Extended TOAST Pointer

#pluggable_toast #custom_pointer #wide_toast_pointer

Oleg Bartunov, Teodor Sigaev, Nikita Glukhov, Nikita Malakhov

Vanilla TOAST Pointer:

- One TOAST table for all table attributes
- One TOAST implementation for all data types
- Implicit data amount limit applied by 32-bit TOAST value_ID: total 4 294 967 296 value IDs in TOAST table limits total rows with toasted attributes in main table. Exceeding this implicit limit (using all possible value IDs) results in server falling into endless cycle of re-checking value ids in search for the free one.

TOAST Pointer

| va_header	1 byte
va_tag	1 byte
va_rawsize	4 bytes
va_extinfo	4 bytes
va_valueid	4 bytes
va_toastrelid	4 bytes

Extended TOAST Pointer:

- Wide TOAST pointer with 64-bit TOAST value_id - removes implicit data amount limit
- Ability to use multiple TOAST tables for relation, up to imitation of columnar storage with the default Heap engine on demand
- Pluggable TOAST, inspired by Pluggable Storage: multiple (different) TOAST implementations for columns and data types

Wide TOAST Pointer

va_header	1 byte
va_tag	1 byte
va_rawsize	4 bytes
va_extinfo	4 bytes
va_valueid	8 bytes
va_toastrelid	4 bytes

Custom TOAST Pointer

va_header	1 byte
va_tag	1 byte
va_rawsize	4 bytes
va_tptrdatalen	4 bytes
va_tptrdata	X bytes

Table / TOAST Table
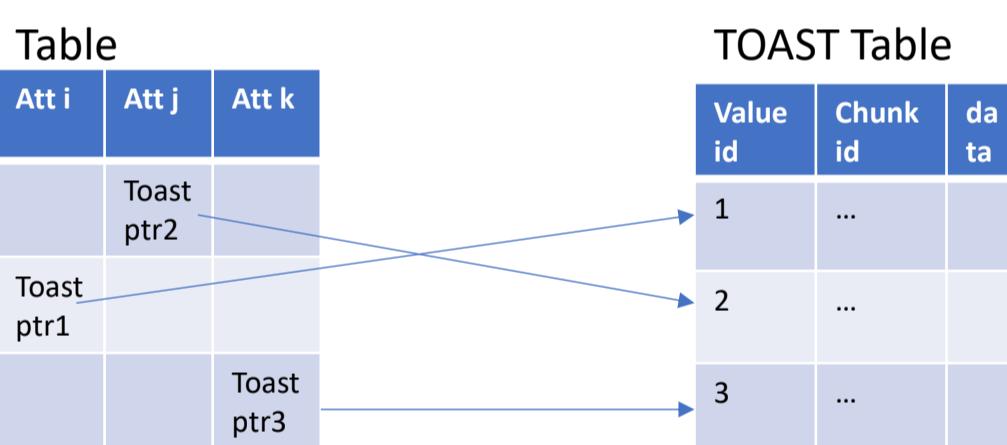
Discussions:

Extending the TOAST Pointer
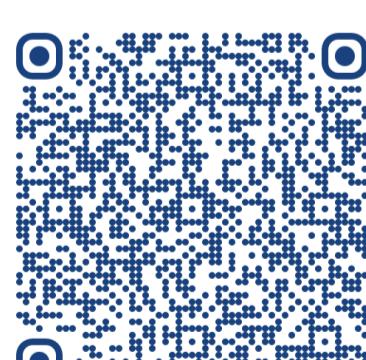
64-bit TOAST Value Id
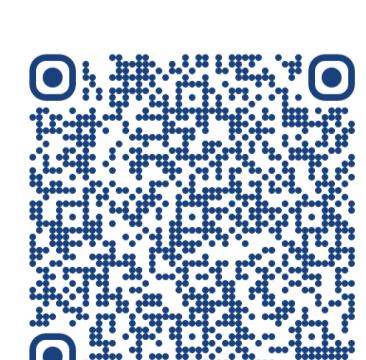
Pluggable TOAST

Collaboration wanted!
1. Overall design review needed
2. Logical replication engine revision required
3. Further TOAST improvements to be suggested

Talks:

PGConf.dev 2025