

■pg\_statsinfoでの取得情報とpostgres.expoterの取得情報の比較

※2015年の成果物を元に比較。当時のpg\_statsinfoのバージョンは3.0.2、2019/1時点での最新版は10(3.1→3.2→3.3→10)。

※2015年以降の大きなエンハンス項目(3.3→10)にはロジカルレプリケーションの対応があるが、いずれもpostgres.expoterでは未対応(pg\_publication、pg\_publication\_\*から情報取得していない)

※postgres.expoterは2019/1/30時点の版で机上検証、実機検証を実施。システムカタログ(ビュー)の直接参照を基本機能とし、カスタムクエリ機能で任意の情報取得できる考え方。

※DB全体に関する情報はビルトインとして取得可能(cmd/postgres.expoter/go)。その他の情報はqueries.yamlにカスタムクエリに記載して取得可能。

※カスタムクエリのサンプルとしてpg\_stat\_user\_tables、pg\_statio\_user\_tablesの例が既に記載されており、情報取得が可能。

機能大項目	機能中項目	機能項目	pg_statsinfo	データの取得元	postgres.expoter	wmi.expoter	typeperf	ビルトイン(○) or カスタムクエリ(●)	データの取得元	備考	
DB全体	データベース	容量	○	pg_database_size(oid)	○	-	-	●	pg_database_size(pg_database_datname)		
		キャッシュヒット	○	pg_stat_get_db_blocks_fetched(oid)	△	-	-	○	pg_stat_database_bkrs_read	pg_stat_database_bkrs_read、pg_stat_database_bkrs_hitより計算	
		readseq行数	○	pg_stat_get_db_tuples_returned(oid)	○	-	-	○	pg_stat_database		
		read(index)行数	○	pg_stat_get_db_tuples_fetched(oid)	○	-	-	○	pg_stat_database.tup_fetched		
		insert行数	○	pg_stat_get_db_tuples_inserted(oid)	○	-	-	○	pg_stat_database.tup_inserted		
		update行数	○	pg_stat_get_db_tuples_updated(oid)	○	-	-	○	pg_stat_database.tup_updated		
		delete行数	○	pg_stat_get_db_tuples_deleted(oid)	○	-	-	○	pg_stat_database.tup_deleted		
		アーカイブ	アーカイブ取得数	○	pg_stat_archiver.archived_count	x	-	-			
		アーカイブ失敗数	○	pg_stat_archiver.failed_count	x	-	-				
	autovacuum	回収ページ数	○	autovacuum log の "pages: %d removed"	x	-	-				
		回収タプル行数	○	autovacuum log の "tuples: %d removed"	x	-	-				
		実行時間	○	autovacuum log の "elapsed %f sec"	x	-	-				
	autoanalyze	実行時間	○	autoanalyze log の "elapsed %f sec"	x	-	-				
	SQL	遅いquery	△		○	-	-	○			
	lock	デッドロック数	○	pg_stat_get_db_deadlocks(oid)	○	-	-	○	pg_stat_database.deadlocks		
		ロック取得クエリ	○	pg_stat_activity.query	x	-	-			pg_stat_activityの情報自体は取っている。 pg_stat_activityを参照させるための非特権ユーザ、参照用ビューの作成手番がREADMEに記載されている。	
		ロック待ちクエリ	○	statsinfo.last_xact_activity()	x	-	-				
		ロック待ち時間	○	pg_stat_activity.query_start	x	-	-			pg_stat_activityの情報自体は取っている。 pg_stat_activityを参照させるための非特権ユーザ、参照用ビューの作成手番がREADMEに記載されている。	
		レプリケーション	遅延	○	pg_stat_replication.*, 各種関数	○	-	-	○	pg_stat_replication.*	pg_stat_replicationを参照させるための非特権ユーザ、参照用ビューの作成手番がREADMEに記載されている。
		WAL	書き込み量	○	pg_current_xlog_location	○	-	-	○	pg_stat_replication.pg_current_xlog_location	pg_stat_replicationを参照させるための非特権ユーザ、参照用ビューの作成手番がREADMEに記載されている。
		トランザクション	トランザクション数	○	are(pg_database.datfrozenxid)	x	-	-			
			コミット数	○	pg_stat_get_db_xact_commit(oid)	○	-	-	○	pg_stat_database.xact_commit	
			ロールバック数	○	pg_stat_get_db_xact_rollback(oid)	○	-	-	○	pg_stat_database.xact_rollback	
		チェックポイント	トランザクション時間	○	pgstat_fetch_stat_beentry	x	-	-			
	開始日時		○	checkpoint starting log の タイムスタンプ	x	-	-				
	処理時間		○	checkpoint complete log の "total=%ld%03d s"	x	-	-			pg_stat_bgwriter.checkpoint_write_time、 pg_stat_bgwriter.checkpoint_sync_timeは取得しているので、統計情報を記録し始めてからの総時間は算出可能	
		書き込み量	○	checkpoint complete log の "wrote %d buffers"	x	-	-			pg_stat_bgwriter.buffers_checkpoint、pg_stat_bgwriter.buffers_cleanは取得しているので、統計情報を記録し始めてからの総量は算出可能	
	ソート処理	メモリ使用量	x		x	-	-				
		ディスクソート	x		x	-	-				
	待機イベント	待機イベント	x		x	-	-				
各種設定変更の推奨情報	各種	x		x	-	-					
DBオブジェクト	テーブル	行数	○	pg_class.reltuples	x	-	-				
		容量	○	pg_relation_size(oid)	x	-	-				
		readseq回数	○	pg_stat_get_numscans(oid)	○	-	-	●	pg_stat_user_tables.seq_scan		
		read(index)回数	○	pg_stat_get_numscans(oid)	○	-	-	●	pg_stat_user_tables.idx_scan		
		キャッシュヒット(seq)	○	pg_stat_get_blocks_fetched(oid)、Pg_stat_get_blocks_hit(oid)	△	-	-	●	pg_statio_user_tables.heap_bkrs_read、pg_statio_user_tables.heap_bkrs_hit	pg_statio_user_tables.heap_bkrs_read、pg_statio_user_tables.heap_bkrs_hitより計算	
		キャッシュヒット(index)	○	pg_stat_get_blocks_fetched(oid)、Pg_stat_get_blocks_hit(oid)	△	-	-	●	pg_statio_user_tables.idx_bkrs_read、pg_statio_user_tables.idx_bkrs_hit	pg_statio_user_tables.idx_bkrs_read、pg_statio_user_tables.idx_bkrs_hitより計算	
		TOAST キャッシュヒット (seq)	○	pg_stat_get_blocks_fetched(oid)、Pg_stat_get_blocks_hit(oid)	△	-	-	●	pg_statio_user_tables.toast_bkrs_read、pg_statio_user_tables.toast_bkrs_hit	pg_statio_user_tables.toast_bkrs_read、pg_statio_user_tables.toast_bkrs_hitより計算	
		TOAST キャッシュヒット (index)	○	pg_stat_get_blocks_fetched(oid)、Pg_stat_get_blocks_hit(oid)	△	-	-	●	pg_statio_user_tables.tidx_bkrs_read、pg_statio_user_tables.tidx_bkrs_hit	pg_statio_user_tables.tidx_bkrs_read、pg_statio_user_tables.tidx_bkrs_hitより計算	
		insert行数	○	pg_stat_get_tuples_inserted(oid)	○	-	-	●	pg_stat_user_tables.n_tup_ins		
		update行数	○	pg_stat_get_tuples_updated(oid)	○	-	-	●	pg_stat_user_tables.n_tup_upd		
	HOT update行数	○	pg_stat_get_tuples_hot_updated(oid)	○	-	-	●	pg_stat_user_tables.n_tup_hot_upd			
	delete行数	○	pg_stat_get_tuples_deleted(oid)	○	-	-	●	pg_stat_user_tables.n_tup_del			
	最終 vacuum 日時	○	pg_stat_get_last_vacuum_time(oid)	○	-	-	●	pg_stat_user_tables.last_vacuum			
	最終 analyze 日時	○	pg_stat_get_last_analyze_time(oid)	○	-	-	●	pg_stat_user_tables.last_analyze			
	無効vacuum 対象行数	○	pg_stat_get_dead_tuples(oid)	○	-	-	●	pg_stat_user_tables.dead_tup			
	カラムと物理配置の相関	○	pg_statio_statsnumbersN	x	-	-					
	容量	○	pg_relation_size(oid)	x	-	-					
	索引	index scan 実行回数	○	pg_stat_get_numscans(oid)	x	-	-				
		キャッシュヒット	○	pg_stat_get_tuples_fetched(oid)、Pg_stat_get_blocks_hit(oid)	x	-	-				
	関数	実行回数	○	pg_stat_user_functions.calls	x	-	-				
	総処理時間	○	pg_stat_user_functions.total_time	x	-	-					
テーブルスペース	使用率	○	statsfs()	x	-	-					
OS全体	CPU使用量	idle	○	/proc配下の情報	-	○	○				
		iowait	○	/proc配下の情報	-	x	x				
		system	○	/proc配下の情報	-	x	x				
		user	○	/proc配下の情報	-	x	x				
		Load Average	○	/proc配下の情報	-	x	x				
	メモリ使用量	memfree	○	/proc配下の情報	-	○	○				
		buffers	○	/proc配下の情報	-	x	x				
		cached	○	/proc配下の情報	-	x	○				
		swap	○	/proc配下の情報	-	x	x				
		dirty	○	/proc配下の情報	-	x	○				
Disk/O	IO size	○	/proc配下の情報	-	○	x					
	IO time	○	/proc配下の情報	-	x	x					
	ディスク利用状況	○	/proc配下の情報	-	○	○					