NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Postgis “esri union” on a “unlimited” sized data set

NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

What is “Esri Union”

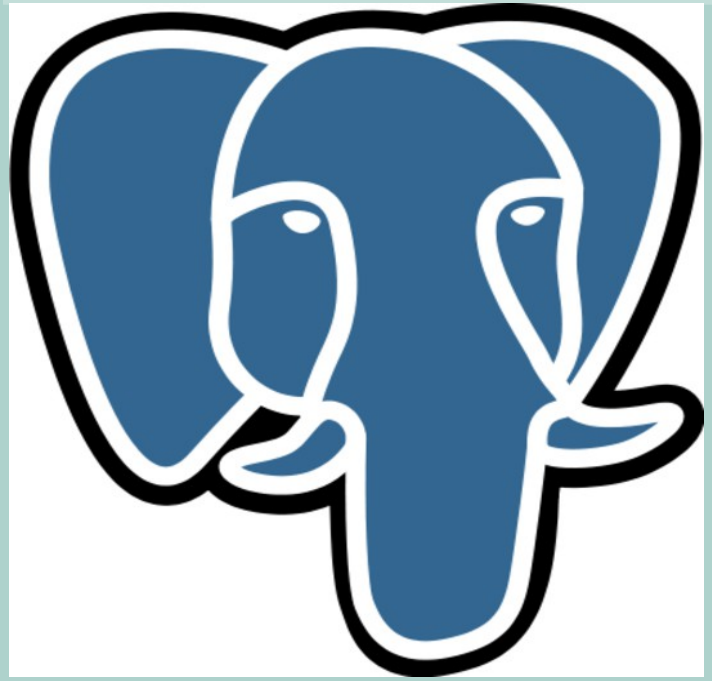
NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

The Norwegian Institute of Bioeconomy Research

The goal of NIBIO is Contribute to food security, sustainable resource management, innovation and value creation through research and knowledge production.

www.nibio.no

NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH



Postgis
Simple Feature

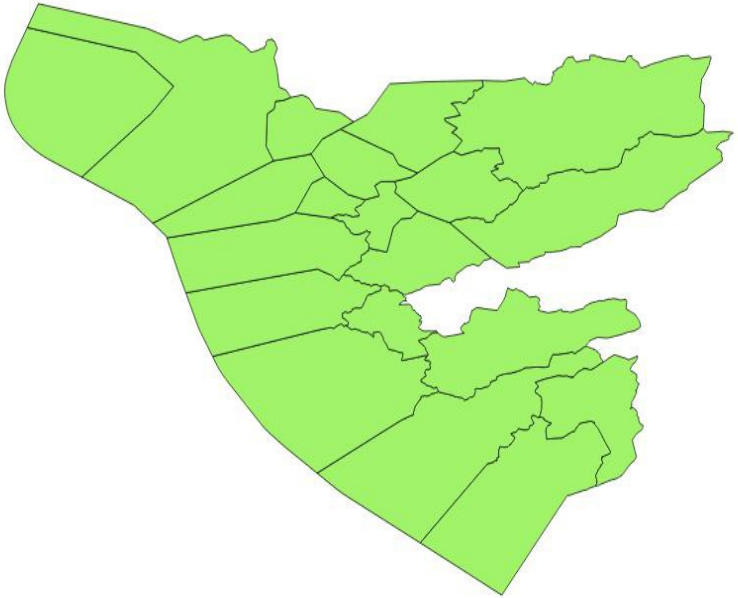
NIBIO

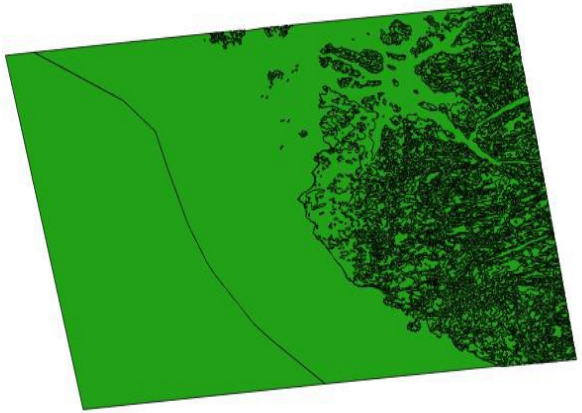
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Why do presentations like this ?

Lars Aksel Opsahl, developer at NIBIO

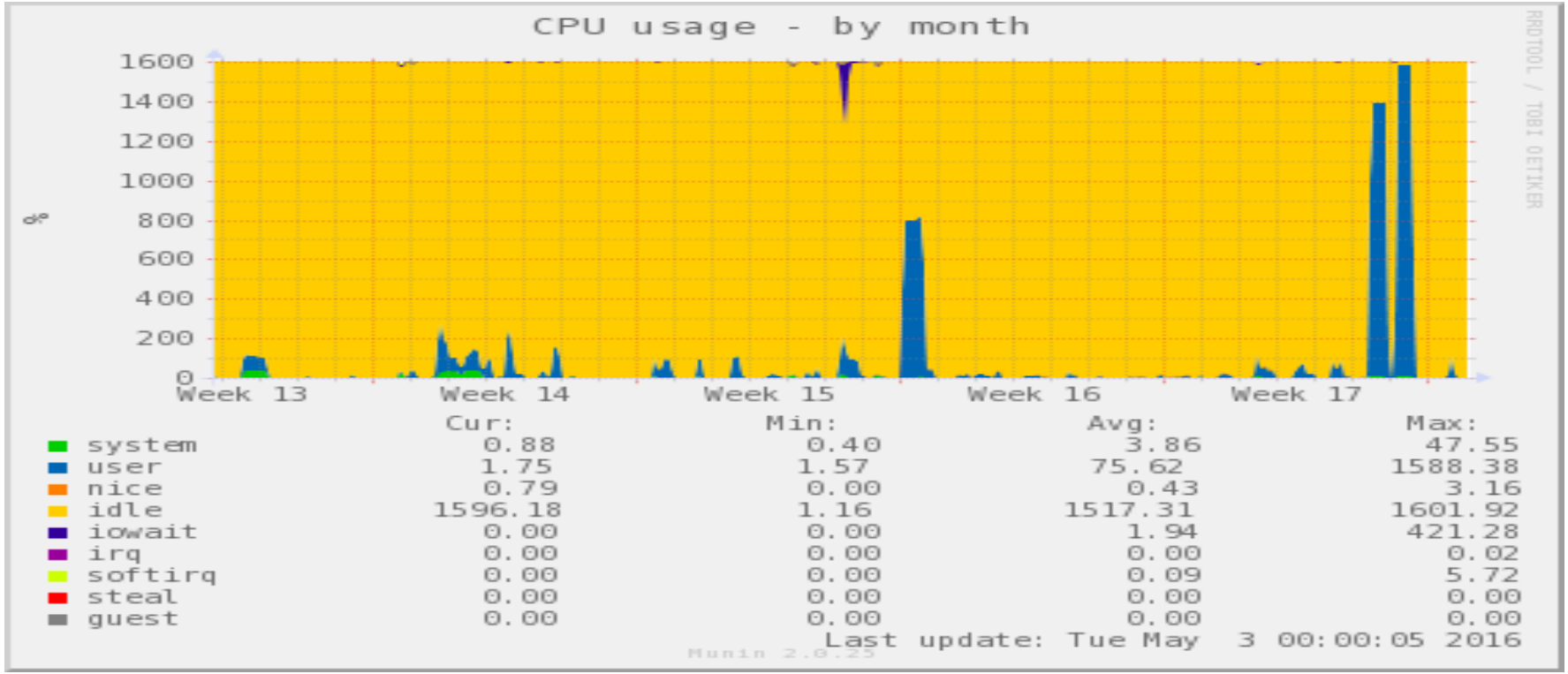
The basic logic





The basic logic

Esri union on postgres solved now ? No !!!!



NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Why not dive into Arcgis ?

NIBIO

NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Why was it important to solve in Postgis ?

NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

The code needs to handle :
- Big tables,
- Big polygon,
- Big servers
- Be generic

NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Unlimited sized table ?
Tested 72 million

If primary key INT and result > 2147483647 rows then fail

Divide and conquer pattern

both on big tables and single big polygons

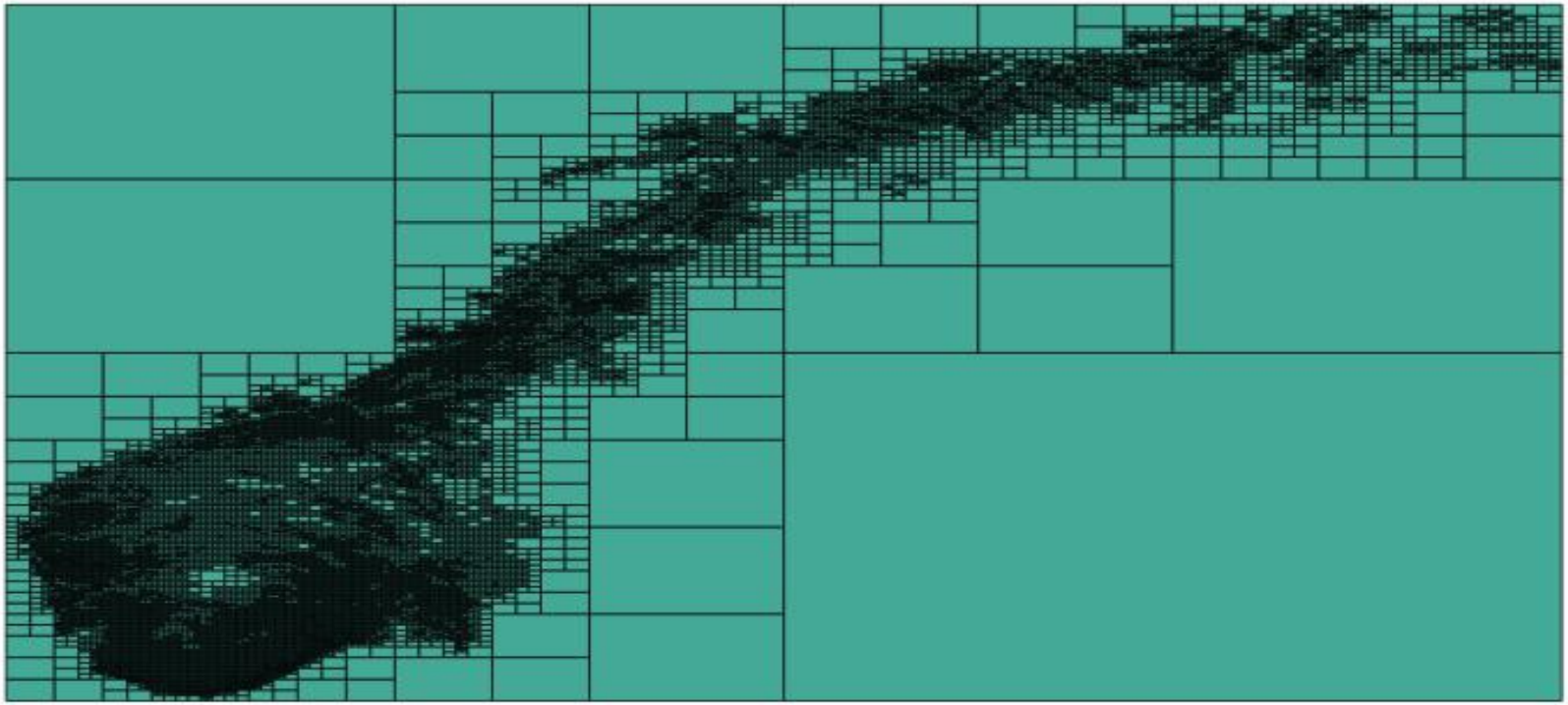
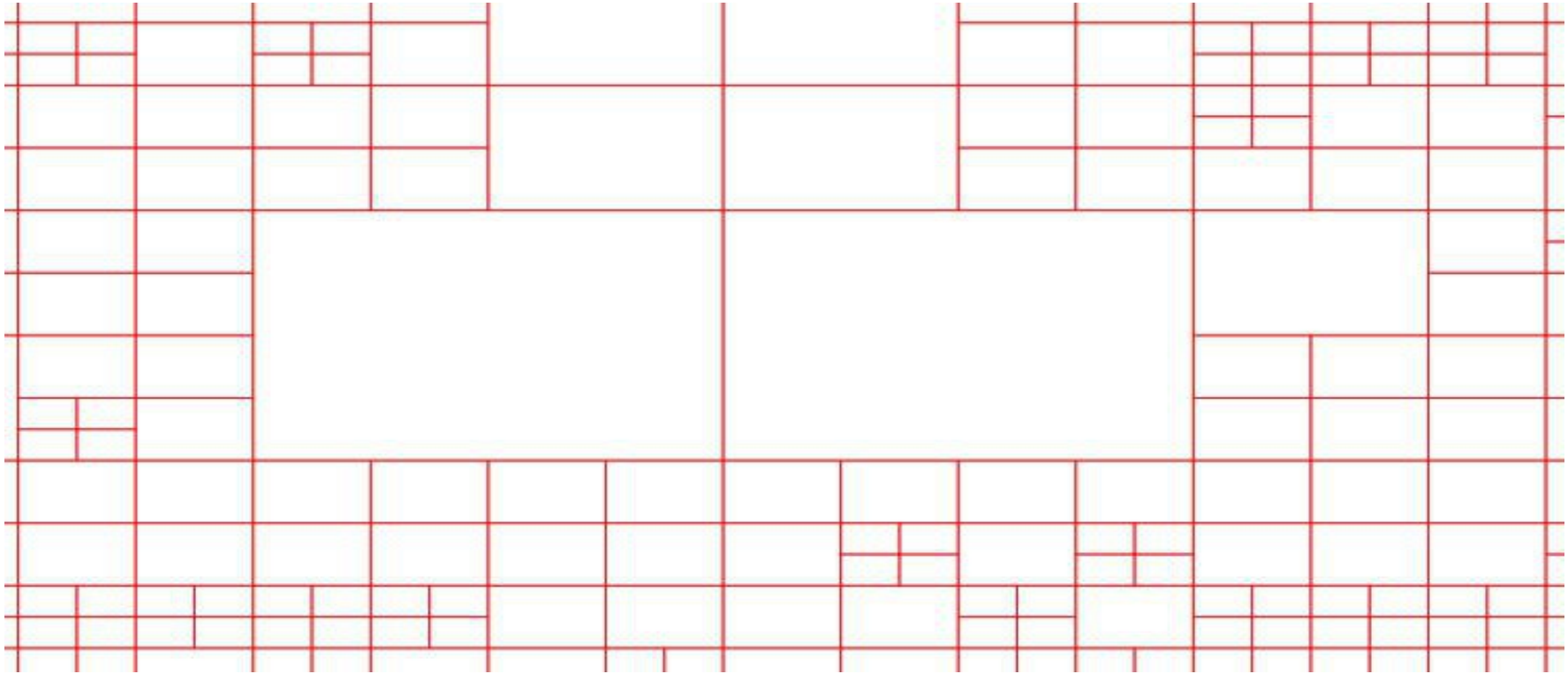
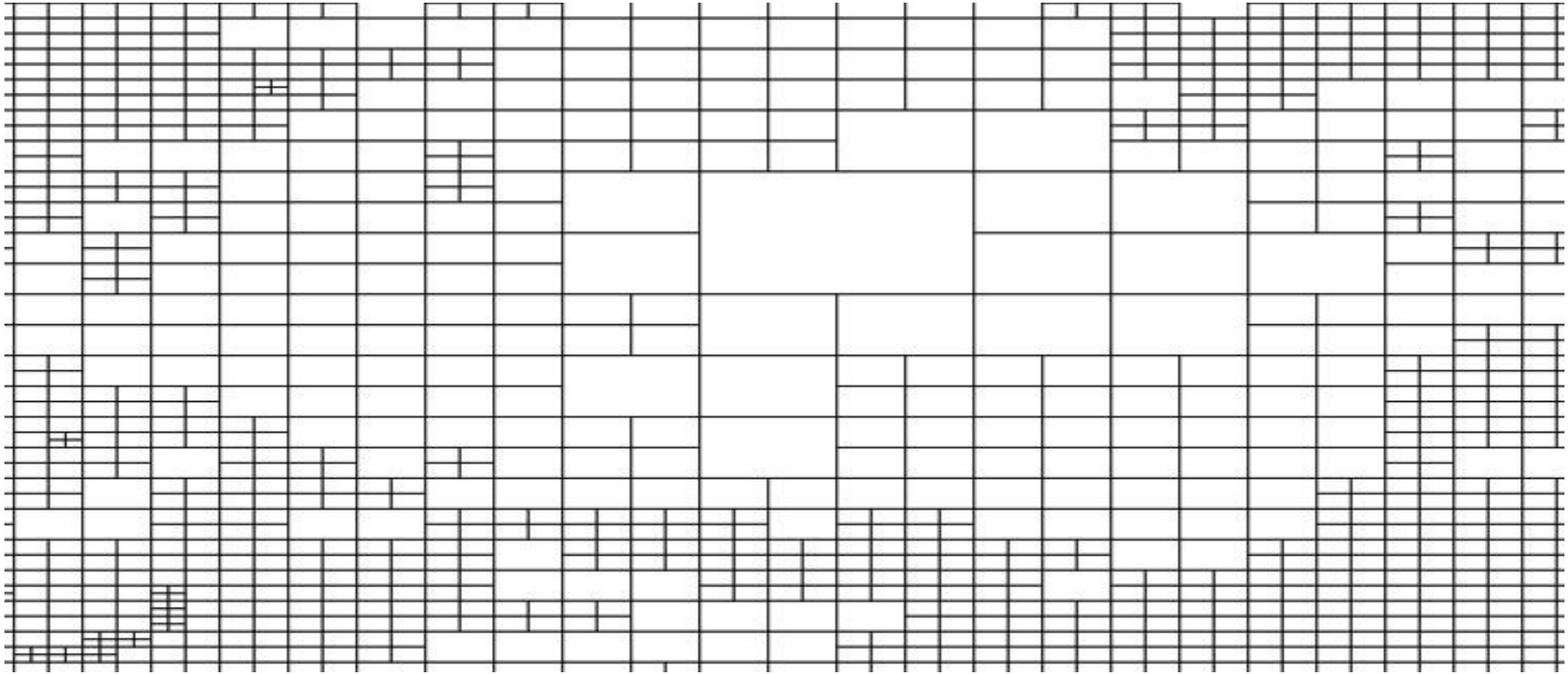


Table extent for both table

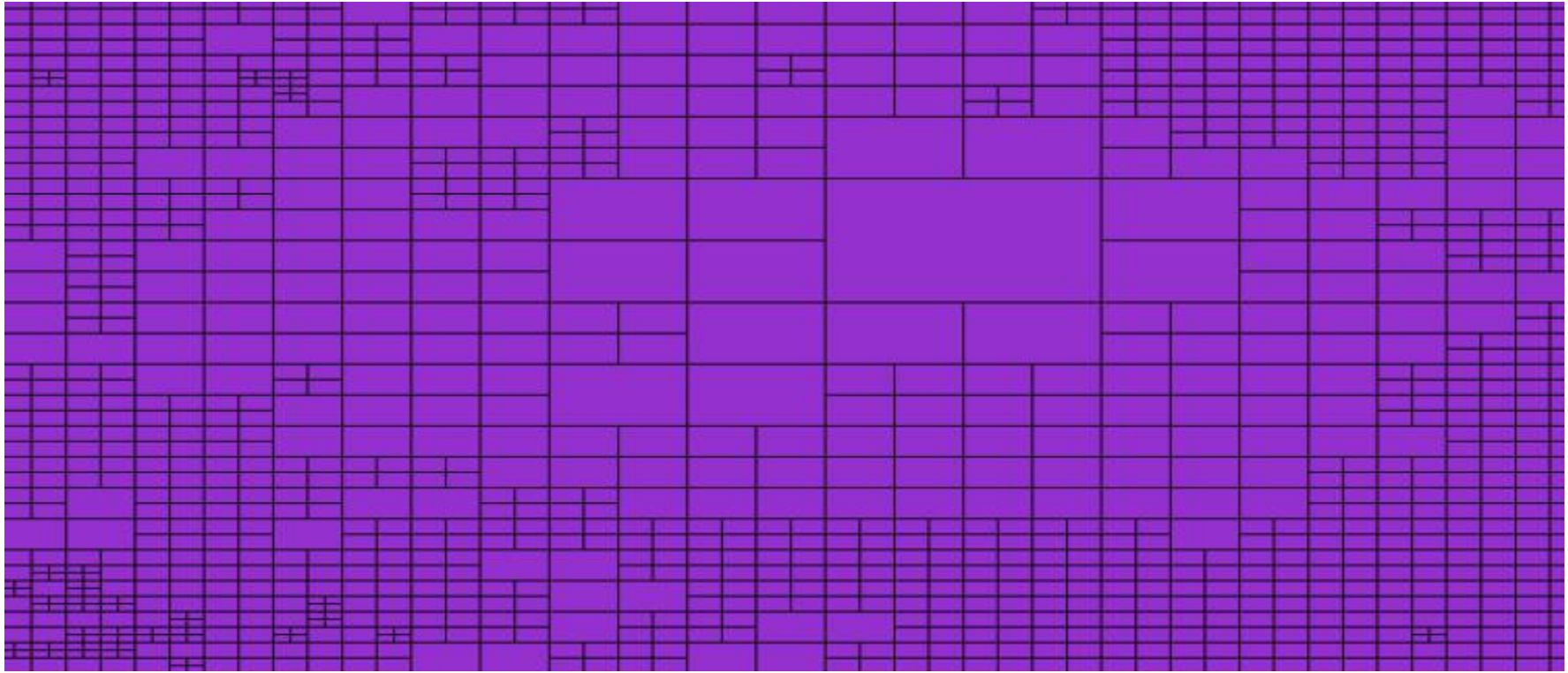
Grid for table one only



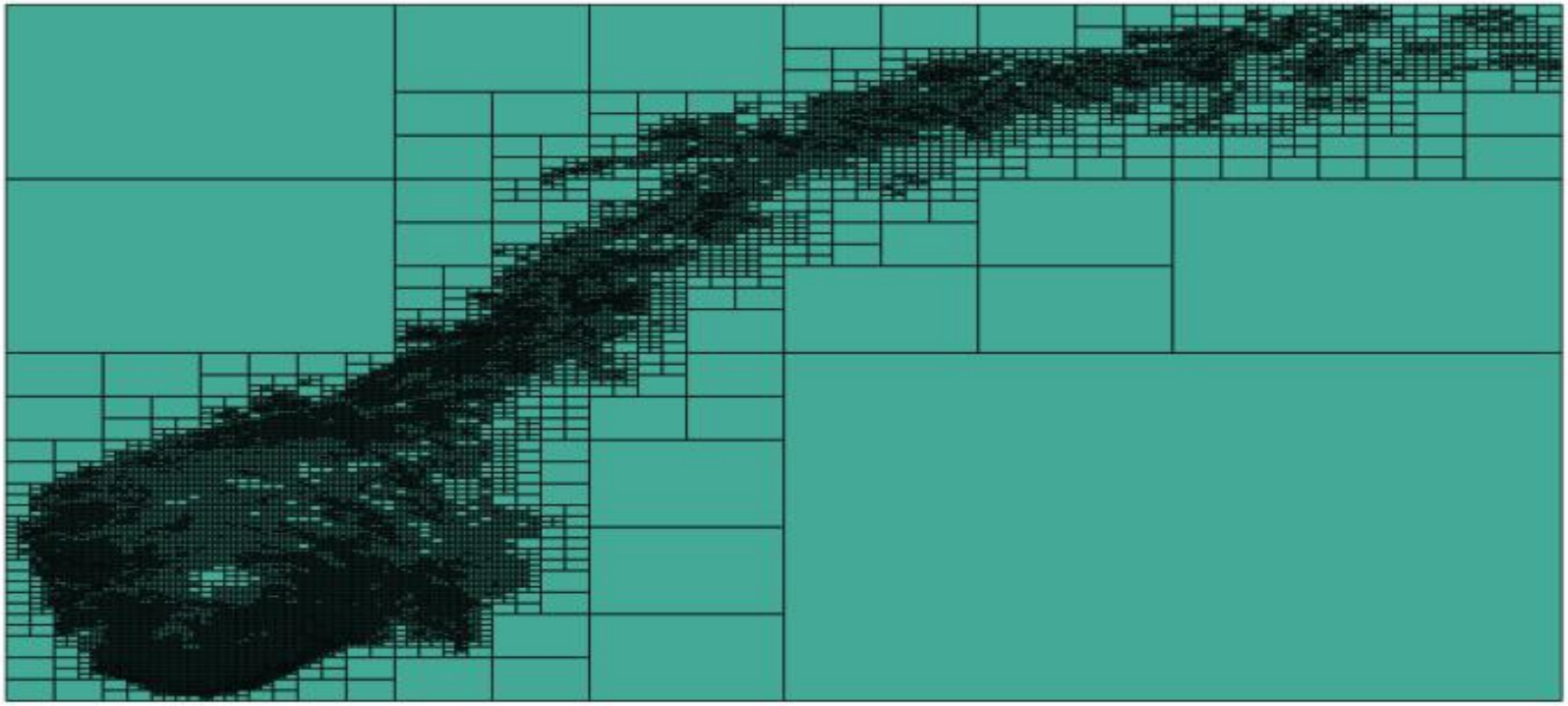
Grid for table two only



Grid cells depends on content from both tables



Grid cells depends on content from both tables



NIBIO

NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Compute Cell size fast or exact ?

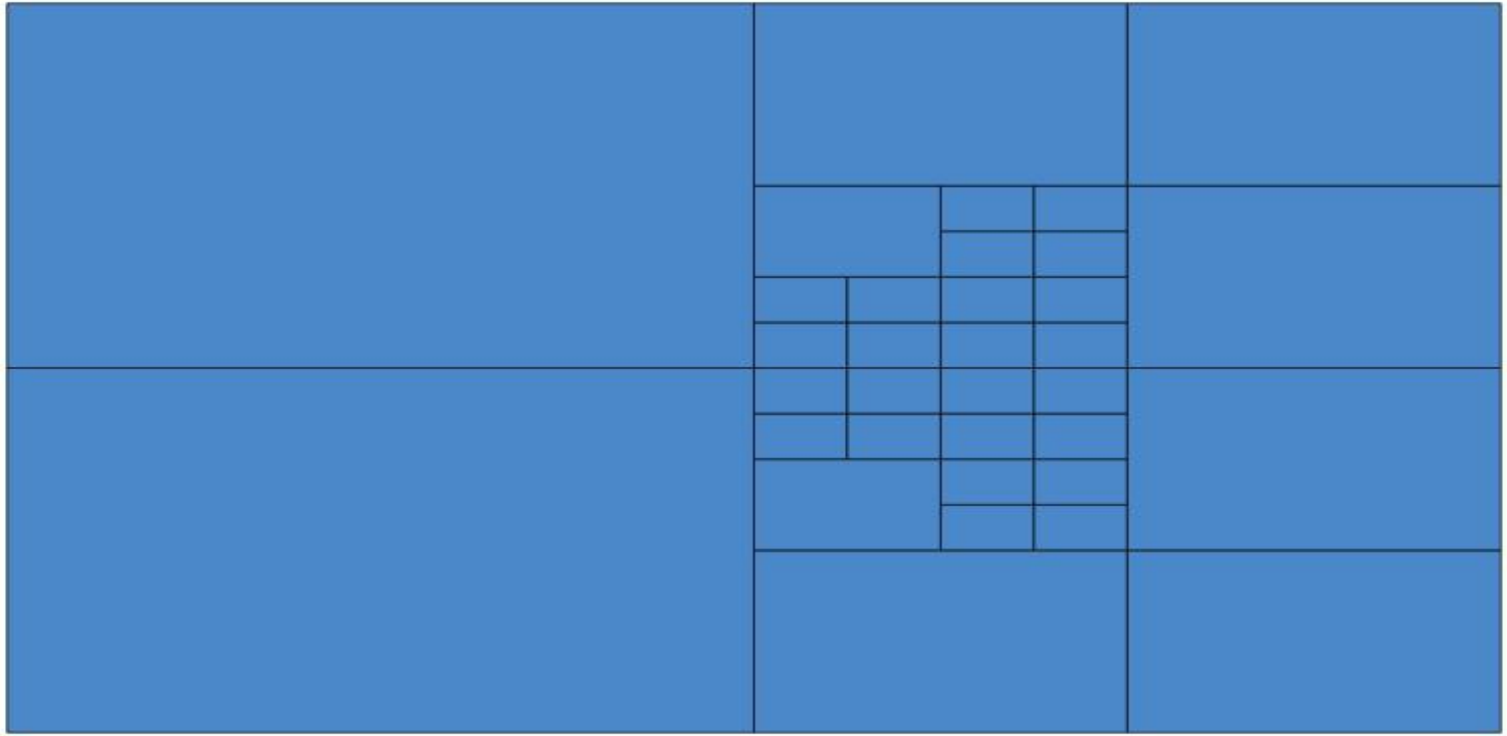
NIBIO

NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

&& operator

Content based balanced grid

https://github.com/larsop/content_balanced_grid

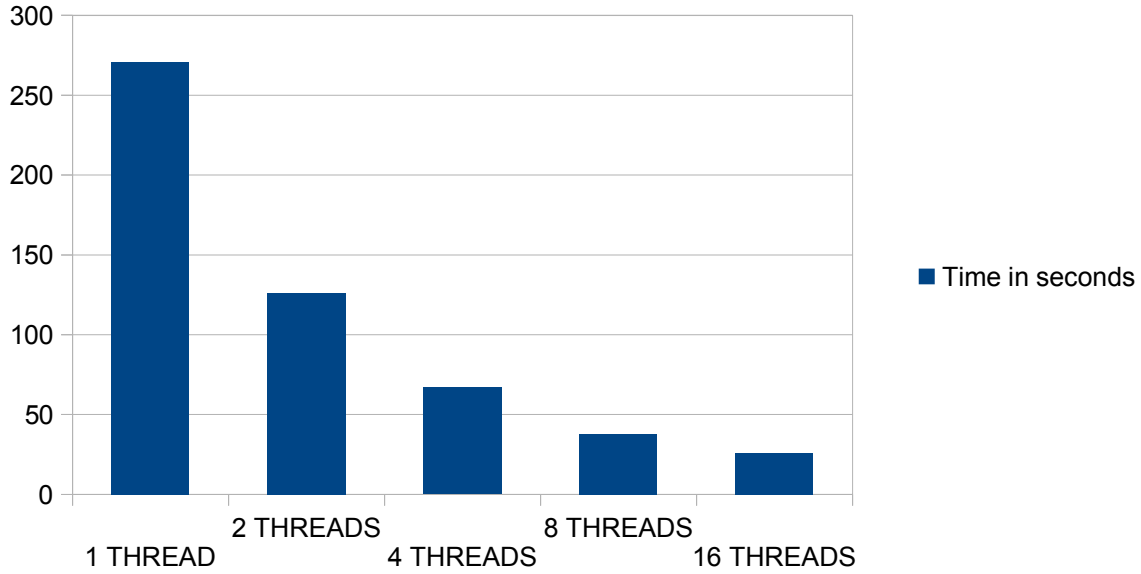


NIBIO

NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

GRIDS -> MORE CPU USAGE
GRIDS -> PARALELL PROCESSING

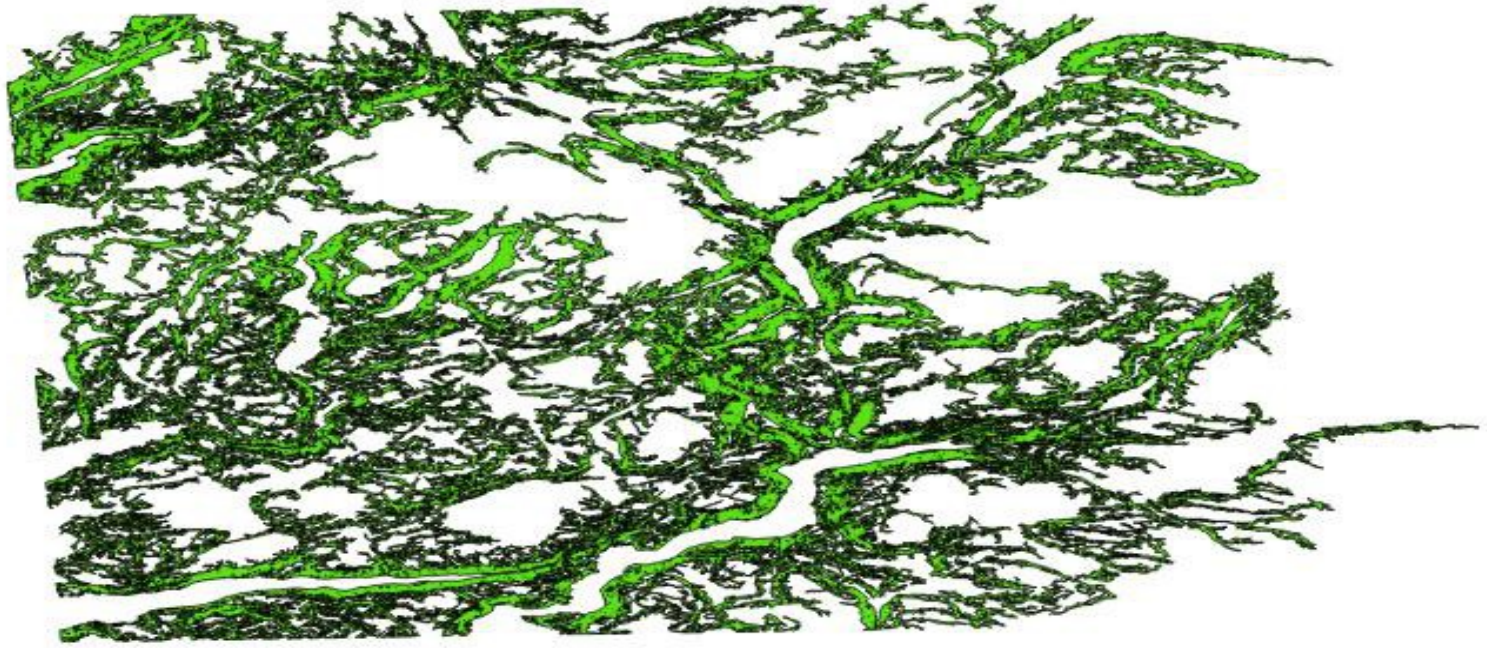
Parallel processing with 8 core dual CPU



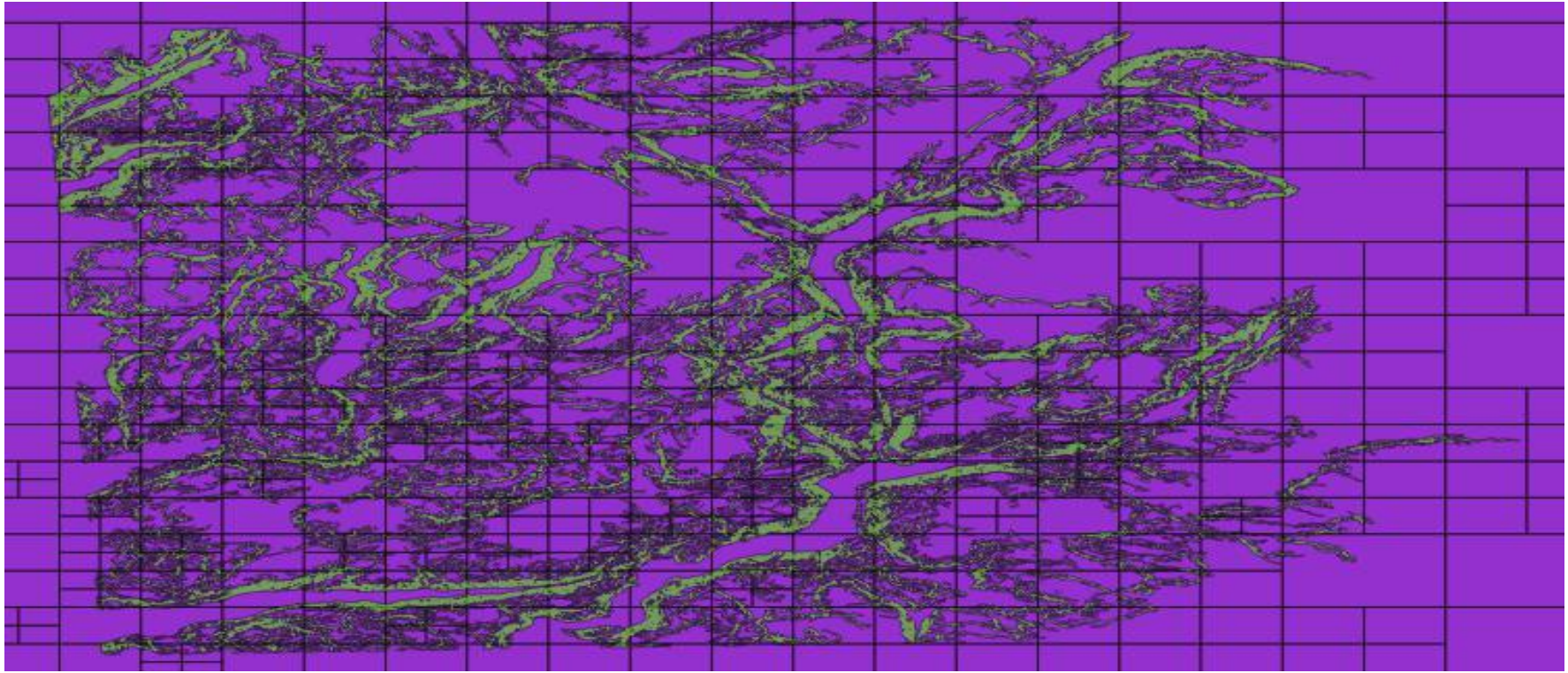
NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

USE THE GRID TO DIVIDE AND CONQUER

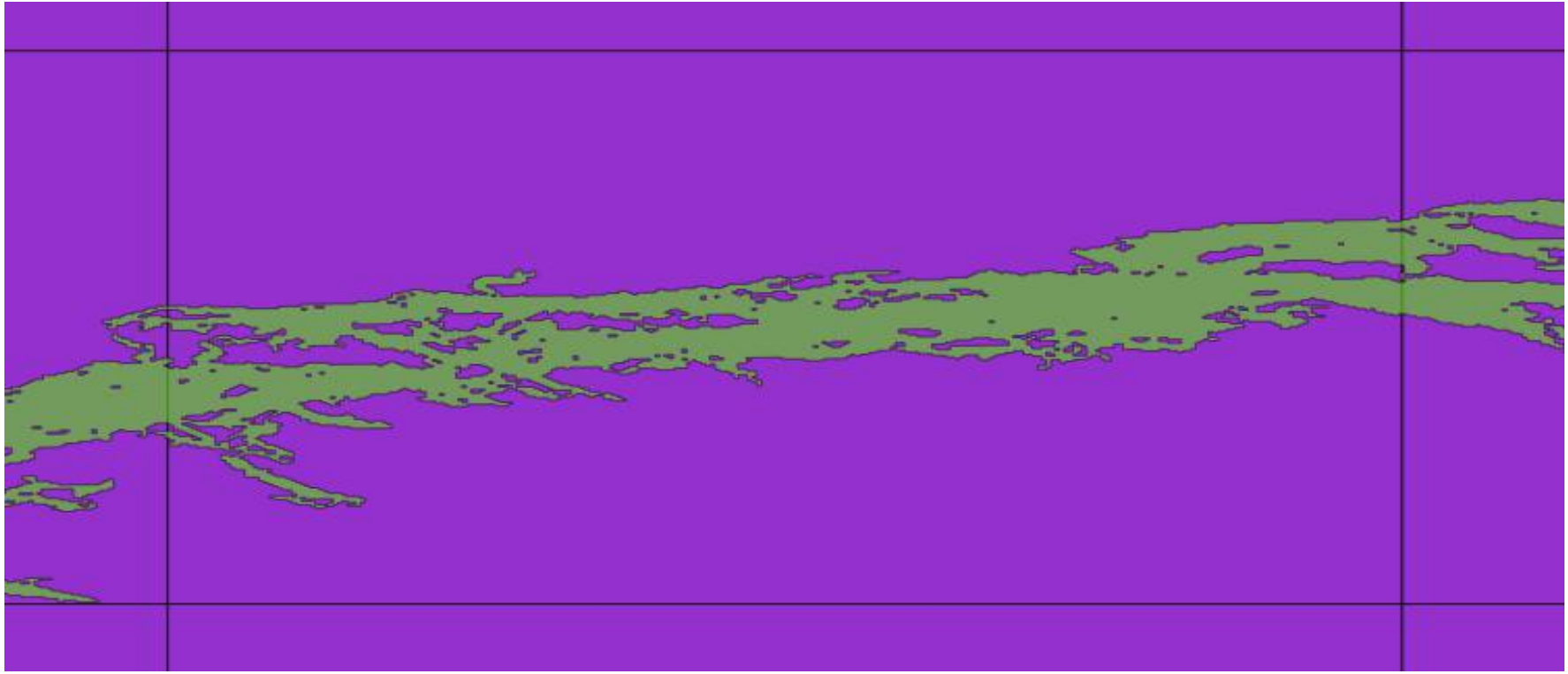
Handle Big Polygons (millions of points)



Big Polygons (millions of points)



Big Polygon and one single cell



Reduce the problem to single SQL

```
SELECT (ST_intersection(a.geo,b.geom)) as area
FROM
sl_lop.helling_data_d1 as a,
sl_lop.grid_ar5_helling b
WHERE b.id = 18 AND gid = 9419961;
```

Interior rings

```
SELECT SUM(num_points) FROM
(SELECT ST_NumPoints(ST_InteriorRingN(a.geo,
generate_series(1, 52079))) as num_points from
sl_lop.helling_data_d1 as a where gid = 9419961) as test;
Time : 474375.633 ms.

SELECT  sum(ST_Numpoints(ST_ExteriorRing(geom))) FROM
(SELECT (ST_DumpRings(a.geo)).geom from
sl_lop.helling_data_d1 as a where gid = 9419961) as test;
Time :  396.959
```

Build a new polygon with interior rings that intersects our current cell

```
SELECT ST_MakePolygon(exterior.exterior_ring,
interior.interior_ring) AS simple_polygon FROM
( SELECT ST_ExteriorRing(a.geo) as exterior_ring FROM
sl_lop.helling_data_d1 AS a WHERE a.gid = 9419961
) as exterior,
( SELECT (array_agg(ST_ExteriorRing(ring))) AS interior_ring
FROM
( select (rec).geom as ring, (rec).path[1] as arrayid from (
SELECT ST_DumpRings(a.geo) as rec
from sl_lop.helling_data_d1 as a
WHERE a.gid = 9419961) as a) as a,
sl_lop.grid_ar5_helling b
WHERE b.id = 18 AND b.geom && a.ring AND a.arrayid > 0
missing holes later
) as interior ) a
```

Lets wrap SQL into simple function

https://github.com/larsop/esri_union/src/main/sql/function_01_esri_union_intersection.sql

```
(SELECT (array_agg(ST_ExteriorRing(a.ring))) AS
interior_ring   FROM
( SELECT (rec).geom AS ring, (rec).path[1] AS arrayid
FROM (
SELECT ST_DumpRings(g1) AS rec
) AS aaa
) AS a
WHERE g2 && a.ring AND a.arrayid > 0
```

Remove grid lines, find rows

```
SELECT id_array,id_to_keep
 FROM
 (
 SELECT * FROM (
 SELECT count(r.*) as counts, array_agg(distinct(r.id)) id_array,
min(r.id) as id_to_keep FROM
 sl_lop.res_ar250_sk_grl r
 WHERE r.t1_sl_sdeid is NOT NULL AND r.t2_komid is NOT NULL
 GROUP BY r.t1_sl_sdeid, r.t2_komid
 ) AS t WHERE counts > 1
 -- UNION WHERE r.t1_sl_sdeid is NOT NULL AND r.t2_komid is NULL
 -- UNION WHERE r.t2_komid is NOT NULL AND r.t1_sl_sdeid is NULL
  AS to_update
```

Remove grid lines, ST_Union and Group by

https://github.com/larsop/esri_union/ in function_01_esri_union_remove_grid.sql

```
UPDATE sl_lop.res_ar250_sk_grl AS r
SET geom =  ST_Multi(u.geom)
FROM (SELECT St_Union(r.geom) AS geom FROM sl_lop.res_ar250_sk_grl
r WHERE r.id = ANY('{3050,3681}') ) AS u
WHERE r.id = '3050';
```

```
DELETE FROM sl_lop.res_ar250_sk_grl AS r
WHERE r.id = ANY('{3050,3681}') AND r.id > '3050';
```

NIBIO
NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Postgis Toplogy and grid lines

NIBIO

NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Generic code

json_each_text

GENERIC CODE, USING EXECUTE ON GENERATED STRING

```
SELECT
array_agg( ''t_''|| %s || ''.''  || quote_ident(update_column) || '' AS
'' ||  '' t'' || %s || ''_'' || quote_ident(update_column)) AS
new_column_as_tmp,
array_agg('' t'' || %s || ''_'' || quote_ident(update_column)) AS
new_column_name_tmp,
array_agg(quote_ident(update_column)) AS org_column_name_tmp
FROM (
SELECT distinct(key) AS update_column
FROM  (SELECT * FROM %s limit 1) AS t, json_each_text(to_json((t)))
WHERE key != %L
) AS keys',i,i,i,
schema_table_name,
geo_colums_array[i]);
```

NIBIO

NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

Processing cells i parallel

Parallel processing, generate code

```
psql -h host -U postgres -t -q -o /tmp/t1.sql sl
-c"select get_esri_union_muti_thread(
'org_ar5.ar5_flate sl_sdeid geo',
'org_helling.hellingklasser_dted10_3soner_flate gid
geo',
'sl_lop.ar5_helling',3000,
'sl_lop.ar5_helling_c1',
False,
False)"
```

Generated code has different parameters for each cell

```
SELECT get_esri_union_cell('(1,37666,"{""org_ar5.ar5_flate AS
t_1"",""org_helling.hellingklasser_dted10_3soner_flate AS
t_2""}","{sl_sdeid,gid}","{""sl_sdeid,kartid,noyaktighet,objtype,informasjon,registrerings
versjon,synbarhet,maalemetode,datafangstdato,artype,arkartstd,arskogbon,kjoringsident,opph
av,verifiseringsdato,argrunnf,areal,artreslag"",""gid,h_klasse,utmsone""}","{""
t1_sl_sdeid, t1_kartid, t1_noyaktighet, t1_objtype, t1_informasjon,
t1_registreringsversjon, t1_synbarhet, t1_maalemetode, t1_datafangstdato, t1_artype,
t1_arkartstd, t1_arskogbon, t1_kjoringsident, t1_opphav, t1_verifiseringsdato,
t1_argrunnf, t1_areal, t1_artreslag"","" t2_gid, t2_h_klasse,
t2_utmsone""}","{""t_1.sl_sdeid AS  t1_sl_sdeid,t_1.kartid AS  t1_kartid,t_1.noyaktighet
AS  t1_noyaktighet,t_1.objtype AS  t1_objtype,t_1.informasjon AS
t1_informasjon,t_1.registreringsversjon AS  t1_registreringsversjon,t_1.synbarhet AS
t1_synbarhet,t_1.maalemetode AS  t1_maalemetode,t_1.datafangstdato AS
t1_datafangstdato,t_1.artype AS  t1_artype,t_1.arkartstd AS  t1_arkartstd,t_1.arskogbon AS
 t1_arskogbon,t_1.kjoringsident AS  t1_kjoringsident,t_1.opphav AS
t1_opphav,t_1.verifiseringsdato AS  t1_verifiseringsdato,t_1.argrunnf AS
t1_argrunnf,t_1.areal AS  t1_areal,t_1.artreslag AS  t1_artreslag"",""t_2.gid AS
t2_gid,t_2.h_klasse AS  t2_h_klasse,t_2.utmsone AS
t2_utmsone""}","{geo,geo}","{t_1.geo,t_2.geo}",sl_lop.ar5_helling,sl_lop.ar5_helling_c1,tm
p_data_esri_intersects_t1_7ce543ab6a2b5dc0c43af863b659d81a,tmp_data_esri_intersects_t2_7ce
543ab6a2b5dc0c43af863b659d81a,tmp_data_esri_intersects_7ce543ab6a2b5dc0c43af863b659d81a)')
```

Parallel processing of each cell

```
parallel -verbose -j 16
psql -h host -U user sl -c :::: /tmp/t1.sql
```

NIBIO

NORWEGIAN INSTITUTE OF
BIOECONOMY RESEARCH

This code,tests and wiki is found at https://github.com/larsop/esri_union

Thanks