Item 67
git_comments:

git_commits:

1. **summary:** LUCENE-8811: Revert on 8.x.
   message: LUCENE-8811: Revert on 8.x.

github_issues:

github_issues_comments:

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

1. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
2. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
3. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
4. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
   label: code-design
5. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
6. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have

themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.

7. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
   label: code-design
8. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
9. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
   description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
   label: code-design
10. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
11. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
12. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
13. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
14. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from

BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.

15. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
16. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
    label: documentation
17. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
18. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
19. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
20. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
    label: code-design
21. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
22. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for

instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.

23. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
24. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
25. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
26. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
27. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
28. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
29. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
    label: code-design
30. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.

label: code-design

31. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
32. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
33. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
34. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
35. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
36. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
37. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
38. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
39. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery

description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.

40. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
41. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
    label: code-design
42. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
43. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
    label: code-design
44. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
45. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
    label: code-design
46. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
    description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
47. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery

description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.
label: code-design

48. **summary:** Add maximum clause count check to IndexSearcher rather than BooleanQuery
description: Currently we only check whether boolean queries have too many clauses. However there are other ways that queries may have too many clauses, for instance if you have boolean queries that have themselves inner boolean queries. Could we use the new Query visitor API to move this check from BooleanQuery to IndexSearcher in order to make this check more consistent across queries? See for instance LUCENE-8810 where a rewrite rule caused the maximum clause count to be hit even though the total number of leaf queries remained the same.

jira_issues_comments:

1. [~jpountz] Thanks for opening this. I posted a patch in LUCENE-8810 which handles the boolean query case. I think this problem would be most prevalent for boolean queries, but if you believe a more general check is valuable, I am happy to put up a patch for checking through IndexSearcher.
2. [^LUCENE-8811.patch] [~jpountz] Attached is a patch implementing the same. Please let me know your thoughts.
3. I'm wondering whether IndexSearcher could return "this" in getSubVisitor (including for MUST_NOT) and then we would count the number of calls to visitLeaf? This way it would work for DisjunctionMaxQuery and custom queries as well. I like that it removes ways to by-pass the maximum clause count by splitting boolean queries into multiple sub boolean queries. Wondering what others think about it [~thetaphi] [~rcmuir] [~mikemccand]?
4. **body:** You'd need to count calls to visitLeaf() and collectTerms(), and it wouldn't be exact because of things like MultiPhraseQuery which treats itself as a combination of phrases and disjunctions, but I like the idea. Setting the maximum permissible query size to be per-searcher rather than global on BQ would also be very nice, as you could do things like restrict the complexity of queries over particular indexes or for particular users.
label: code-design
5. [~jpountz] [~romseygeek] Attached patch implements a QueryVisitor which counts leaf visits and terms which are consumed and adds per IndexSearcher maximum clause count limits. Please let me know your thoughts and comments. [^LUCENE-8811.patch]
6. The last patch had an unintentional type: attached is a fixed patch: [^LUCENE-8811.patch]
7. **body:** Thanks [~atris], this looks very close! I know I said that having the query-size be per-searcher rather than static would be a good thing, but I think it will need to be done in a follow-up as it's confusing things a bit here due to various queries and query builders still referring to the static value. Would you mind backing that part out, and leaving it as static-only? Also, can you use expectThrows() in tests, rather than try/fail/catch? It makes things easier to read.
label: code-design
8. [~romseygeek] Attached is a patch which does just that and depends on BooleanQuery's max clauses as the fence for all queries. I have opened LUCENE-8839 - Add IndexSearcher Level Query Max Clause Count Limits and will post a follow-up patch there. Let me know if this is fine. [^LUCENE-8811.patch]
9. **body:** Sorry, I wasn't clear - can you move the BooleanQuery max clause count to IndexSearcher (as it makes more sense to be there now that it's a general limit on queries, not just applied to booleans), but keep it static. I also think we should run the query count after rewriting the query, rather than before - some multi-term queries can end up expanding into very large boolean combinations, and we should try and catch that.
label: code-design
10. [~romseygeek] Ok, I misinterpreted your previous comment to do the actual refactor of max clauses count from BooleanQuery to IndexSearcher in a follow up patch due to the different constructs it touches, apologies. I wanted to confirm our line of thought here: max clauses count would be a part of IndexSearcher now (as would be TooManyClauses exception, as done in previous iteration). However, IndexSearcher will not have the ability to have per searcher custom values for the maxclauses limit, and

the new QueryVisitor will still use the static value. The custom per searcher value will be done in a separate patch. Would that be correct? I wanted to double check before posting another iteration, thanks.
11. That's correct, yes.
12. [~romseygeek] Thanks, attached is an updated patch. [^LUCENE-8811.patch]
13. I am not sure if it's correct to count in both visitTerms and visitLeafs. Would this not count 2 times? The main query is counted anyways because visit() calls the visitor's visitLeaf, so the clause is counted.
14. Otherwise it's a great idea to use visitors.
15. [~thetaphi] Arent consumeTerms and visitLeaf mutually exclusive, since consumeTerms is called by queries matching on specific terms, and visitLeaf by queries which do not do that disambiguation?
16. **body:** This is why I am asking. I had a project and I implemented several visitors. And neither the documentation nor the behavior looks consistent. From my understanding, visitLeaf is called for every query you see. And visitTerms additionally if there are terms. Check the code, Query.visit by default always calls with itself and then delegates to a protected method.
    label: documentation
17. > Check the code, Query.visit by default always calls with itself and then delegates to a protected method. Atri is correct here, there is no protected method.
18. [~romseygeek] Please let me know if the current iteration needs changes – happy to follow-up
19. [~romseygeek] Please let me know if this looks fine. Thanks
20. **body:** In general this looks good to me, I'd like to give [~romseygeek] a chance to review the query visiting logic which he knows better than I do. Some minor comments: - can you add a note to the MIGRATE.txt? - Javadocs of setMaxClauseCount say it is the maximum number of clauses per BooleanQuery but it's now actually the maximum number of clauses across the entire query - can you use expectThrows instead of try/catch blocks in tests whenever applicable?
    label: code-design
21. Thanks [~jpountz], attached is an updated patch [^LUCENE-8811.patch]
22. Please let me know if the patch looks fine. Happy to iterate
23. +1 from me, I'd like [~romseygeek] to have a look at the visitor logic before pushing. (No hurry Alan, make sure to enjoy Berlin Buzzwords. :))
24. Any chance we could push this one? Happy to make any changes
25. Commit fa3bf88783aca02b16b1abd9103140b5f45959b7 in lucene-solr's branch refs/heads/branch_8x from Alan Woodward [https://gitbox.apache.org/repos/asf?p=lucene-solr.git;h=fa3bf88] LUCENE-8811: Move max clause checks to IndexSearcher
26. Commit 53f56fb7ad44afab9c78fab2e89737438093247f in lucene-solr's branch refs/heads/master from Alan Woodward [https://gitbox.apache.org/repos/asf?p=lucene-solr.git;h=53f56fb] LUCENE-8811: Move max clause checks to IndexSearcher
27. Commit 6751c072ab5a7b28a13d0e639a2fbe08e13f02a5 in lucene-solr's branch refs/heads/master from Alan Woodward [https://gitbox.apache.org/repos/asf?p=lucene-solr.git;h=6751c07] LUCENE-8811: Remove deprecated BooleanQuery maxCount methods
28. Thanks for the iterations [~atris]
29. **body:** Thanks [~romseygeek] for pushing! A small nit: I think git somehow botched up the patch during commit ? (I see your name as both author and committer).
    label: code-design
30. **body:** I was reviewing the changelog for 8.2, this change looks a bit too breaking for a minor and should probably wait for 9.0? We can separately address LUCENE-8810 by disabling the flattening of disjunctions if the new BooleanQuery would have more than 1024 clauses?
    label: code-design
31. [~jpountz] I had originally raised a patch which implemented your suggested approach, should we commit that for 8.2, and let all other branches have the actual change introduced by this JIRA?
32. [~atris] If the patch you are thinking of is the one on LUCENE-8810, I was thinking of something even simpler that would catch the TooManyClauses exception when trying to flatten the query.
33. [~jpountz] Yeah, that is what I was thinking of, but I see your view point. I will raise a PR shortly
34. Thanks! I'll revert this change from 8.x and 8.2 in the meantime.
35. Commit 2885a115782b5844972bdbb9e281f720ee542c1a in lucene-solr's branch refs/heads/branch_8x from Adrien Grand [https://gitbox.apache.org/repos/asf?p=lucene-solr.git;h=2885a11] LUCENE-8811: Revert on 8.x.
36. Commit 607c46c9979899f9ef701ae501fd273ce4f0f555 in lucene-solr's branch refs/heads/master from Adrien Grand [https://gitbox.apache.org/repos/asf?p=lucene-solr.git;h=607c46c] LUCENE-8811: Undo removal of deprecations.

37. Commit 193c77fe1a0a37dc16ad220f1be644c1c971769c in lucene-solr's branch refs/heads/branch_8_2 from Adrien Grand [https://gitbox.apache.org/repos/asf?p=lucene-solr.git;h=193c77f] LUCENE-8811: Revert on 8.x.
38. Closing after the 8.2.0 release
39. Given that the change was reverted on 8.x shouldn't this issue be reopened? I guess the fixVersion should also be modified.
40. I've changed the fixVersion to be 9.0. The change is still in master, so no need to reopen.
41. **body:** Naively, I thought that after this change the number of clauses in a {{TermInSetQuery}} will be the number of terms that it contains. Looking better into the implementation of {{getNumClausesCheckVisitor}} (thanks [~rubenql] for noticing) it seems that the whole {{TermInSetQuery}} will count as only one clause. This behavior seems a bit inconsistent especially since a {{TermInSetQuery}} is somehow equivalent to a {{BooleanQuery}}. Why there should be limits on one and not for the other?
    label: code-design
42. Friendly reminder to see if someone can answer [~zabetak]'s question in the previous comment.
43. **body:** TermInSetQuery is designed to be a more efficient replacement for a boolean disjunction of terms, so having it trip the max clauses check would defeat the point of having it in the first place.
    label: code-design
44. So if I interpret well your response [~romseygeek], you are saying that {{TermInSetQuery}} should accept an unlimited number of terms. Is that correct? Looking again into the summary and discussion of this issue, I see that the goal was to "make this check more consistent across queries". I don't clearly see why {{TermInSetQuery}} should remain unbounded. On the other hand, if we wanted to enforce the check only for a {{BooleanQuery}} then why using the {{getNumClausesCheckVisitor}} visitor on every query. I see that {{TermInSetQuery#visit}} for example already iterates through all terms so if we don't need then we are just wasting CPU cycles without a very good reason. Sorry to insist on this but it is a change that will likely break our current implementation in the downstream project and I guess it will also affect quite a few others.
45. **body:** Hi [~zabetak], the idea is to try and prevent runaway query execution; TermInSetQuery is implemented so that it can handle a very large number of terms efficiently, while the equivalent boolean disjunction would be very slow and use lots of memory. So we primarily want to run this check on BooleanQuery, but we also want to handle nested booleans, booleans wrapped in other queries, etc, in which case a QueryVisitor makes most sense. > I see that TermInSetQuery#visit for example already iterates through all terms so if we don't need then we are just wasting CPU cycles without a very good reason. This should definitely be fixed! I think we probably want to change TermInSetQuery#visit to call consumeMatchingTerms() with a lazy runAutomaton implementation; the num clauses check visitor also needs to count consumeMatchingTerms calls.
    label: code-design
46. I opened LUCENE-9349 to address TermInSetQuery's visit() implementation.