

Gathering Lines of Code in an instance

If you're wondering the amount of Lines of Code tracked by your SonarQube installation, then the below pointers should help you get some good insight.

SonarQube LTS v6.7 and newer

The global amount of Lines of Code is directly reported under the [System Info](#) admin page, just expand the *System* section.

Note : if you're analyzing branches with the [Branch Plugin](#), note that only the lines from the largest branch will be counted. The LOC in the rest of the branches are ignored.

Versions older than 6.7

The best approach there is to query the database. The actual query varies based on the version of SonarQube and the database engine. Two queries are provided:

- one query that counts LOCs across **all** projects
- one query that filters out project *branches* (i.e. projects analysed with *sonar.branch* parameter). **Note**: this query is accurate only if projects with branches are also analysed once without *sonar.branch*.

SonarQube LTS v5.6.x Global LOCs

```
select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc';
```

LOCs without sonar.branch ^[1]

```
select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc'
and (
INSTR(p.kee, ':') = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTR(p.kee, 1, LENGTH(p.kee) - INSTR(REVERSE(p.kee), ':'))
)
);
```

Global LOCs

```
select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc';
```

LOCs without sonar.branch

Not feasible on this specific database

Global LOCs

```
select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc';
```

LOCs without sonar.branch [1]

```
select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc'
and (
POSITION(':', IN p.kee) = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTRING(p.kee, 0, LENGTH(p.kee) - POSITION(':', in REVERSE(p.kee)) + 1)
)
);
```

Global LOCs

```
select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc';
```

LOCs without sonar.branch [1]

```
select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc'
and (
INSTR(p.kee, ':') = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTR(p.kee, 0, INSTR(p.kee, ':', -1) - 1)
)
);
```

Global LOCs

```

select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc';

```

LOCs without sonar.branch [1]

```

select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.project_id = p.id
inner join project_measures pm on pm.snapshot_id = s.id
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_resource_id is null
and pm.person_id is null
and m.name='ncloc'
and (
CHARINDEX(':', p.kee) = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTRING(p.kee, 0, LEN(p.kee) - CHARINDEX(':', REVERSE(p.kee)) + 1)
)
);

```

SonarQube 6.x (<6.7)

Global LOCs

```

select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.component_uid = p.uid
inner join project_measures pm on pm.analysis_uid = s.uid
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uid is null
and pm.component_uid = p.uid
and pm.person_id is null
and m.name='ncloc';

```

LOCs without sonar.branch [1]

```

select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.component_uid = p.uid
inner join project_measures pm on pm.analysis_uid = s.uid
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uid is null
and pm.component_uid = p.uid
and pm.person_id is null
and m.name='ncloc'
and (
INSTR(p.kee, ':') = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTR(p.kee, 1, LENGTH(p.kee) - INSTR(REVERSE(p.kee), ':'))
)
);

```

Global LOCs

```

select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.component_uuid = p.uuid
inner join project_measures pm on pm.analysis_uuid = s.uuid
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uuid is null
and pm.component_uuid = p.uuid
and pm.person_id is null
and m.name='ncloc';

```

LOCs without sonar.branch

Not feasible on this specific database

Global LOCs

```

select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.component_uuid = p.uuid
inner join project_measures pm on pm.analysis_uuid = s.uuid
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uuid is null
and pm.component_uuid = p.uuid
and pm.person_id is null
and m.name='ncloc';

```

LOCs without sonar.branch [1]

```

select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.component_uuid = p.uuid
inner join project_measures pm on pm.analysis_uuid = s.uuid
inner join metrics m on m.id=pm.metric_id
where s.islast = true
and p.enabled = true
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uuid is null
and pm.component_uuid = p.uuid
and pm.person_id is null
and m.name='ncloc' and (
POSITION(':', IN p.kee) = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTRING(p.kee, 0, LENGTH(p.kee) - POSITION(':', in REVERSE(p.kee)) + 1)
)
);

```

Global LOCs

```

select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.component_uuid = p.uuid
inner join project_measures pm on pm.analysis_uuid = s.uuid
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uuid is null
and pm.component_uuid = p.uuid
and pm.person_id is null
and m.name='ncloc';

```

LOCs without sonar.branch [1]

```

select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.component_uuid = p.uuid
inner join project_measures pm on pm.analysis_uuid = s.uuid
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uuid is null
and pm.component_uuid = p.uuid
and pm.person_id is null
and m.name='ncloc'
and (
INSTR(p.kee, ':') = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTR(p.kee, 0, INSTR(p.kee, ':', -1) - 1)
)
);

```

Global LOCs

```

select sum(pm.value) as global_loc from projects p
inner join snapshots s on s.component_uuid = p.uuid
inner join project_measures pm on pm.analysis_uuid = s.uuid
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uuid is null
and pm.component_uuid = p.uuid
and pm.person_id is null
and m.name='ncloc';

```

LOCs without sonar.branch [1]

```

select sum(pm.value) as loc_without_branch from projects p
inner join snapshots s on s.component_uuid = p.uuid
inner join project_measures pm on pm.analysis_uuid = s.uuid
inner join metrics m on m.id=pm.metric_id
where s.islast = 1
and p.enabled = 1
and p.qualifier = 'TRK'
and p.scope = 'PRJ'
and p.copy_component_uuid is null
and pm.component_uuid = p.uuid
and pm.person_id is null
and m.name='ncloc'
and (
CHARINDEX(':', p.kee) = 0 or not exists(
select * from projects p_root where p_root.kee = SUBSTRING(p.kee, 0, LEN(p.kee) - CHARINDEX(':', REVERSE(p.kee)) + 1)
)
);

```

[1] : the *LOCs without sonar.branch* query is accurate only if projects with branches are also analysed once without sonar.branch