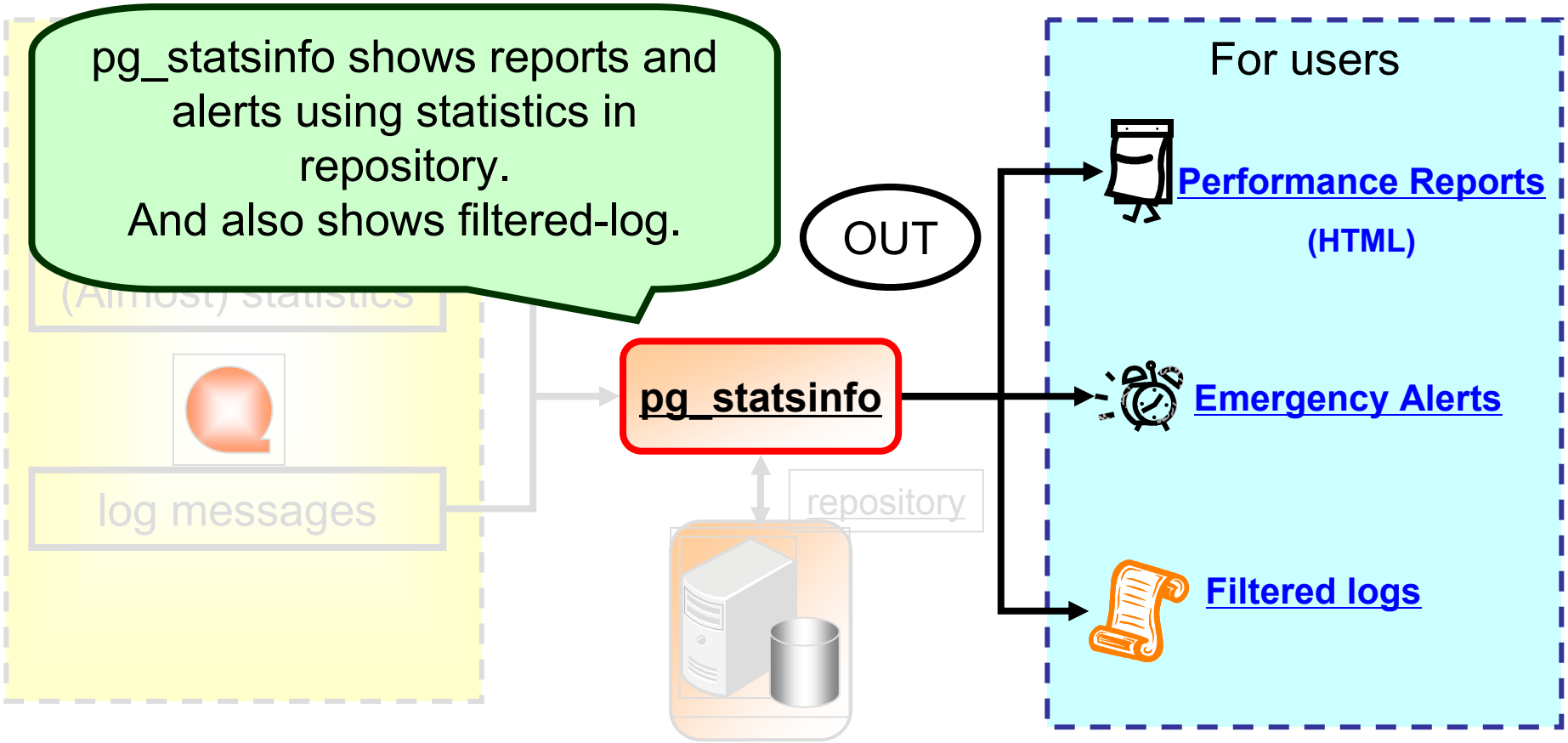
overall view – input sources -



overall view - intermediate info -



overall view – outputs -



Output of the pg_statsinfo
- Alert and Report -

Alert

NTT Open Source Software Center

- Alert is written to log as log-message
  - At present, pg_statsinfo show DBAs brief alert messages
  - DBAs can see the danger without analysis

```
ALERT : DB “test” garbage ratio exceeds threshold : 30%, actually : 32%
ALERT : rollbacks/s in “test” instance exceeds threshold : 20, actually : 121
```

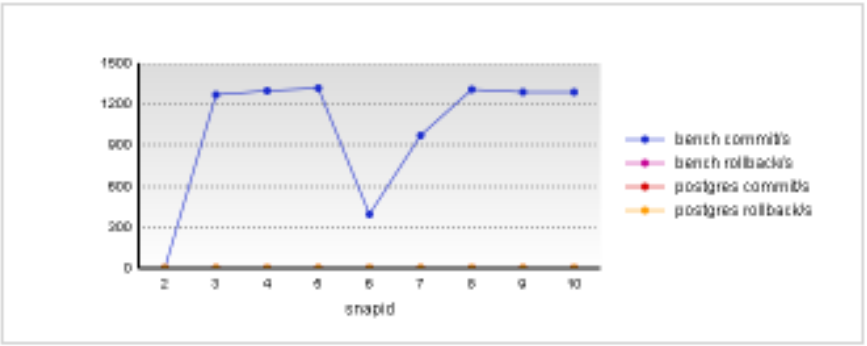
Report

- a part of HTML file

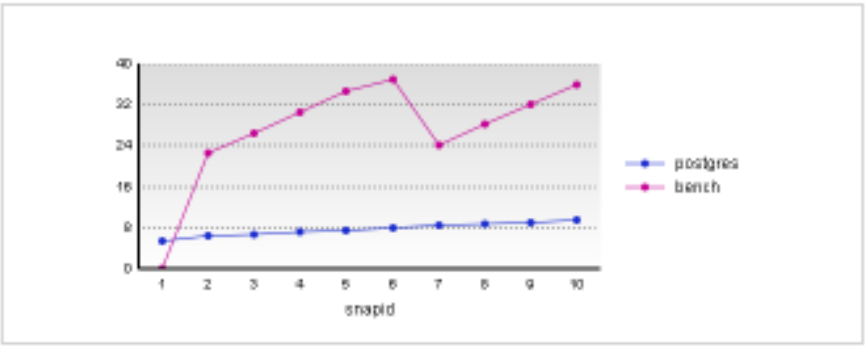
Database Statistics

ID	database	MB	+MB	commit/s	rollback/s	hit%	gets/s	reads/s	rows/s
1	postgres	9	4	0.952	0.005	99.400	540.826	3.422	908.826
2	bench	35	35	1012.830	0.005	99.900	23343.504	23.109	9182.790

Transaction Statistics



Database Size



Report - Instance summary -

- We can check instance whole activity

Summary

name	54546385018968368 59
hostname	postgres.linux
port	5432
pg_version	9.0devel
snapshot begin	2010-03-31 12:45:00
snapshot end	2010-03-31 12:55:00
snapshot duration	00:10:00
total database size	9801 kB
total commits	571
total rollbacks	3

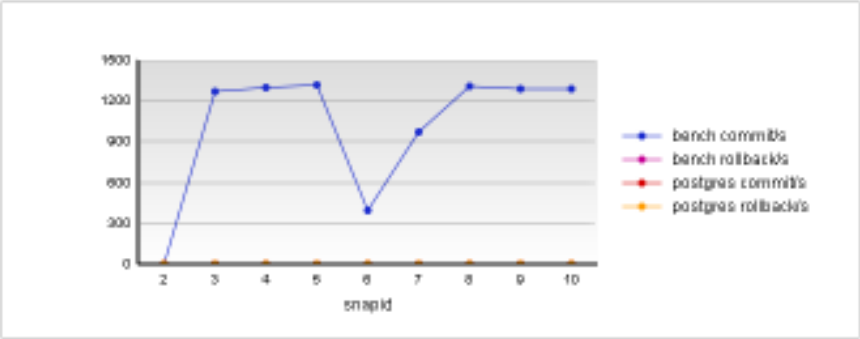
Report - Database Statistics -

NTT Open Source Software Center

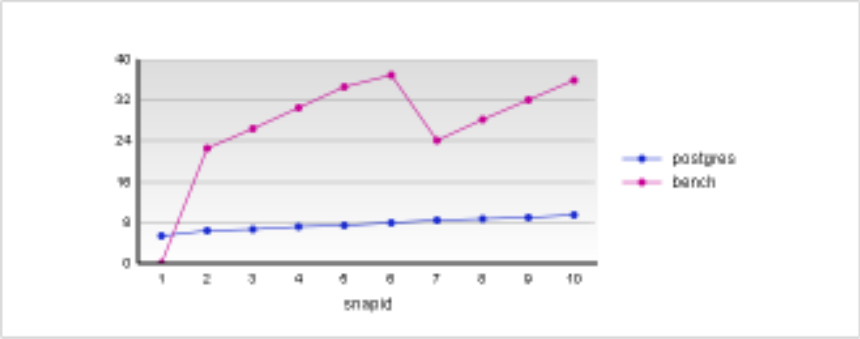
Database Statistics

ID	database	MB	+MB	commit/s	rollback/s	hit%	gets/s	reads/s	rows/s
1	postgres	9	4	0.952	0.005	99.400	540.826	3.422	908.826
2	bench	35	35	1012.830	0.005	99.900	23343.504	23.109	9182.790

Transaction Statistics



Database Size



Report - Disk Usage -

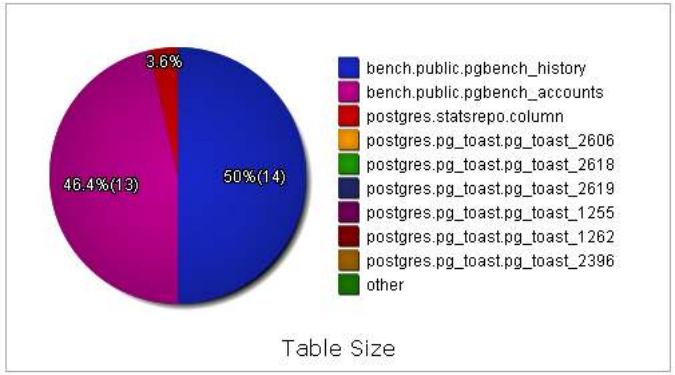
Disk Usage

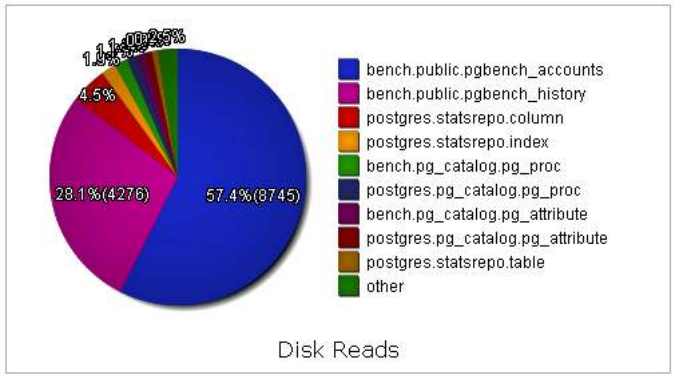
Disk Usage per Tablespace

ID	tablespace	location	device	used (MB)	avail (MB)	remain%
1	pg_default	/home/postgres/pgdata/head	253:0	9532	264162	96.517
2	pg_global	/home/postgres/pgdata/head	253:0	9532	264162	96.517

Disk Usage per Table

ID	database	schema	table	MB	table reads	index reads	toast reads
1	bench	public	pgbench_accounts	13	7916	829	0
2	bench	public	pgbench_history	14	4276	0	0
3	postgres	statsrepo	column	1	591	98	0
4	postgres	statsrepo	index	0	262	22	0
5	postgres	statsrepo	table	0	102	20	0
6	postgres	statsrepo	statement	0	48	7	0
7	bench	public	pgbench_tellers	0	43	4	0
8	bench	public	pgbench_branches	0	27	4	0
9	postgres	statsrepo	setting	0	8	8	0
10	postgres	statsrepo	function	0	7	7	0





Report - Heavily Updated Tables -

NTT Open Source Software Center

Heavily Updated Tables

ID	database	schema	table	INSERT	UPDATE	DELETE	total	HOT%
1	bench	public	pgbench_accounts	100000	607502	0	707502	99.100
2	bench	public	pgbench_tellers	10	607501	0	607511	100.000
3	bench	public	pgbench_branches	1	607501	0	607502	100.000
4	bench	public	pgbench_history	607499	0	0	607499	
5	postgres	statsrepo	column	10141	0	0	10141	
6	postgres	statsrepo	index	2014	0	0	2014	
7	postgres	statsrepo	table	1453	0	0	1453	
8	postgres	pg_catalog	pg_statistic	114	603	0	717	53.100
9	postgres	pg_catalog	pg_attribute	590	2	23	615	100.000
10	postgres	pg_catalog	pg_depend	494	0	36	530	

Report - Heavily Accessed Tables -

NTT Open Source Software Center

Heavily Accessed Tables

ID	database	schema	table	seq_scan	seq_tup_read	tup_per_seq	hit%
1	bench	public	pgbench_accounts	1	100000	100000.000	99.800
2	postgres	statsrepo	table	15	11865	791.000	99.800
3	postgres	statsrepo	statement	7	1314	187.714	95.400
4	postgres	statsrepo	setting	7	736	105.143	96.600
5	postgres	statsrepo	schema	732	66343	90.633	99.700
6	bench	public	pgbench_tellers	138504	1162700	8.395	100.000
7	postgres	statsrepo	snapshot	10	39	3.900	100.000
8	postgres	statsrepo	database	9	32	3.556	100.000
9	postgres	statsrepo	instance	5	3	0.600	97.400
10	bench	public	pgbench_branches	1181485	607504	0.514	100.000

Report - Autovacuum Activity -

NTT Open Source Software Center

Autovacuum Activity

ID	database	schema	table	count	avg index scans	avg removed rows	avg remain rows	avg duration (sec)	max duration (sec)
1	postgres	pg_catalog	pg_statistic	1	1.000	23.000	448.000	0.040	0.040
2	bench	public	pgbench_tellers	8	0.750	296.375	15.250	0.005	0.020
3	bench	public	pgbench_branches	8	0.750	73.750	7.875	0.002	0.020

Report - Statements -

NTT Open Source Software Center

Statements

ID	user	database	query	calls	total time (sec)	time/call (sec)
1	postgres	bench	UPDATE pgbench_branches SET bbalance = bbalance + $1 WHERE bid = $2;	607630	543.502	0.001
2	postgres	bench	UPDATE pgbench_tellers SET tbalance = tbalance + $1 WHERE tid = $2;	607631	137.384	0.000
3	postgres	bench	UPDATE pgbench_accounts SET abalance = abalance + $1 WHERE aid = $2;	607633	30.323	0.000
4	postgres	bench	SELECT abalance FROM pgbench_accounts WHERE aid = $1;	607631	10.180	0.000
5	postgres	bench	INSERT INTO pgbench_history (tid, bid, aid, delta, mtime) VALUES ($1, $2, $3, $4, CURRENT_TIMESTAMP);	607627	7.097	0.000
6	postgres	postgres	CREATE DATABASE bench;	1	0.647	0.647
7	postgres	bench	END;	607627	0.367	0.000
8	postgres	bench	BEGIN;	607634	0.346	0.000
9	postgres	bench	copy pgbench_accounts from stdin	1	0.334	0.334

Input sources of pg_statsinfo
- Statistics and Logs -

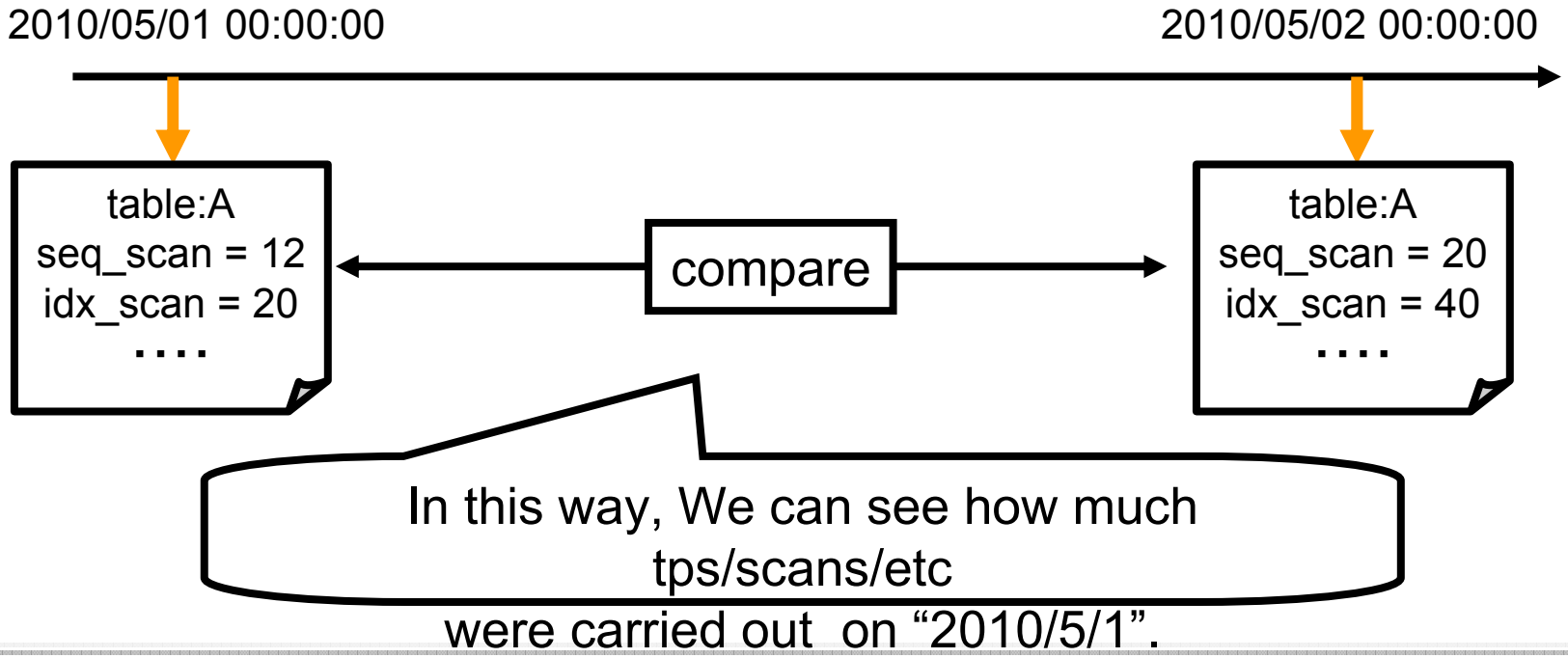
Aspects of statistics

NTT Open Source Software Center

- PostgreSQL outputs many kinds of statistics
- There are two types of statistics
  - accumulated stats
    - e.g.
    - Numbers of commits/rollbacks per database
    - Numbers of seq/index scans, block-read/hit per table
  - Momentary stats
    - e.g.
    - backend process activity (idle or run)
    - locking status
- By these types, a gathering method and viewpoints are different

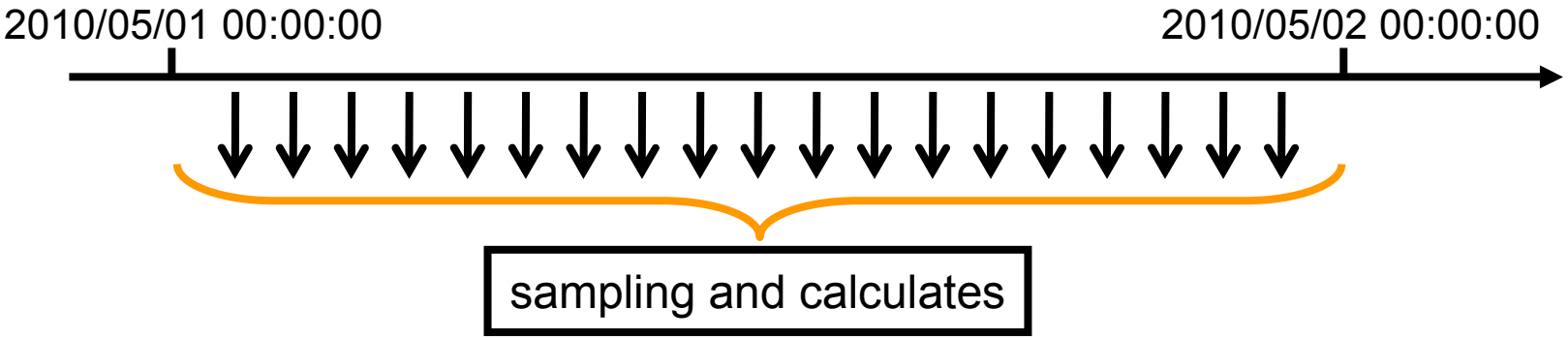
Accumulated stats

- These are gathered periodically
- And use the difference of these statistics that are gathered at beginning and end of the period with the interests.



Momentary stats

- These are gathered periodically (at short intervals)
  - For accuracy enhancement
- And use the occurrences or average of these statistics which are gathered during the period with the interests
  - That's what we call a “sampling”



In this way, We can see how much processes were in status of ‘idel’ or ‘wait’ or ‘run’ on “2010/5/1”.

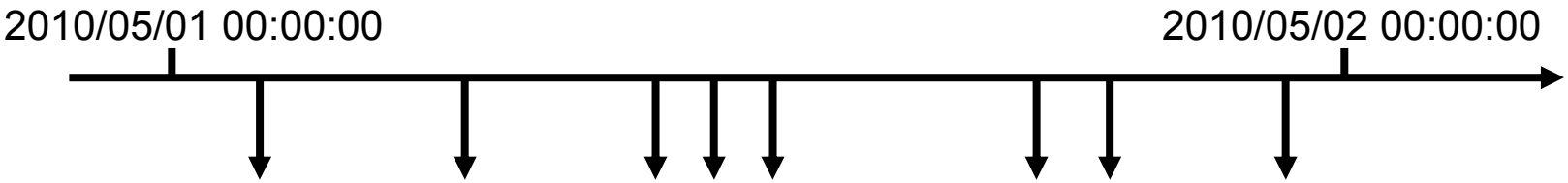
Aspects of Logs

NTT Open Source Software Center

- Log contains various kinds of messages
- At present, there are two types of logs
  - Performance
    - e.g.
    - checkpoint activity
    - autovacuum activity
  - Notification
    - e.g.
    - ERROR logs
    - NOTICE logs

performance

- These are gathered at all times
- And use the occurrence of them which are gathered during the period with the interests
  - In this way, we can retrieve performance info any period



gathering each log messages and counts

We can see how many times did chekcpoint/ autovacuum occur on “2010/5/1”.

Notification

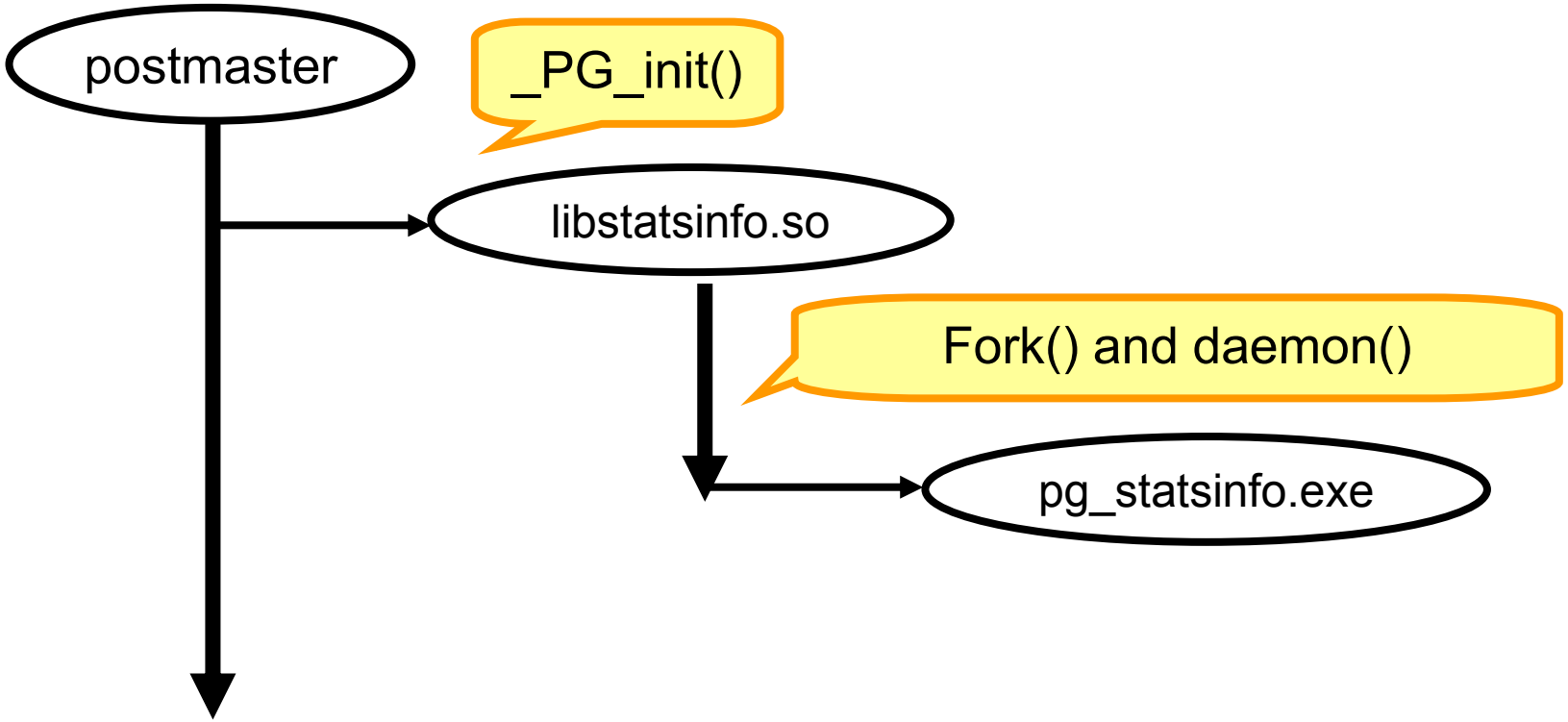
NTT Open Source Software Center

- All of csv log messages are parsed and divided into syslog and textlog
  - Log-level threshold for each log destination
- For example
  - PANIC and FATAL and LOG level messages are important for DBAs and monitoring tools (like SNMP agent)
  - On the other hand, NOTICE level messages are only noise for them
    - But these are necessary for developer
  - Therefore it is happy in some cases to filtering such noise

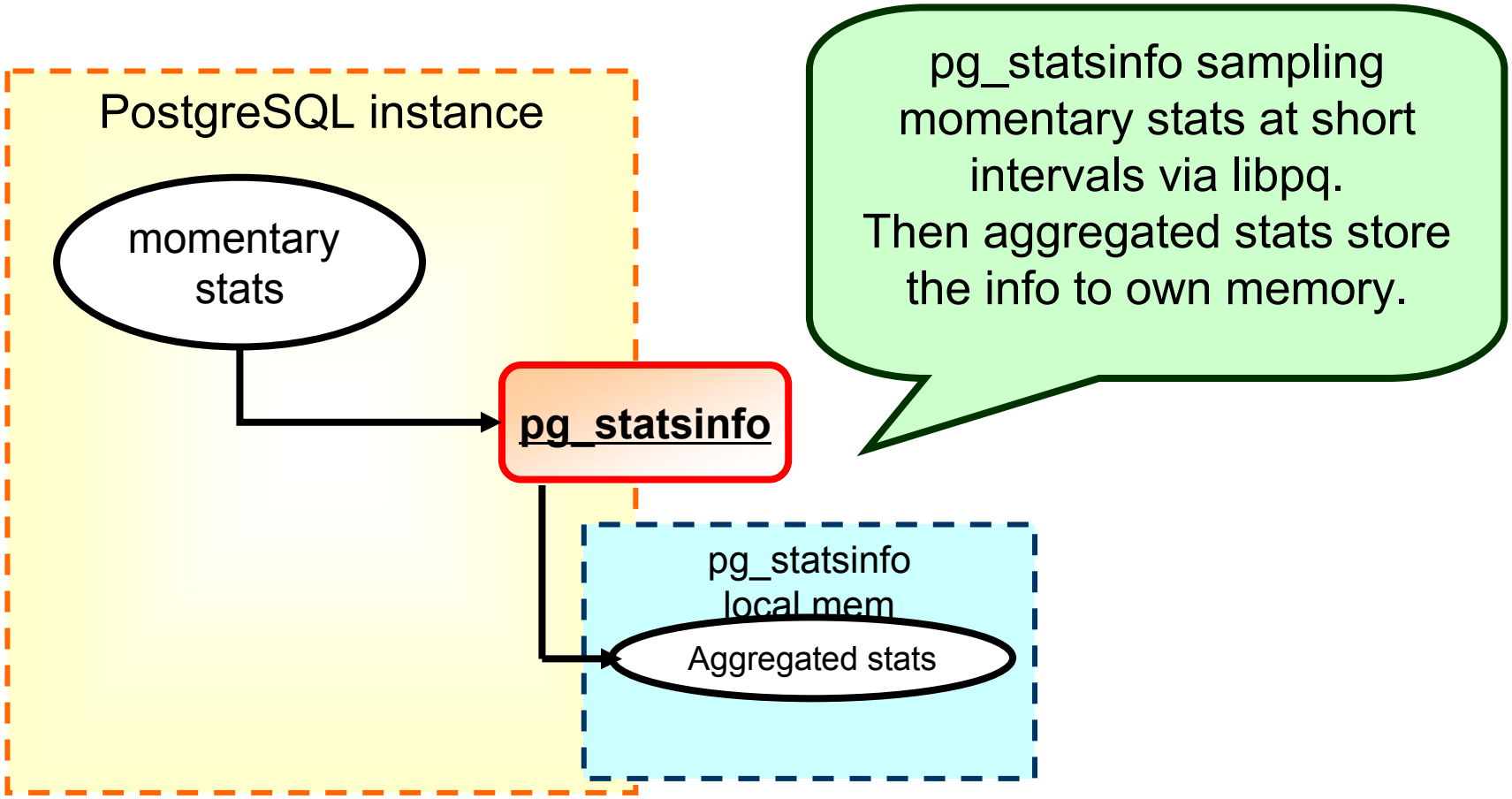
How pg_statsinfo works ?
- More details of pg_statsinfo activities -

Starting pg_statsinfo

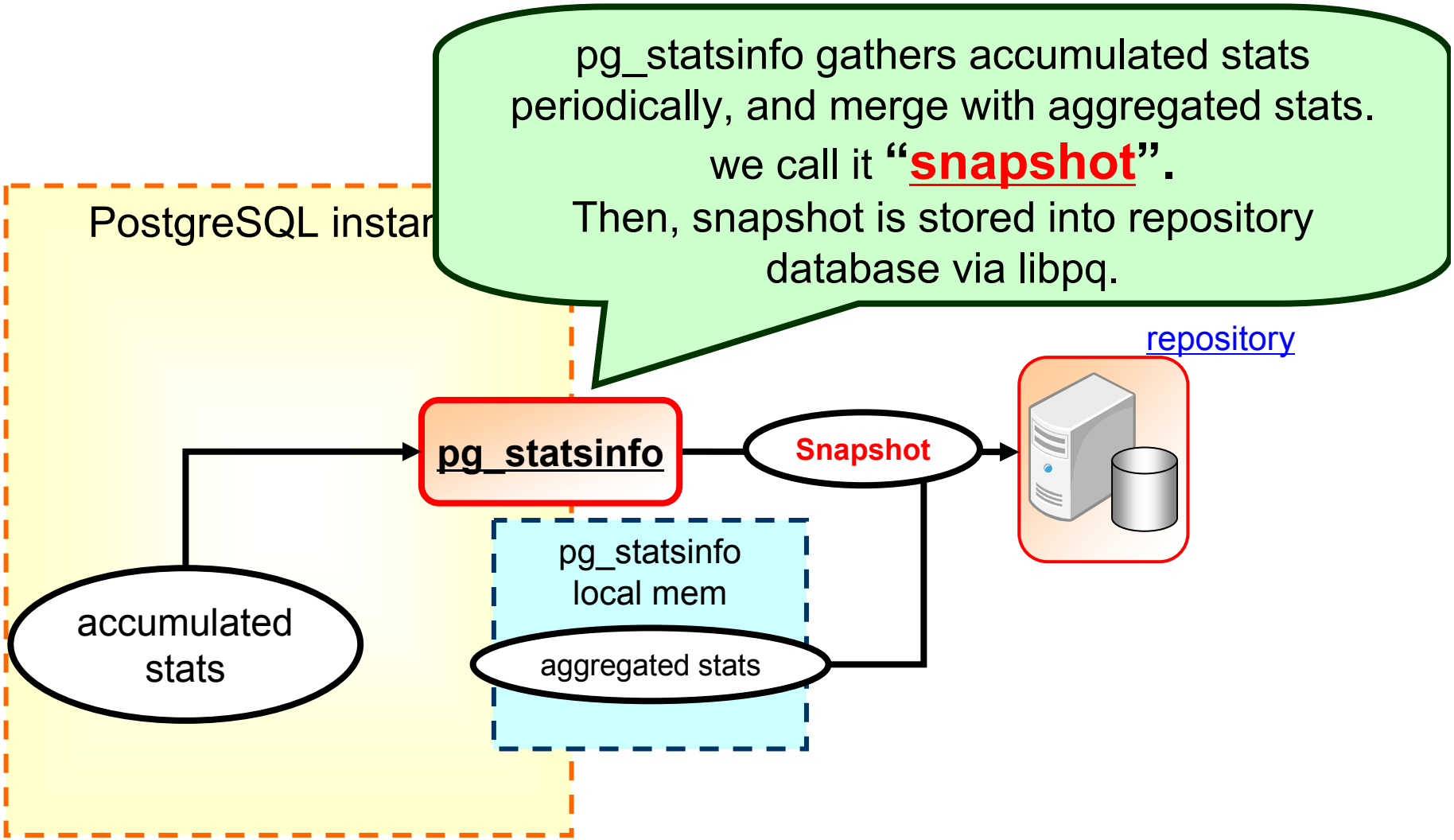
- Users only execute “pg_ctl start” to start pg_statsinfo
  - No additional operation required



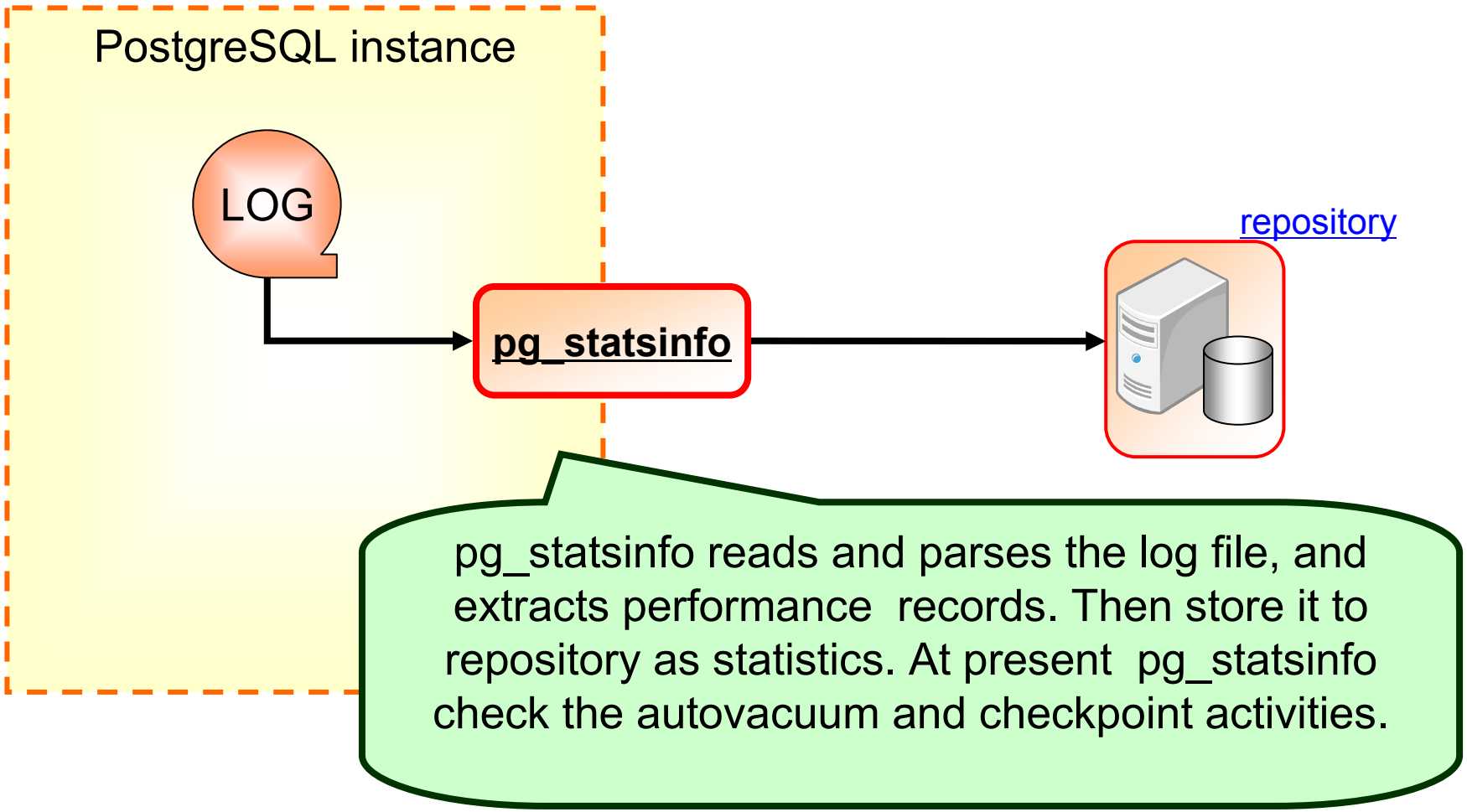
sampling



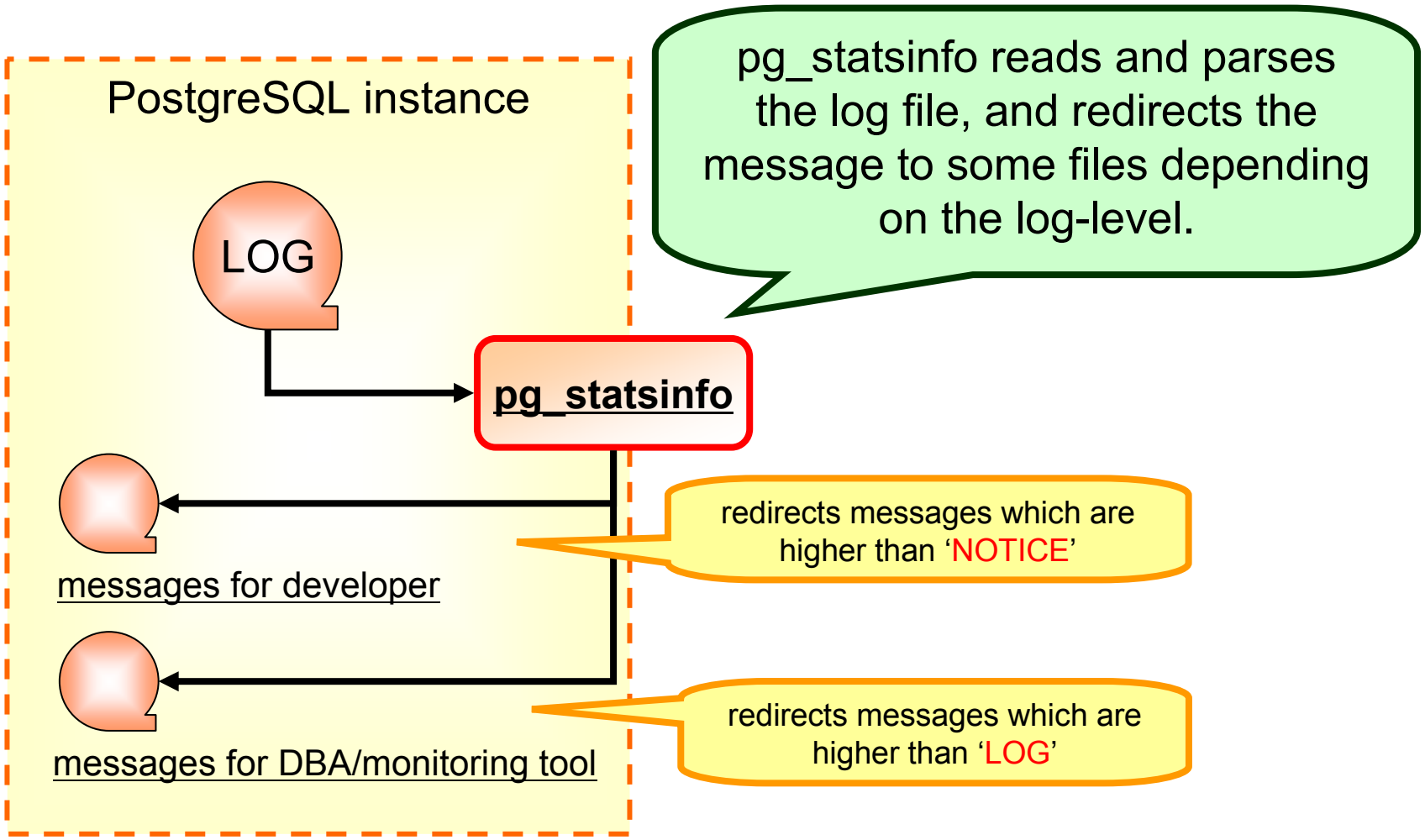
Gathering statistics and storing



Extracting performance information and storing

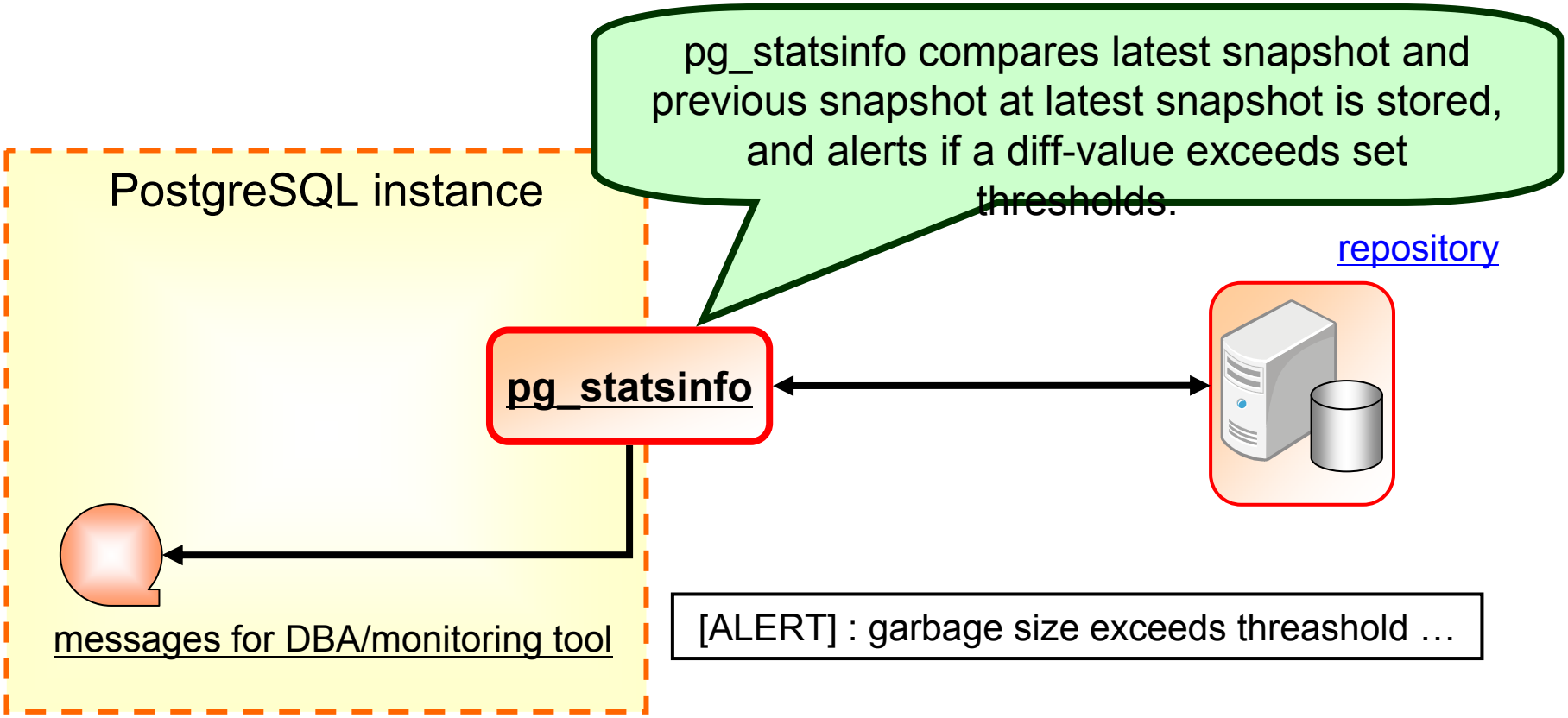


Redirecting log messages



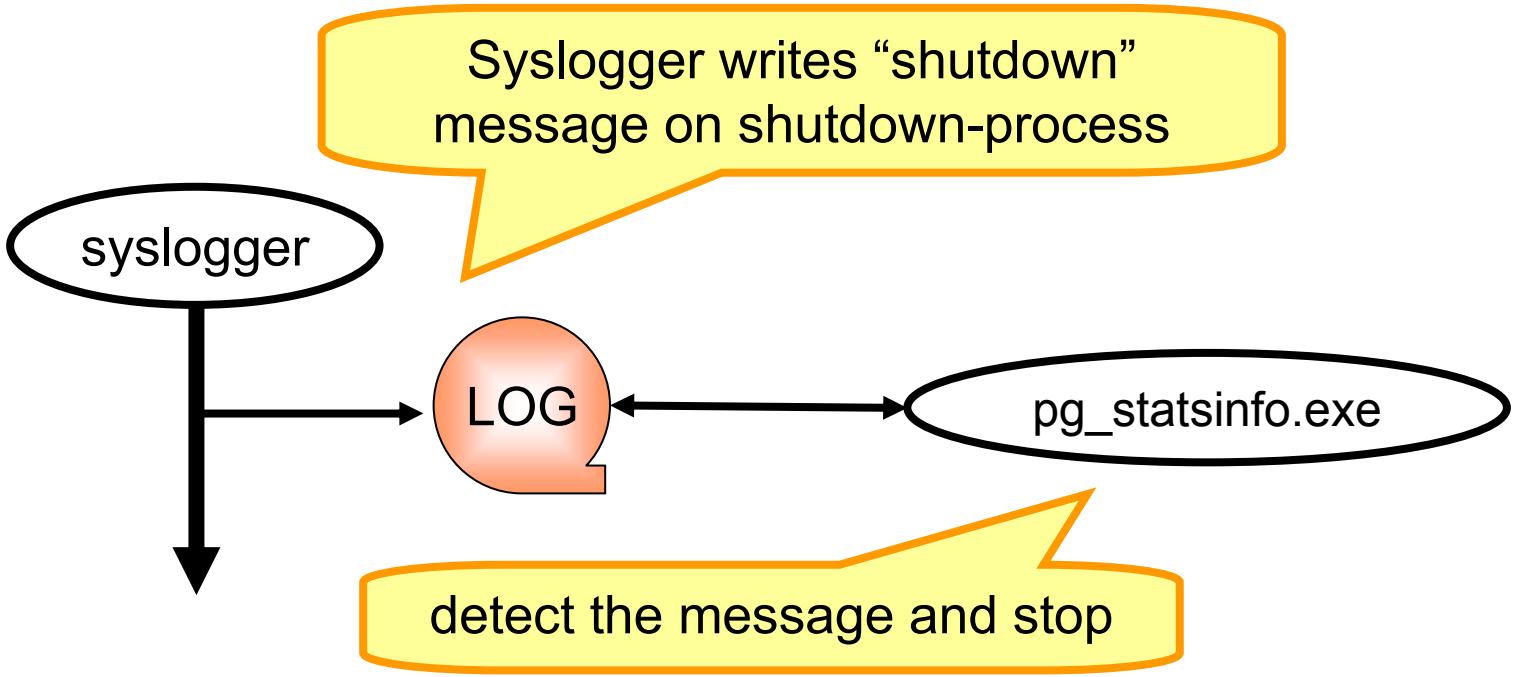
Alert

- An alert function will calculate an delta of snapshots
- Raise alerts as a set of text when abnormal activities are found



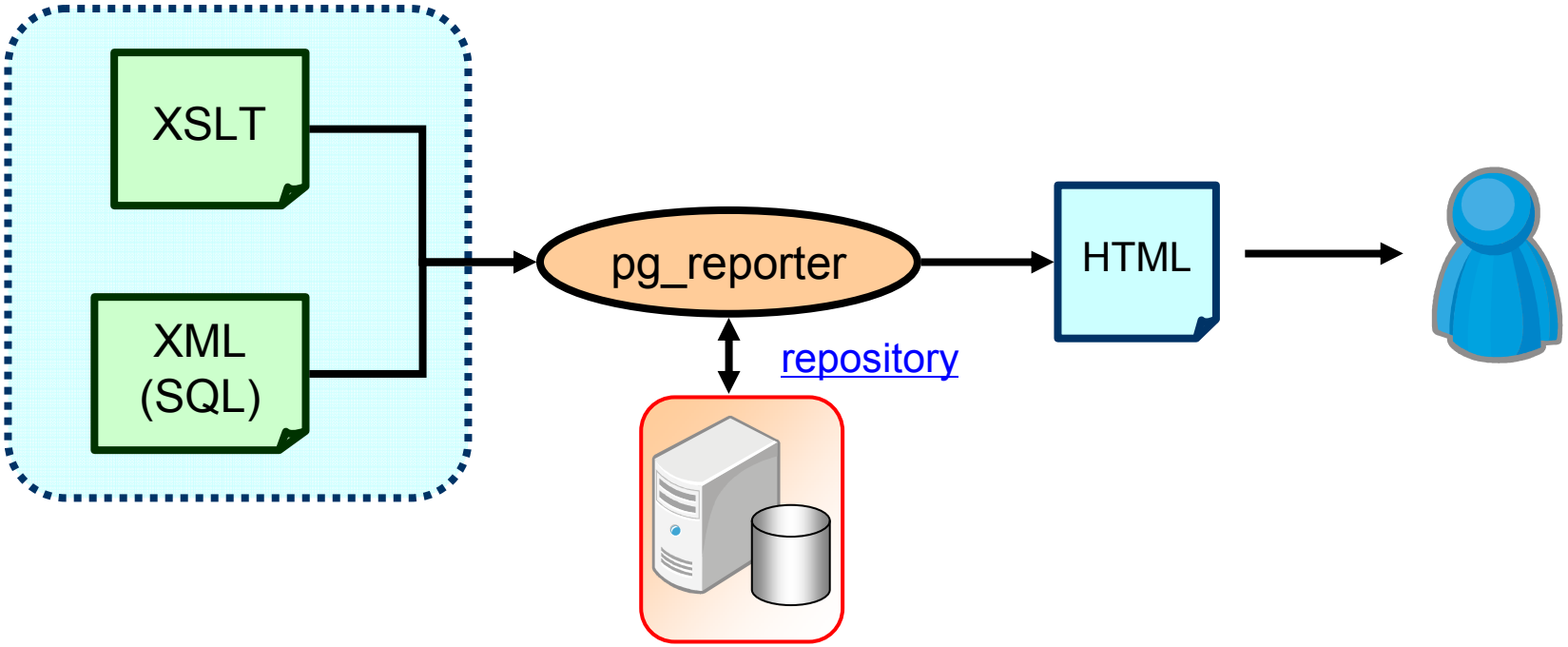
Stopping pg_statsinfo

- Only execute “pg_ctl stop” to stop pg_statsinfo
  - pg_statsinfo has connections to gather statistics, so postmaster will wait smart-shutdown until pg_statsinfo stopping
  - Therefore pg_statsinfo check log-messages to avoid above situation



Report

- Reporter generates reports from statistics stored in repository
  - pg_reporter is a sub-module of pg_statsinfo
  - It is a very important part to show the user-friendly-information



XML file with SQL

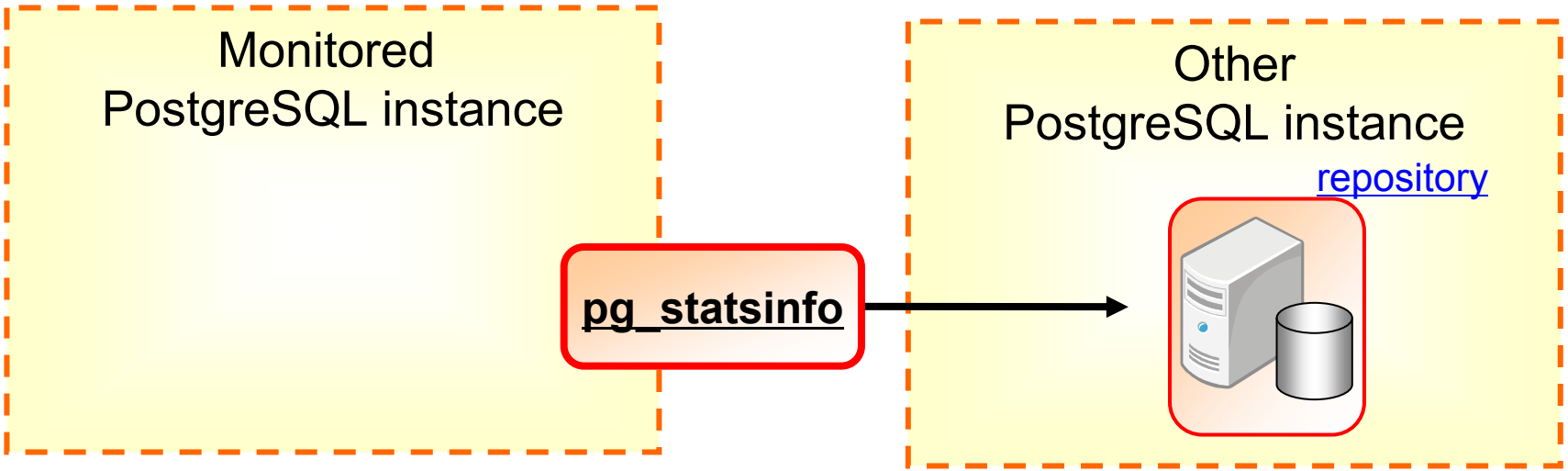
NTT Open Source Software Center

- Users can describe template as “SQL Embedded HTML”
  - Easy to modify the report contents !

```
 . . . .
<h2 id="database" style="clear:both">Database Statistics</h2>
<table><![CDATA[
SELECT ed.name AS "database",
       ed.size / 1024 / 1024 AS "MB",
       (ed.size - sd.size) / 1024 / 1024 AS "+MB",
       statsrepo.tps(
         ed.xact_commit - sd.xact_commit,
         es.time - ss.time) AS "commit/s",
       statsrepo.tps(
         ed.xact_rollback - sd.xact_rollback,
         es.time - ss.time) AS "rollback/s",
       statsrepo.div(
         ed.blks_hit - sd.blks_hit,
         ed.blks_read - sd.blks_read + ed.blks_hit - sd.blks_hit
         ) * 100 AS "hit%",
       statsrepo.tps(
         ed.blks_read - sd.blks_read + ed.blks_hit - sd.blks_hit,
         es.time - ss.time) AS "gets/s",
       statsrepo.tps(
         ed.blks_read - sd.blks_read,
         es.time - ss.time) AS "reads/s",
       statsrepo.tps(
         ed.tup_returned - sd.tup_returned + ed.tup_fetched - sd.tup_fetched,
         es.time - ss.time) AS "rows/s"
  FROM statsrepo.snapshot ss,
 . . . .
```

Repository database

- pg_statsinfo store gathered stats into repository
  - Repository can be in the same database with the monitored instance or in another instance
  - You can retrieve statistics from repository if monitored instance was down



How to use?

Requirements

NTT Open Source Software Center

- OS
  - Linux and Windows !
  - 32bit & 64bit
- PostgreSQL
  - 8.3 ～ 9.0
  - PostgreSQL have to be configured with “--enable-thread-safety” , “--lib-xml” and “--lib-xslt” option
    - pg_statsinfo is a “multi-thread application” and uses libpq
    - It also uses XML and XSLT features and to generate reports
- Others
  - Their processing are lightwight
  - We could get snapshot during benchmark-test (TPC-W 1600tps) without influence on performance
  - One snapshot size is about 600 kB

configuration

NTT Open Source Software Center

- Very simple
  - Only describe bellow parameters (for minimum action)
    - shared_preload_libraries = ‘pg_statsinfo’
  - And there are some parameters to control pg_statsinfo
    - These parameters are described in postgresql.conf
    - Use “custom_variable_classes”

configuration

NTT Open Source Software Center

- A part of configurations

Category	Name	Description
snapshot	sampling_interval	sampling interval (def. 5sec)
	snapshot_interval	snapshot interval (def. 5min)
	excluded_dbnames	Unmonitored database names (def. template1/0)
	repository_server	conn-string to connect repository (def. postgres on local)
logging	textlog_filename	text log name (def. postgresql.log)
	textlog_permission	text log permission (def. 0600)
	textlog_min_messages	Minimum message levels for textlog (def. warning)
	syslog_min_messages	Minimum message levels for textlog (def. disable)

*_min_messages follow the ‘log_min_messages’ of PostgreSQL

~ Demo ~

NTT Open Source Software Center

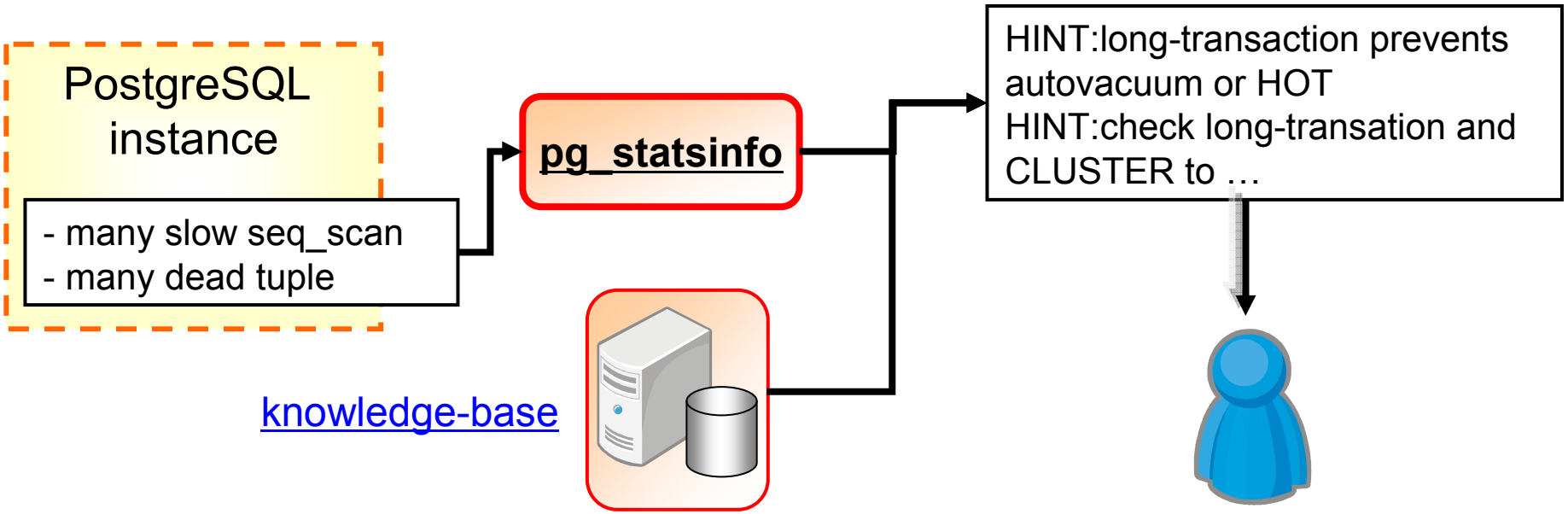
- I will show the brief demo
  - I want to tell you how easy it is to use this product
- Conditions
  - Ubuntu 8.10 & PostgreSQL9.0beta1
  - pg_statsnfo, pg_reporter, and some contrib modules are already installed
- Demo contains following actions
  - Initdb and run PostgreSQL server (and pg_statsinfo)
  - pgbench test
  - Check the database activities shown by report
  - Stop PostgreSQL server (and pg_statsinfo)

NTT Open Source Software Center

Future work
- pg_statsinfo own challenges -

Alert with knowledge-base

- The conditions alerting are different every system
  - pg_statsinfo will be able to show more specific alert messages with each system's own knowledge



Report with OS's resouce statistics

NTT Open Source Software Center

- OS's resources often retrieve existing tools
  - e.g.) sar, vmstat, iostat
- In some cases, troubles in DB-server are caused by products other than DBMS
  - Therefore, check the statistics of PostgreSQL and OS to identify the bottom cause
- This function may be point of reporter improvement

Future work
- Cooperate with PostgreSQL -

Cooperate with PostgreSQL

NTT Open Source Software Center

- There are 3 topics
  - More statistics !
  - More easy to log-filtering !
  - More smart works with PostgreSQL !

More statistics

NTT Open Source Software Center

- Pg_statsinfo wants to get “Top 5 timed event”
- PostgreSQL show us many statistics, but not enough
- Oracle has more advanced statistics collector
  - “Top 5 timed event” is a most watched information by DBAs of Oracle
    - “Top 5 timed event” shows most time elapsed processes
  - It is useful to see a bottle-neck quickly

More statistics

NTT Open Source Software Center

- Dtrace / Systemtap can retrieve sampling-statistic and show us nearly “Top 5 timed event”
  - But..
  - Dtrace / Systemtap are not good at periodical monitoring
    - These modules are often used on development environments, but are few used on commercially environments
    - We have to prepare scripts for sampling
    - And also have to follow these products updating …
  - So, it might be the best way that PostgreSQL has own sampling system
    - Of course, it is very difficult ..

More statistics !

NTT Open Source Software Center

- Pg_statsinfo also wants to get each backend memory consumptions
- Some objects (prepared statement / Cursor) are not release until disconnect or CLOSE.
  - Sometime users encounter the swap caused by memory consumption of these objects
  - It is useful to see the memory consumption for each backends
  - It is also useful to see how large sort/hash are performed each backends

More easy to log-filtering !

NTT Open Source Software Center

- pg_statsinfo wants the editable log structure on memory
  - Such log structure make pg_statsinfo reading and parsing log-messages easily
  - It is also useful to modify the log-level/code/message by purpose
    - We can set, for example, not LOG but PERF level about performance-messages
    - We can change log-level which suits to each system

More smart works with PostgreSQL !

NTT Open Source Software Center

- pg_statsinfo uses PG_init to be kicked by postmaster, and checks log-messages to shutdown with postmaster
  - It is a not smart way
    - If only pg_statsinfo process was down, there are no-way to restart pg_statsinfo
    - pg_statsinfo has some restrictions for handling log-locale, log-timezone, and so on
      - External modules can not use these configurations well
  - Such pg_statsinfo activities should be controlled by the postmaster not external process

More smart works with PostgreSQL !

NTT Open Source Software Center

- A similar topic was desicussed on Hackes-ML
  - “[HACKERS] scheduler in core”
  - Following proposal is pretty much the requirement of pg_statsinfo !

> Simon Riggs wrote:
> (snip)
> integrated_user_processes = 'x, y, z' would run x(), y() and z()
> in their own processes.
>
> These would execute after startup, or at consistent point in recovery.
> The code for these would come from preload_libraries etc.
>
> They would not block smart shutdown, though their shudown sequence
> might delay it. User code would be executed last at startup and first
> thing at shutdown. API would be user_process_startup(), user_process_shutdown().

Conclusion

NTT Open Source Software Center

- pg_statsinfo is useful to see and check problems and causes
  - pg_statsinfo can show DBAs the Alert, Report, Filtered log
  - DBAs only have to check it
- And very easy to use it
  - Instllation and configuration, and starting are simple
  - It runs as a PostgreSQL-native-backend
- But still not get enough information for DBAs
  - Some works need for pg_statsinfo and PostgreSQL
- This products will never fail to PostgreSQL DBAs !
  - Please let’s try it

references

NTT Open Source Software Center

- If someone wants to see more detail / specific information about pg_statsinfo, please check the following site
  - http://pgstatsinfo.projects.postgresql.org/index.html
- Following products have a same aim of pg_statsinfo
  - Please check these products too
  - Posuta
    - http://code.google.com/p/posuta/
  - pg_statspack
    - http://pgfoundry.org/projects/pgstatspack/

- Any questions?

- Thank you for your kind attention !