jdk14

13 May 2020

Roman Ivanov created group «jdk14» with members Roman Ivanov, Pavel Bludov, Richard Veach and Erik Silkensen

Roman Ivanov   07:46

RI

Hi all

Richard Veach   07:46

RV

Hello

Roman Ivanov   08:52

RI

@nmancuso , hi

I do not see a way to add Nick to this group   08:54

Richard Veach   08:56

RV

Is this a private channel?

Roman Ivanov invited Nick Mancuso

Roman Ivanov   08:57

RI

Hi Nick

Nick Mancuso   08:57

NM

Hello!

Pavel Bludov   10:15

PB

Hello

Nick Mancuso   10:18

NM

Hi Pavel!

Roman Ivanov   20:23

RI

We need to think on plan and sequence of updates to grammar
https://docs.google.com/document/d/1VtM3CWSJSYh_ZSPIgJSYbCTWKebR p_D9aP0bY7KogeU/edit?usp=sharing , if we some concerns on breaking compatibility it is better to raise in advance

Nick Mancuso   20:29

NM

Pavel and I have had some discussion already regarding enhanced instanceOf and pattern matching here:
https://github.com/checkstyle/checkstyle/issues/7290#issuecomment- 609025203

plan to fix

14 May 2020

Pavel Bludov changed group photo

J

Pavel Bludov changed group photo

J

Pavel Bludov changed group photo

J

15 May 2020

Roman Ivanov pinned this message

RI **Roman Ivanov** 05:43

Hi All, we need to make some planning activity, please define exact list of jdk14 syntax features to need, exact sequence of implementation of them in Gdoc. Please put in Github project related issues.

NM **Nick Mancuso** 05:47

Hi Roman, I tried to add the existing issues from checkstyle to the project the other day when you posted, but it appears that I am unable to. Is adding issues to the project only for mentors?

RI **Roman Ivanov** 06:18

Yes, please share here links to related issues, mentors will add them to project

NM **Nick Mancuso** 06:19

Ok, I'll get on this now.

https://github.com/checkstyle/checkstyle/issues/7290 06:20

https://github.com/checkstyle/checkstyle/issues/7103 06:21

https://github.com/checkstyle/checkstyle/issues/6615 06:22

Roman Ivanov

jdk14

Pavel Bludov   10:09

PB

Done. We have an issue for module-info.java
https://github.com/checkstyle/checkstyle/issues/3059
@romanivanovjr The issue is closed, but not solved. The parser should be
able to parse Java modules.
Should we reopen this issue or create a new one?

Roman Ivanov   12:23

RI

A long time ago we decided to not support it, as it is not a Java grammar, it is
something else(config of some type), we do not have that much to validate in
it. We try to skip it, as update to grammar was a bit scary. So we can revisit
this, but let's do this as low priority.

                                16 May 2020

Pavel Bludov   00:22

PB

Yes, module-info is not a regular Java file. Most Checkstyle checks have
nothing to do with this. But at least there are may be comments and Javadoc.
We should validate them. Also, there may be annotations. We have checks for
annotations. And maybe we will have checks specifically for modules
sometime. I'll start an issue to discuss.

https://github.com/checkstyle/checkstyle/issues/8240   00:32
And yes, this should be optional low priority issue not related to this project. I
agree to postpone this until we start moving our codebase to java11.

Richard Veach   08:57

RV

What is the scope of this project? We are adding new syntax. Are we going to
add their support to existing modules? Are we going to do any analysis on how
new syntax might damage other checks? Have we identified any projects that
use this new syntax we can do regression on?

Nick Mancuso   09:01

NM

Richard I will start identifying projects this week. Off the top of your head, are
there any we currently test where warnings for the newer features are
suppressed?

Roman Ivanov   13:09

RI

Please create exact list and sequence of fixes it will help a lot.

Richard Veach   13:10

RV

If you don't have access in github, then add it to the google doc and then
mentor can do the updates

Nick Mancuso   13:12

NM

Ok, thanks!

Nick Mancuso

right?

17 May 2020

Roman Ivanov   07:58

@Nmancuso , I a bit confused. This issue is about support of new syntax, we have to support it

Nick Mancuso   08:04

Maybe I misinterpreted the issue, but I read it as the user wants us to add a feature with the ability to ignore unsupported syntax (any, not just records) when checking a file. I thought he just used records as an example in this case.

I definitely plan to implement records, I just didn't think that this issue   08:07
directly related to records. More that it was related to being able to ignore ANY unsupported syntax.

Erik Silkensen   08:13

Oh, yeah I see what you mean Nick - but my interpretation is that we just want to support the record syntax. I agree that some feature to support arbitrary invalid syntax / tell checkstyle to not parse sections of a file seems like a separate request / out of scope for your project

Roman Ivanov   08:14

He wants to support it, and white it is not easy to support it completely, he suggesting to skip it in AST completely (workaround before good implementation)

Please consider it as option, if we can do this - it would be awesome   08:14

We can release it sooner, so user keep using us on new jdk, even we   08:15
ignore all violations in records

Erik Silkensen   08:26

I can imagine something like, define a special start/end comment that tells checkstyle parser to accept everything in between, which could be Java code like records but also really any text at all, i.e. the parser rule is basically ".*" - but is that what you were thinking Roman?

Or are you saying: be able to parse record syntax specifically, and it's ok if initially we don't find validations inside the records, but we are at least able to parse them?

Roman Ivanov   08:33

We add record as lexem, but in parser try to not build any AST below it. So it will looks like leaf node RECORD_DEF (no children AST)

Nick Mancuso

full support.

Roman Ivanov   08:43

Please consider to try, community will love it, all they care is to not crash

Nick Mancuso   08:45

@romanivanovjr thanks for taking the time to explain. It will be done.

Roman Ivanov   08:47

Please beware that it will require regression testing too, we should not damage anything for user on old jdk

Nick Mancuso   08:47

Noted.

18 May 2020

Nick Mancuso   13:13

Hello all, I just wanted to provide an update to make sure we're staying in touch. I will complete all but one of my finals this week, so I plan on taking a significant amount of time next week to finalize planning and make sure I'm ready to start coding on June 1st. I plan to focus on wrapping up my remaining PR's in Checkstyle this week and the beginning of next. In my free time, I have been looking through the projects that we are testing on already to see what, if any, of the newer JDK features they are using. Can any mentors offer advice for finding more projects that use the JDK 14 features? I have done quite a bit of Googling and hunting on Github, but I haven't turned up many larger/ stable projects that could be used for testing. What do you think would be a "good" number of projects to use in regression?

Roman Ivanov   15:57

Openjdk source code have a new syntax usage, contribution repo has details on it https://github.com/checkstyle/contribution/blob/master/checkstyle-tester/projects-to-test-on.properties#L18

Nick Mancuso   16:16

I didn't notice them the last time I was in contribution. Thanks for the new addition!

19 May 2020

Nick Mancuso   08:28

Hi everyone, I just wanted to point out that a PR exists for the Java 14 switch statement: https://github.com/checkstyle/checkstyle/pull/8185 I have checked this commit out, and it works well except for the bug that is introduced for lambda expressions.

NM

Nick Mancuso   22:19

Thanks!

---

24 May 2020

NM

Nick Mancuso   10:30

Hello all, I have just sent out a link to everyone for a chart I've created on google sheets (it's not complete yet). I plan to finish this by Tuesday so that I can update my proposal if necessary. Please, take a look and offer any feedback (or just make edits). Thanks for being patient with me through the last week so that I could get through finals, and thanks for any feedback you can offer. Here is the chart:

https://docs.google.com/spreadsheets/d/1SPw3azb7GDEcVhifmeMUTTS-iFS8D-PGknfgW3LBPvQ/edit?usp=sharing

---

26 May 2020

NM

Nick Mancuso   17:50

Hi guys, I'm finishing up my chart and looking deeper into implementation specifics right now. I see that strkkk has a PR for text blocks, is it OK to pick up where he left off for my project? Also, I plan to start with the "simple" record implementation on Monday to close issue #8023, then dive in deeper, unless there's any objections.

ES

Erik Silkensen   17:53

That all sounds good to me 👌

NM

Nick Mancuso   17:57

Since this would close the issue in my project for "records", should I raise a new issue for "full record support" or something?

---

27 May 2020

PB

Pavel Bludov   01:59

You're right. Once this issue is resolved, much more work should be done. We have a new type of Type. Many checks that handle classes should be able to verify records. Please create such issue.

RI

Roman Ivanov   07:41

@Nmancuso , please pay attention that due to nature of ANLRL and our model of continuous develpemnt and releases, and NO big PRs policy - we need to change your plan a bit.

or at least you should be aware that some parts like tests inputs can be   07:42
submitted in separate PRs

change in grammar can be small , but affecting a lot of Checks, so please 07.44
expect long phase of Checks update to handle new syntax

I remember we have some support of AST tree comparison in   07:46
checkstyle-tester project to detect regression in parser, did you become
familiar with it ?

While you have pauses, we need to find good sources of Inputs for new   07:48
syntax (jdk inputs, blogs, articles, ...)

such search could take a while ... it is better to start earlier   07:49

Nick Mancuso   07:50

@romanivanovjr You are referring to
https://github.com/checkstyle/contribution/blob/master/checkstyle-
tester/launch_diff_antlr.sh right?

Roman Ivanov   07:50

yes

Nick Mancuso   07:50

I have used it a few times now, yes.

I have found some input for records already, and plan to find more this   07:51
weekend. Should I post links to the content, or just have it handy to create
input files?

In reply to this message   07:55

So, only run ANTLR regression testing on Sevntu and standard checkstyle?
I'm glad to hear that, because it took me three days to run the test the last time
I used all projects!

Do you think my order of new features is good? records->text blocks-   07:59
>pattern->enhanced switch->pattern matching for instanceof->pattern
matching for enhanced switch ? I thought records would be the most
challenging, so I wanted to get started on it first.

Roman Ivanov   11:58

Please keep all links at Gdoc

I would start to do easy changes first, so I think instanceof might be good   12:01
and relatively simple

It is better to do most challenging at the end, so at least we will practice 12:02
testing of changes and acceptance of them on simple code changes

Nick Mancuso   12:10

Ok. Roman, thanks for all your guidance. I will update the GSOC and my
schedule as specified. Do we want to add basic records support at beginning
too?

Roman Ivanov

jdk14

Nick Mancuso   07:15

NM

I have raised an issue for full records support, for discussion related to post-Issue #8023 records implementation. It is here:
https://github.com/checkstyle/checkstyle/issues/8267

29 May 2020

Nick Mancuso   10:23

NM

I have raised a new issue to add the missing tokens to the end of the tokens list in java.g: https://github.com/checkstyle/checkstyle/issues/8274#issue-627420569

Nick Mancuso   11:51

NM

I have updated the chart at
https://docs.google.com/spreadsheets/d/1SPw3azb7GDEcVhifmeMUTTS-iFS8D-PGknfgW3LBPvQ/edit#gid=1115838130 to reflect the changes to the order of updates, and added some granularity to the tasks.

If the sequence of updates and tasks I've outlined in the chart are   11:56
acceptable, then I will amend the Gdoc (proposal) to reflect the new plan. I am adding some links to repos using the Java 14 features to the Gdoc now.

Roman Ivanov   15:44

RI

Coding starts on 1 June , on Monday :)

Nick Mancuso   16:10

NM

:-) Is there anything else I should do to prepare?

Roman Ivanov   16:51

RI

No, mentors are much slower than you

30 May 2020

Roman Ivanov   08:46

RI

hi mentors , do we want to close PR -
https://github.com/checkstyle/checkstyle/pull/7965 , I donot want PR to hanging for long time, we can always reopen it later on. Please review PR and suggest Nick how to proceed with it

Erik Silkensen   09:05

ES

Nick's last comment about setting that PR aside for now seems like a good idea to me, it is a lot of work done already and would be great to get it across the line at some point. I think it makes sense to focus on this project for now, and defer to you as far as closing the PR

jdk14

RI   **Roman Ivanov**   09:47

Erik, I recommend to review Nick status each day or each second day

NM   **Nick Mancuso**   13:32

@esilkensen I will make sure to tag you when submitting PRs.

@romanivanovjr I apologise for not being able to close #7965 before   13:42
GSOC. I had planned to, but it ended up being more complicated than I initially
thought. I put a lot of time into it, and I am determined to finish it before I return
to school in fall at some point.

RI   **Roman Ivanov**   13:44

It was good excecise to let you know that grammar change is not complicated,
but whole complexity is in side effects to existing Checks

NM   **Nick Mancuso**   13:45

I have definitely learned this!

RI   **Roman Ivanov**   13:45

Standard and sevntu Checks are not all Checks, there are thousands of
Checks in whole world as close source, and very custom to specific project

NM   **Nick Mancuso**   13:48

I never considered that. You mean that closed source projects modify
Checkstyle for their own use, in house?

RI   **Roman Ivanov**   13:58

They do not modify checkstyle, they have bunch of custom Checks

If we detect breaking compatibility in our Checks, you can be sure that   14:00
you will break bunch of others that we are not aware of. So you will make hard
choice for them to drop/update their Checks and use jdk14 features OR stay
on old checkstyle version

Please be aware that we need compatible changes :) , breakiy changes   14:03
are last resort

NM   **Nick Mancuso**   15:15

👌

RI   **Roman Ivanov**   17:38

Nick, please confirm that you know how to run full diff regression on all
standard Checks(javadoc and java) and sevntu Checks, and make sure it such
execution is not failing with OOM, We will need two report - AST diff, and no
diff from all Checks.

jdk14

Nick Mancuso   18:42

I have done regular regression and ANTLR regression, but never with javadoc. I will study how to run for Javadocs and make some test runs on this tomorrow, to be ready for Monday.

I have already been trying out some preliminary grammar and running   18:44
regression on it, for the past week. Where should I post results and discuss, in each issue?

Roman Ivanov   18:56

sure, if you have something already - please share and push on mentors

javadoc Checks are easy - just another Checks unlikely be affected but it 18:57
is better to test as some other them do in javadoc and out javadoc validation

Nick Mancuso   20:16

Ok, I will take care of this ASAP

31 May 2020

Nick Mancuso   08:34

I'm having an issue with running regression on this line only in `projects-to-test-on.properties` :
https://github.com/checkstyle/contribution/blob/0e1265db313d54a99f6e16284f0f15500c2f7e4f/checkstyle-tester/projects-to-test-on.properties#L7

I also tried using older branches that I have successfully ran regression   08:37
on before, and it still fails with: `NoViableAltException occurred while parsing file /home/nick/IdeaProjects/contribution/checkstyle-tester/src/main/java/checkstyle/src/test/resources-noncompilable/com/puppycrawl/tools/checkstyle/gui/mainframemodel/InputMainFrameModelIncorrectClass.java` I've tried to modify this line in `projects-to-test-on.properties` to recursively ignore all subdirectories, and explicitly added this file to the ignore list, and neither attempt has worked.

From what I can tell, this file isn't new, and should be ignored anyway, so 08:39
I was wondering if anyone knew what might be going on before I start digging in.

Nick Mancuso   08:59

I just tried to delete my clone of contribution and re-clone, it appears that the exclude folders aren't being excluded

Erik Silkensen   09:00

I will check it out too - can you share the full command you're running?

Nick Mancuso

others that I did successfully run regression on previously and they also failed.

Erik Silkensen   10:14

ES

Ah, it looks like diff.groovy passes ignoreExcludes to launch.groovy, so I think that explains why it is trying to process that file -- but I wasn't able to reproduce this using clean checkouts of master on the checkstyle and contribution repos. I assume this exception is only on the patch branch? Or what if you start from the latest master on checkstyle, and then create another local branch from there that's identical to it, and try the diff report on that?

Richard Veach   10:18

RV

diff groovy ignores any excluded files as it is regression and should process all files for any differences.

if the exception is new, it will only have a green and no red. If it has a   10:18
green and red, then the exception changed somehow

if the exception is halting the regression run, then your configuration is   10:20
missing the halt property from the config

Nick Mancuso   10:26

NM

Richard that's exactly what it is. Since google_checks.xml was updated, I made a new config file, and failed to add it. Thank you so much, you likely saved me hours.

Richard Veach   10:27

RV

no problem, I would look at the normal regression config as you should include those top properties in every run.

Nick Mancuso   10:30

NM

Ok, thanks again! I wanted to have tests to share for instanceof today and tomorrow, and some take so long, I wanted to get them running ASAP

I'm going to run google_checks and checkstyle_checks, then run ANTLR 10:32 diff on checkstyle_checks, separated into three runs. Does that sound good?

Oh, and javadoc   10:39

Roman Ivanov   12:17

RI

@Nmancuso , please put a link of "JAVA 14 SYNTAX FEATURES" to proposal Gdoc. To keep single document with all details. I thinks that it will be much better to have time table in Gdoc , as in spreadsheet commenting is not very convenient and visible

you do not plan to return back to some parts, so such diagram view is not 12:19 very convenient as it keep a lot of unused space. So I think if you make simple table/list in Gdoc it will be much better. Commenting will be easier.

I have concerns on your first week plan, I need to heavily comment it   12:21

Are two week blocks OK? 12:27

Roman Ivanov 12:29

RI

yes

Nick Mancuso 12:29

NM

I'll have it done in ten minutes

Roman Ivanov 12:42

RI

Hey all, so coding period is about to start. Preparation period did not result in strict plan, sad. So it mean for me that we are not sure on what to focus at what time, partly my bad as i did not have enough time to lead this process. So to keep development on leash I am imposing each day report in Telegram for @Nmancuso (visible only for mentors). We will use it for 1-2 weeks, to make sure we are on track.
In addition to telegram @Nmancuso have to do 2 time per week report in public mail-thread (requirement from Google). @Nmancuso , please start mail thread on first day and share what you did on first day.

lets make mail thread with title "GSOC - jdk14 - June" (each month will 12:44 have separate mail-thread) , and place some report on Monday and Thursday.

Nick Mancuso 12:45

NM

So, I will report my plan for the day every morning in Telegram, then report progress in mail thread Monday and Thursday, correct?

Roman Ivanov 12:47

RI

while plan in not strict, please report plan and what was done/problems/failures/..... . Ones plan become more clear, you can just drop us your status.

To make mentors be clear when you stuck and need help or advice 12:47

Nick Mancuso 12:48

NM

Ok, hopefully not too many failures!

I will definitely communicate if I need help. I only forsee a few roadblocks, 12:50 which I plan to investigate further tonight and post in appropriate issues by tomorrow.

Roman Ivanov 13:37

RI

@Nmancuso , after that long time in regression testing, you might find some good automation for this, please do not hesitate to contribute it, we can add such regression testing in CI probably to ease this process during GSOC and after gsoc.

Nick Mancuso 13:39

NM

That is a good idea, I did plan on writing at least a bash script to help out.

Good morning everyone. I am getting started on writing input files for records and instanceof, and plan to submit PRs for both today. I will also submit a PR for JDK 14 test item in Travis. Should I also submit PR for initial record grammar, even if the grammar still stumbles on something like `LogRecod record;`, just so mentors can monitor my progress with that task?

Nick Mancuso   07:00

I just want to provide an update regarding regression reports: I had planned to provide several today, so I spun up two EC2 instances to keep up with the demand and not tie up my desktop. I've been having some issues completing the reports with these projects: https://gist.github.com/nmancus1/a89d2614131e8c0a44f3b31cb6775598 and this config, based off of checkstyle_checks.xml: https://gist.github.com/nmancus1/d11ca3d587eb487b15064456d21072f1 Is there any outstanding issues with either one of these files? I have run all of these tests, with all the same projects selected, at least 20 times before with no issue (on my desktop, not in the EC2 instance)

Roman Ivanov   07:23

> Should I also submit PR for initial record grammar, even if the grammar still stumbles on something like LogRecod record;, just so mentors can monitor my progress with that task?

Please do draft PR - https://github.blog/2019-02-14-introducing-draft-pull-requests/

Nick Mancuso   07:24

Ok!

Roman Ivanov   07:27

> I've been having some issues

Please be exact, share full log of execution or exception or .... . Please create issue in contribution repo, to keep all context in one place and share link here and request help, keep pinning mentors each day on problems you have.

Nick Mancuso   07:29

Ok. I am trying one more run, then I will create an issue.

Nick Mancuso   13:11

Hello all, today I have created two PR

Hello all, today I have created two PR's, (input files for records and   13:16
instanceof with pattern matching), and two draft PR's (rocrd and instanceof
grammar. I think I have regression testing all set up in my two EC2 instances.
The problem was likely that they didn't have enough memory, I started out with
the 4 gig instances, but I went up to 10 now, with a compute specific setup. I
am now running AST structure regression reports and check behavior

Roman Ivanov 13:17

Please do not forget to start open for all mail thread that track your status and progress

Nick Mancuso 13:19

In google groups?

Roman Ivanov 13:19

Yes

Nick Mancuso 13:21

I'm typing it right now. Is there anything else I can work on for tonight? I'm not done for the day, but I wanted to check in before it was too late for Pavel.

Roman Ivanov 13:23

Good first day, thanks a lot.

Nick Mancuso 13:24

I wish I could've had my EC2 instances running reports all day instead of constantly failing... but other than that, not bad. You're welcome.

Erik Silkensen 13:27

Nice work, Nick - I will check out your PR's later this evening.

Nick Mancuso 13:27

Oh, I made a bucket on AWS for reports if we run into problems with github.io.

Nick Mancuso 13:49

Email thread here: https://groups.google.com/forum/#!topic/checkstyle-devel/-_se7-Pjrag

Roman Ivanov 21:13

Nick, it would help a lot if you clearly define in your status report(mail, telegram) exact list of PR links, to make it very clear what is mentors Todo :) list

2 June 2020

Nick Mancuso 05:33

Morning status report: today I plan to work on the automation of reports, perfecting the preliminary record grammar, posting reports, providing more information in issues. I need help from mentors with closing the three PR's that are open (https://github.com/checkstyle/checkstyle/pull/8284 , https://github.com/checkstyle/checkstyle/pull/8289 , https://github.com/checkstyle/checkstyle/pull/8290) and for them to take a look

one, with a better email? I misunderstood the intent of the email thread, I thought it was supposed to be blog-like.

Roman Ivanov   06:42

No need to create new thread, you can keep it for blog like purpose if you want, but please make some section in it for a status (what is done and what is blocking you, pinning mentors to respond to merge, question, ...)

You can take a look at records grammar in pmd project, antlr4 project   06:44
also have example of Java grammar but I am not sure of status such grammar

Nick Mancuso   15:03

Evening status report: Today I have almost finished a script to automate check regression testing for my EC2 instance, found here: https://gist.github.com/nmancus1/40b52daac298266c7f1713d0af1ebc2a. It is by no means perfect, but I intend to keep using it throughout the summer and improving it as I use it more. I have posted check regression reports for initial records grammar https://github.com/checkstyle/checkstyle/pull/8293 and enhanced instanceof grammar https://github.com/checkstyle/checkstyle/pull/8294 . I have continued to try different approaches to the records grammar. I will post the two examples of code that I am struggling to parse correctly, and push my latest attempt to the draft PR here: https://github.com/checkstyle/checkstyle/pull/8293.

It looks like instanceof will be very straightforward to complete, I wonder   15:08
if we should pursue that and keep trying different approaches for the record grammar. I plan to complete and post AST regression testing and Javadoc regression testing by noon tomorrow for instanceof. I had hoped to complete the initial grammar for records today, but it is not as simple as I thought, since "record" hasn't always been a keyword.

I plan to put in a few more hours this evening, please let me know if   15:09
there's anything else I can take care of today.

Erik Silkensen   18:06

Sounds good, thanks Nick - I will check out the draft grammar PR's tonight. I think it sounds like a good plan to me to take them one at a time, if instanceof is straightforward, it should be a good way to get things started!

Nick Mancuso   18:11

Thanks! The record grammar is pretty rough, but I thought it best to share early and often to keep everyone in the loop.

Nick Mancuso   19:12

Mentors, I have posted a google doc with a few questions that I have about testing just to make sure there is no time wasted. You can find the doc here: https://docs.google.com/document/d/1KxyTPtL3Ers0c505_Dltmi2Zs9elG5kv8tJMdqgTL6o/edit

3 June 2020

jdk14

AST regression report to instanceof PR, and write UT's for instanceof once PR's for input are merged. Also, I plan to continue work on my check regression script. I see that I was added as a collaborator last night, thank you! I am honored. For the smaller PR's, once all is green and two mentors have approved, may I merge them? If so, all I need from mentors today is to take a look at the google doc I've posted and respond to my points. https://docs.google.com/document/d/1KxyTPtL3Ers0c505_Dltmi2Zs9elG5kv8tJMdqgTL6o/edit

Erik Silkensen 07:08

Sounds good, I had the same question about merging PR's - that said, I wouldn't let the separate PR for instanceof input block you from writing the UT's, IMO it's small enough to include in the grammar change PR or in any case can be kept in sync / rebased once a separate PR is merged.

For your testing questions in the google doc, I'm hoping Roman, Richard or Pavel might comment having more knowledge of it than I do... like for 2 about sventu - but your plans sound good to me.

Roman Ivanov 07:57

@Nmancuso , I replied you in doc (if you agree, embed comments in text to make it clear for all). I do not like Gdocs as we loosing some valuable context to refer to it later on. As we finish discussion in gdoc we should move content it to some issue to make it as plan (issue description is modifiable, and I can remove comments to keep it not messy) - please do.

@Nmancuso , there is huge pressure on me now, as all all project need 08:01 initial guidance on how exactly operate, I can not follow all conversations, please pin us, if somethig is not done(review/merge). Please in your EOD report - list links to PR that mentors need to review, keep pinning them (mentors/me). If I am not responding it might be I missing something, or missed due to overload.

Nick Mancuso 08:02

@romanivanovjr if it's ok, I was going to update contribution/checkstyle-tester documentation with this information, and I will also put in issue as well

Roman Ivanov 08:03

yes, please. We always open to improvment.

Nick Mancuso 15:34

EOD report: today I updated the instanceof PR (https://github.com/checkstyle/checkstyle/pull/8294), all is green and it is ready for review. We still need to get https://github.com/checkstyle/checkstyle/pull/8290 , https://github.com/checkstyle/checkstyle/pull/8284 , and https://github.com/checkstyle/checkstyle/pull/8289 merged. I also submitted a PR for contribution, but I moved it to draft status, because I would like to add more information about testing with sevntu checks, once I get that figured out.

update to ~~https://github.com/checkstyle/checkstyle/pull/8290~~ .

I really have only two issues blocking me right now,(1) I can't figure out   15:37
how to run sevntu checks. @romanivanovjr I followed the travis script, but
there is no "sevntu-checks" item in travis.sh anymore. (2) How to avoid parsing
"record" as a `LITERAL_record` when it is a variable name. I have tried many
different approaches to no avail.

I plan to put in an hour or two later tonight, please let me know if I can do 15:38
anything else.

Oh, and I also added the AST regression reports (three of them) to PR   15:45
https://github.com/checkstyle/checkstyle/pull/8294

Roman Ivanov   20:44

RI

https://github.com/sevntu-
checkstyle/sevntu.checkstyle/blob/master/.ci/travis.sh#L55 there is sevntu
build in Travis

Roman Ivanov   22:27

RI

@pbludov , please review prs with inputs. @Nmancuso , please send me gist
with all logs of what you tried with sevntu

Erik Silkensen   22:44

ES

Thanks Nick, and nice work on the instanceof PR. I suspect Roman will be
able to get you squared away with sevntu, but if you need a second set of eyes
I will be happy to jump in as well.

My initial thought on "record" is that it may be a special case where, since it is
sometimes a keyword and sometimes not, we probably don't want it to be a
new LITERAL_record token.

I wonder if it'd be possible to write the new record grammar rules using a
semantic predicate to check whether an IDENT token's value is "record" - what
do you think? (see the docs/examples here, if you haven't already:
https://www.antlr2.org/doc/metalang.html#SemanticPredicates)

4 June 2020

Nick Mancuso   06:12

NM

Plan for today: Get sevntu testing figured out, continue trying to get records
working, update PR in contribution with sevntu testing information. I will also
post in the email thread this afternoon. @esilkensen thanks for the ideas, I
have tried using the semantic predicate, but not the removal of the
LITERAL_record token. I will try that today and report back. @romanivanovjr I
apologize, I was confused and thought we were talking about travis for regular
checkstyle. I will reread through our conversation yesterday and check out that
link. I will document what I've tried and supply it in a gist for you.

Nick Mancuso

Basic Single Report , and the checks-sevntu-error.xml config.

Nick Mancuso   09:45

NM

Update: sevntu check regression report is up and running.

Nick Mancuso   13:30

NM

@esilkensen you rock. Removing LITERAL_record did the trick!

Erik Silkensen   13:31

ES

Nice, that is great to hear!

Nick Mancuso   15:13

NM

Instead of posting EOD here, I am composing another email for the thread right now.

Nick Mancuso   15:28

NM

https://groups.google.com/forum/#!topic/checkstyle-devel/-_se7-Pjrag

5 June 2020

Nick Mancuso   05:18

NM

Plan for today: general housekeeping, finish PR in contribution, add code that is failing to parse in CI from records PR to records input file, modify records grammar to handle it. I have some check regression reports for current records grammar, I will post them. I am also going to run some check regression reports on JDK14 for instanceof.

Erik Silkensen   06:56

ES

Sounds like a good plan, nice work this week!

Nick Mancuso   07:02

NM

@esilkensen thank you. If you have a moment, can you take a look at https://nmancus1.github.io/full_diff-initial-pattern-def-grammar_6_1/diff_sevntu_checkstyle and make sure that I ran the report correctly? I wanted to get the contribution documentation finished up today.

Nick Mancuso   09:24

NM

Coverage for records is now 100%, I'm removing the jacoco report.

Erik Silkensen   09:31

ES

In reply to this message

I actually haven't run one of those myself, but looks good to me 👌

Nick Mancuso   10:21

NM

Can a mentor share the werker failure with me please?
https://github.com/checkstyle/checkstyle/pull/8293

seeing - maybe we need an admin or someone with higher privileges

Richard Veach 10:32

NoErrorTest - Orekit
Caused by: java.lang.IllegalStateException: /pipeline/source/.ci-temp/Orekit/src/main/java/org/orekit/bodies/JPLEphemeridesLoader.java:499:5
4: unexpected token: record

Roman Ivanov 10:40

@esilkensen , please login to wercker GitHub account and share with me your account , I will give you access

Nick Mancuso 10:51

It's normally taking me about 4 hours to run check regression reports for all checkstyle checks and javadoc checks on all projects except JDK14. I've had a JDK14 **only** check regression test running for about five hours now, and it's still on the first config file(javadoc). It appears that it's still running, but I think it might have died silently. Has anyone ran check regression with JDK 14 before? Should it take much longer than normal?

Richard Veach 10:52

theres an issue with newer JDKs. you may have hit it

https://github.com/checkstyle/checkstyle/issues/8291 10:52

Nick Mancuso 10:54

I'll look into it

Nick Mancuso 11:30

https://app.wercker.com/checkstyle/checkstyle/runs/build/5eda863e7b63df001a18d7e3

Nick Mancuso 14:25

EOD: made more headway on records grammar, addressed all points from Richard in https://github.com/checkstyle/contribution/pull/484 , added more troublesome code to the records UT, and began researching solutions for https://github.com/checkstyle/checkstyle/issues/8291 .It seems like once the number of nested lambdas approaches 25, the issue arises. I'm going continue to look at this problem, since it is critical that I am able to test on JDK14. Since it looks like I will need to continue to work on the records grammar, I am going to hold off posting the check regression reports that I made last night, and run more after the grammar improves.

8 June 2020

Nick Mancuso 05:24

Plan for today: provide latest records implementation check regression reports, provide latest records AST regression reports, try to refine records grammar,

https://github.com/checkstyle/checkstyle/pull/8204 merged

I put a few hours into 05:29
https://github.com/checkstyle/checkstyle/issues/8291 this weekend, and I
haven't been able to determine what the cause of the issue is. I will take
another look this week. Should we add the file that is causing the issue to the
excludes in projects-to-test-on.properties file for now?

Pavel Bludov   09:43

@Nmancuso this is test case for https://bugs.openjdk.java.net/browse/JDK-
8055984
This bug was fixed in the compiler, but not at the grammar level.

Nick Mancuso   10:16

Should we just exclude it then?

Pavel Bludov   11:11

I agree. Ignoring this test would be a reasonable solution. This sample is
synthetic and should not occur in real code.

Richard Veach   11:12

Don't forget difference regression ignores the exclude list

You should just not run regression on these affected projects   11:13

Nick Mancuso   11:13

Thanks for the reminder!

Richard Veach   11:17

As for why the issue happens, it basically relates to how antlr decides what the
next item is (variable, method, etc...). Antlr isn't that smart and basically has to
guess based on the definitions given to it. When it guesses wrong, it has to
unwind to a certain point and guess again. The lag occurs based on how much
guessing it has to do and how much unwinding it has to do.

If 2 guesses are very similar, it compounds the issue more and it   11:18
increases the guesses and unwinding when it fails.

Something similar happened to the javadoc antlr, and we forced the issue 11:20
through by telling antlr how to guess correctly. We didn't have anyone with
enough experience to resolve the issue.

See   11:21
https://github.com/checkstyle/checkstyle/issues/1064#issuecomment-
200387660

Nick Mancuso   11:27

I guess that much logic in an action is undesirable, right? That's why you call it
"forced"?

Nick Mancuso   16:03

NM

EOD/Monday report: https://groups.google.com/d/msg/checkstyle-devel/-_se7-Pjrag/4PkQss3aAwAJ

Erik Silkensen   16:42

ES

Thanks Nick - I don't think there is anything you need to do with the instanceof PR until Roman/Richard has a chance to review. I am taking a closer look at the records grammar now and will get back to you, hopefully later tonight!

Nick Mancuso   16:46

NM

Awesome, thank you very much. I'm really not happy with what I had to do to get it working (all the logic in the lexer) but I've tried about 50 different variations of solutions I've found online, and nothing works 100%. If you think it would help, I can share (in a gist) what I did that worked pretty good for records but messed up the rest of the grammar, maybe you can tell what is breaking everything else.

Erik Silkensen   16:47

ES

Sounds good, yeah if you want to share a gist with that, I will check it out too

Nick Mancuso   18:00

NM

I just pushed the whole branch, and left a comment for you here: https://github.com/checkstyle/checkstyle/pull/8293#discussion_r437079535

Erik Silkensen   23:57

ES

Nice, thanks! Yeah that branch is what I was wondering about... I have started looking at this, will pick it up again

<div align="center">9 June 2020</div>

Nick Mancuso   05:55

NM

Plan for today: finish up check regression report script, submit PR. Try another approach to records grammar (having lexer pass all keywords to the parser as keywords, then have a parser "id" rule). Hopefully get https://github.com/checkstyle/contribution/pull/484 merged, or more feedback on how to improve.

Nick Mancuso   07:11

NM

Mentors, do I need to create an issue for diff.groovy automation before submitting the PR for my script?

Roman Ivanov   07:29

RI

It it always good to create issue and put there all details with root reasons why such update is required

I am prepared to submit a PR as soon as this is approved.   09:55

Nick Mancuso   12:24

@esilkensen I nailed the records grammar, finally. I just wanted to let you know so that you don't waste anymore time looking into it.

Erik Silkensen   12:25

Awesome! Thanks for the heads up, I will look forward to checking out what you came up with 👌

Nick Mancuso   12:46

https://github.com/checkstyle/checkstyle/issues/8308 I can submit a PR for this today, if approved.

Roman Ivanov   13:09

Mentors please read
https://github.com/checkstyle/checkstyle/pull/8290#issuecomment-641541734

Nick Mancuso   13:30

@romanivanovjr I was wondering about that one.. after our discussion for instaceof input. Should I edit the PR?

Erik Silkensen   13:32

In reply to this message

Noted, thanks

Nick Mancuso   15:35

EOD: Finished up check regression report script, created two issues:
https://github.com/checkstyle/checkstyle/issues/8308 and
https://github.com/checkstyle/contribution/issues/486, and got a working record grammar that I'm satisfied with. I still need approval for both issues, if mentors could help with that.

Erik Silkensen   16:28

Sounds good, if the changes for #8308 are small, I think it would be fine to fold them in as part of your PR for records without necessarily creating a separate issue. If they are significant then certainly better to separate it out (and of course I defer to the admins either way)

10 June 2020

Nick Mancuso   06:14

Plan for today : post check regression reports to
https://github.com/checkstyle/checkstyle/issues/8308, run AST regression reports for #8308 and get them posted today or tomorrow. Unless any mentors have specific tasks in mind for me, I'm going to begin studying groovy and converting my check regression report script (written in BASH now) to groovy.

features?

Roman Ivanov   06:16

yes, never be blocked by slow mentors

Nick Mancuso   06:19

@romanivanovjr What is the "right way" to continue the work of another? See https://github.com/checkstyle/checkstyle/issues/7103 and https://github.com/strkkk/checkstyle/commit/7f1ae58a91726e655530ed973c7d 16a9a4de3fa4

Just make a new branch from the existing commit?   06:20

Roman Ivanov   06:30

yes, you can always send PR that contain commit from pending to merge PR, just mention in PR description that some set of commits to not review as thay are part of something else. After merge of old PR, you will simply rebase and be ready for final review with single commit.

Nick Mancuso   06:47

The old PR was closed, does that make any difference?

Roman Ivanov   06:52

if some PR is merged already, just create new branch of latest master

Nick Mancuso   06:55

strkkk closed it: https://github.com/checkstyle/checkstyle/pull/7285#issuecomment-585770863 (under your comment), it wasn't merged in this case.

Roman Ivanov   06:56

it was closed just to keep size of PRs on leash

take this commit, reuse it, and continue it   06:57

size = amount   06:57

now you are dedicated to jdk14 feature , we can have as much PRs as   06:58
reasonable to be open and active till end of GSOC

Nick Mancuso   06:59

Ok, thanks for explaining. I will start on this today, unless you think I should look at enhanced switch first.

Roman Ivanov   06:59

did you make progress with testing tool ?

did you provide all regression reports for records and instanceOf ?   07:00

instanceof. I was waiting to make sure that https://github.com/checkstyle/checkstyle/issues/8308#issuecomment-641995056 gets approved, so I can use that in records grammar, then I will make reports for that.

I am currently running reports (check and AST) for   07:03
https://github.com/checkstyle/checkstyle/issues/8308#issuecomment-641995056 and will have both posted by EOD

Roman Ivanov   07:03

please keep pinning mentors to unblock you, to approve issues and and review PRs and ......

Nick Mancuso   07:06

Issues that need approval:

https://github.com/checkstyle/checkstyle/issues/8308#issuecomment-641995056 ,
https://github.com/checkstyle/contribution/issues/486#issuecomment-641705953

PR's that need review: https://github.com/checkstyle/checkstyle/pull/8294 07:07
, https://github.com/checkstyle/contribution/pull/484

Nick Mancuso   07:40

All reports posted to issue #8308
https://github.com/checkstyle/checkstyle/issues/8308#issuecomment-642053460

Erik Silkensen   07:55

I added the approved label to the id issue - I would also echo what Roman said about being blocked, for example if you have code ready for that id rule or something you believe is the right way forward, I'd say go for it, push it to your branch, open a new PR, etc. - especially if you don't have something else you're working on, don't let an issue approval stop you from starting. Anyway, I will look forward to checking out those grammar updates, and let me know if there's anything I can help out with in the meantime!

Nick Mancuso   08:23

PR posted: https://github.com/checkstyle/checkstyle/pull/8313

I just realized that my master that I ran the reports on was two commits   08:27
behind my IDENT-to-id branch. It caused a bunch of differences from those two commits in between (JavadocMethod and EmptyLineSeparator). Do I need to rerun all of these reports?

Disregard that question, I just realized I missed a few IDENTS in the   08:37
parser anyway, so I will update the PR and rerun the reports.

Nick Mancuso

Nick Mancuso 13:54

I've tried to specify another path in the "includes" header to include the generated files in pom.xml, but it hasn't worked.

Nick Mancuso 14:42

EOD report: I've pushed what should be the final update needed to get https://github.com/checkstyle/checkstyle/pull/8313 merged, except that I need to somehow show that this change doesn't impact performance, which I am not sure how exactly to quantify. Also, I've merged that commit into my records grammar implementation, and updated the grammar to work with classBlock, so that we can easily incorporate new and old checks to be compatible with the records construct. This grammar works great, except I cannot seem to get coverage above 55% for `recordDeclaration` . All of my other rules have 100% coverage. Unfortunately, since I can't seem to view the source code report in Jacoco for GeneratedJavaLexer.java, I can't determine which branches in `recordDeclaration` aren't fully covered. I am going to run reports for the new records grammar this evening, to post tomorrow. The new reports for #8308 are almost done, and I will post them this evening.

This coverage issue is frustrating, since I've manually traced the program 14:43 and can confirm every branch is being hit except for the default "NoViableAltException" branch.

Richard Veach 17:52

I thought we did fix jacoco so you could see coverage for the lexer. I know I can see coverage in Eclipse using its plugin.

Be wary of code coverage for the generated class. It likes to build a lot of 17:54 dead code because it is auto generated and isn't very well optimized.

Nick Mancuso 19:46

I need to see coverage for the recognizer, though. I can see coverage in IntelliJ, but that's Intellij's analysis, not Jacoco's. Intellij shows total coverage except for the default case. How can I proceed? The generated code is actually very simple in this case, and only has three branches including the default. If I drop the required coverage from 0.73 to 0.72, I can pass `mvn clean verify` with no problem. If we have to stick with 0.73 as our minimum, could I try to add some test cases for something unrelated and increase coverage for the entire file to offset this? It seems like 55% coverage is pretty decent for most of the more significant rules.

Reports for #8308 have been updated. 19:49

Roman Ivanov 22:19

I remember https://github.com/checkstyle/checkstyle/wiki/How-to-run-certain-phases-and-validations#how-to-generate-ut-coverage-report this worked for me, and even played with numbers over such generated code, very recently.

ES

> I need to somehow show that this change doesn't impact performance, which I am not sure how exactly to quantify

What do you think about timing the checkstyle execution time, base branch vs. patch, for a couple of projects?

> How can I proceed? The generated code is actually very simple in this case, and only has three branches including the default. If I drop the required coverage from 0.73 to 0.72, I can pass `mvn clean verify` with no problem.

On the one hand, my personal feeling is if your changes are well tested, chasing after 1% coverage isn't necessarily a good use of time and I'd probably vote to lower the number to 0.72. That said, it looks like your latest grammar introduces a number of new, uncovered branches where for example "LA(2)==IDENT" turns into "LA(2)==IDENT||LA(2)==LITERAL_record" - I suspect you could cover some of those by using "record" as a normal identifier in different ways - and maybe those could be good additions to the test case, anyway.

Nice work figuring everything out -- I think you're making great progress! 00:33

Nick Mancuso   05:28

NM

Plan for today: time execution of checkstyle on several different projects using master and my records grammar branch, to present to mentors, and address items in PRs. I am also going to dig in more to the coverage issue.

Also I will provide new reports for records grammar.   05:28

Nick Mancuso   06:08

NM

Reports for records grammar:
https://github.com/checkstyle/checkstyle/pull/8293#issuecomment-637073067

Roman Ivanov   06:10

RI

Please do not go too crazy in coverage of generated code, we are ok to make some noncovered code, just share what you did to try to cover them

Nick Mancuso   06:16

NM

@romanivanovjr the instructions that you pointed me to worked great to make a report that shows the source. Thanks! Everything I've added is covered 100% except for recordDeclaration, and it is 55%, which is typical for some of the more complicated AST structures. I am sharing a coverage report from recent master and my changes to the above link. I have one more idea I'm going to try to get coverage up. If that doesn't work, can I drop the coverage minimum in pom.xml?

Roman Ivanov   06:51

RI

Yes, you can, but do not forget to share all details to mentors to verify

Nick Mancuso

the change in pom.xml stay as separate commit?

Nick Mancuso   14:49

NM

EOD report: I have added some timed execution averages to #8293 and
#8313. I have also added details for dropping the minimum coverage in #8293,
however, I will likely be reverting that commit if everything works out with
Richard 's suggestion, which I will apply tomorrow. I am running reports again
for records grammar, which I will be posting tonight or tomorrow morning. I
updated #8313 to address @esilkensen 's items. I am posting to the email
thread as soon as I'm done writing this.

Nick Mancuso   15:21

NM

Email thread post:https://groups.google.com/d/msg/checkstyle-devel/-_se7-
Pjrag/6IkOWaX7AwAJ

Erik Silkensen   20:28

ES

Nice, thanks Nick. I think we just need Roman and/or Richard to review
instanceof to get that across the line.

I don't think I am able to add labels / approve issues in the contribution repo -
but IIRC that one doesn't have the same CI restrictions as the main checkstyle
repo. I think you can go ahead with converting your script to groovy and
sending a PR!

12 June 2020

Nick Mancuso   04:25

NM

Plan for today: Make one more attempt to get coverage up for records, post
reports for records, make records ready for review. I will also attend to general
housekeeping, and address all items from mentors in all PR's. I am going to
put the groovy script on the back burner for now, and work on it throughout
next week. Since I have fully automated the check regression report
generation already with my bash script, this isn't a high priority unless
someone else needs it.

Nick Mancuso   14:26

NM

EOD: I have addressed all items in all PR's, and I am just waiting on AST
regression report for records PR to finish (it's taking longer than expected), I
will post it as soon as it is done and mark the records PR ready for review. I
have posted all check regression reports to records PR. After that, all PR's will
be ready for review. https://github.com/checkstyle/checkstyle/pull/8313
https://github.com/checkstyle/contribution/pull/484
https://github.com/checkstyle/checkstyle/pull/8294
https://github.com/checkstyle/checkstyle/pull/8293

Nick Mancuso   14:57

NM

Starting Monday, I will begin hacking on TextBlocks grammar, unless I have
items to address for current PR's. I don't think it's a good idea to start on the

any objections.

Nick Mancuso   15:12

NM

Does anyone know about the GSOC evaluations? Does this get sent out as a form via email?

Nick Mancuso   17:57

NM

AST regression report posted to records: https://github.com/checkstyle/checkstyle/pull/8313 . It is now marked as ready for review.

I'm running another AST regression test on another project. I will try Java 17:58 14 first, and if that hangs, then I'll try something else.

Roman Ivanov   22:04

RI

GSoC evaluation is a simple form in GSoC web site, Google send email as reminder

13 June 2020

Roman Ivanov   06:43

RI

Please add this record case to our inputs https://twitter.com/graemerocher/status/1270640823168831496?s=19

Nick Mancuso   06:49

NM

Ok, I'm looking on github for the exact file right now

Which import should I use? Does it matter?   06:54

Roman Ivanov   06:59

RI

We just need to put word record in any possible place, just to make sure it works

Nick Mancuso   08:29

NM

@romanivanovjr can you take a quick look at https://github.com/checkstyle/checkstyle/pull/8294 , https://github.com/checkstyle/checkstyle/pull/8293 and https://github.com/checkstyle/checkstyle/pull/8313 to see if there's anymore tests I need to provide? I'd like to run them all weekend if I can.

Nick Mancuso   08:49

NM

@romanivanovjr I mean test reports

15 June 2020

Nick Mancuso   04:46

NM

Plan for today: post AST regression report for records, address all items in PR's, submit PR for TextBlocks input, begin hacking TextBlocks grammar.

jdk14

The JDK 14 AST regresion test for records helped me find a few more test cases we need, I'm going to add them to the inputs test and tweak the grammar again.

Nick Mancuso   07:47

Ok, I have amended records grammar to work with all failed examples from JDK14 AST regression test. I am changing records PR back to draft status, sorry for any confusion.

Nick Mancuso   08:18

Text blocks input: https://github.com/checkstyle/checkstyle/pull/8320

I will mark records PR as ready for review once I have rerun all tests and   08:20
posted the results.

Nick Mancuso   09:04

A quick fix here for consistency:
https://github.com/checkstyle/checkstyle/pull/8321

Can a mentor provide me with wercker failure please: 09:11
https://app.wercker.com/checkstyle/checkstyle/runs/build/5ee796ef7b63df001a
1cc39b

Richard Veach   09:14

NoErrorTest - Spring Integration
Could not GET
'https://oss.sonatype.org/content/repositories/snapshots/org/jxmpp/jxmpp-
core/maven-metadata.xml'. Received status code 502 from server: Bad
Gateway

looks like a random failure   09:14

Nick Mancuso   09:15

Ok, thank you. Can we just restart CI?

Richard Veach   09:15

I did, but sometimes it may take some time for a random failure to work itself
out

Nick Mancuso   15:03

EOD found here: https://groups.google.com/g/checkstyle-devel/c/-_se7-
Pjrag/m/MNOSCP7jBwAJ

I will post all reports to records PR within the next few hours, they should   15:04
be done by 8:30 at the latest.

*8:30 EST   15:04

16 June 2020

https://github.com/checkstyle/checkstyle/pull/8293. I can run them today if so.

Nick Mancuso   14:52

NM

EOD: all reports posted, and have a working implementation of text blocks grammar. Mentors, please see https://github.com/checkstyle/checkstyle/issues/7103#issuecomment-645028044 and comment on the AST. I tried several different approaches, including trying to use the hidden lexer channel and a separate tokenstream in the parser, which worked to varying degrees of success, but added a lot of code to JavaParser.java, which I didn't like. My final approach seems to be the most simple and error-free; I have entirely implemented this solution within the grammar. I need to work on it a bit, but I think that the basic skeleton is good, and I should be able to perfect it during the week and begin running regression tests if everyone approves of the AST.

17 June 2020

Erik Silkensen   01:04

ES

Thanks Nick -- hopefully we can catch up to you with some of those PR's waiting to merge (sorry I've been swamped lately, am trying to get onboard at a new job). "simple and error-free", I like the sound of that ☺ That does sound like a nice approach if it can be done in the grammar

Nick Mancuso   06:01

NM

Plan for today: keep working on text blocks grammar, and respond to all items in PRs.

Nick Mancuso   14:14

NM

EOD report: not really the end of the day, I am picking up later (to finish PR items), but I just wanted to check in. After spending about half of the day working on the hacky text blocks implementation that I came up with on Monday and Tuesday, I have decided that I should use a separate token stream for the text block content. I really liked @pbludov 's suggestion here: https://github.com/checkstyle/checkstyle/issues/7103#issuecomment-645266909, and the best way to implement this is to use a token stream selector. I have done this, and created a "java14textblock" grammar. Even if we don't do much more with this now, in the future, we could easily implement things like language-specific nodes, clear out extraneous whitespace, parse content, and much more. Since I have been keeping up fairly well so far, I would like to take my time on text blocks and really make this a solid addition to our Java14 syntax support, unless mentors disagree. It will take me a couple more days to get a text blocks PR submitted, likely next week, if that's OK.

Nick Mancuso   15:09

NM

Also, I ran another AST regression report for records, on JDK14, just to make sure all bases are covered: https://nmancus1.github.io/initial-record-grammar_check_diff_reports_2020_06_15/AST_diff_6-17_jdk14 . I have posted this report to the PR as well.

NM

Plan for today: keep hacking on text blocks grammar, address items in PR's, post in email thread. I have completely rewritten my text blocks implementation to use multiplexed token streams and written separate text blocks grammar. Everything seems to work except for a maddening issue: the second stream returns it's first token BEFORE the token where I'm pushing the input to the other stream. So basically, the "TEXT_BLOCK_LITERAL_END" token is being returned before the "TEXT_BLOCK_LITERAL_BEGIN" token! If I can't get this figured out today, I'm going to push the branch to my fork and post a link to see if any mentors can help. I have a few more ideas about how to proceed, I will try all of them and share my results.

Are any mentors able to share this wercker failure?   07:22
https://app.wercker.com/checkstyle/checkstyle/runs/build/5eeb02d07b63df001a1e4895

Erik Silkensen   09:12

[WARNING] Rule 0: org.apache.maven.plugins.enforcer.RequireJavaVersion failed with message:
Detected JDK Version: 1.8.0-232 is not in the allowed range [11,).

It looks like this isn't specific to your PR - https://github.com/checkstyle/checkstyle/pull/8324 is also failing the same way

Nick Mancuso   19:35

NM

EOD posted in google groups thread: https://groups.google.com/g/checkstyle-devel/c/-_se7-Pjrag/m/3jfJ8ILeCAAJ

19 June 2020

Nick Mancuso   05:24

NM

Plan for today: continue hacking text blocks grammar. I am getting pretty close to having a working implementation, I will push my changes and share my branch along with a gist of what I'm struggling with, later today. The only real challenge left with this is to figure out why the TEXT_BLOCK_LITERAL_BEGIN token is out of order, and how to best keep track of newlines, since "\n" is able to be used in text blocks.

Nick Mancuso   10:29

NM

draft PR for text blocks is up:
https://github.com/checkstyle/checkstyle/pull/8327

Nick Mancuso   11:07

NM

Mentors, please take a look at the AST for text blocks and let me know if this what we want: https://github.com/checkstyle/checkstyle/pull/8327/files#diff-005f4a4ab29fcdadc426ab80fbcda6c3

I really liked @pbludov 's idea to make it more like the javadoc tree, so I   11:08
wanted to implement it, sans the "white space" nodes.

I have submitted a PR for text blocks today, and I am continuing to work on the grammar for it, then I will move onto coverage. Unless any mentors object, I will not be working on Monday, since I have the opportunity to take my family to the beach for the day. I have put in quite a few extra hours this week, and plan to next week as well to make up for it... I would like to get a few PRs merged next week if we can, please let me know what else I can do to make that happen.

Nick Mancuso   19:13

Check regression reports for text block grammar have been posted here: https://github.com/checkstyle/checkstyle/pull/8327#issuecomment-646775858. Everyone have a nice weekend!

23 June 2020

Nick Mancuso   04:35

Plan for today: Work on text blocks grammar and unit tests. I should be able to update the text block draft PR and mark it ready for review by tomorrow or Thursday. I need review on https://github.com/checkstyle/checkstyle/pull/8313 and https://github.com/checkstyle/checkstyle/pull/8294 from @romanivanovjr. I also need review on https://github.com/checkstyle/checkstyle/pull/8293 , https://github.com/checkstyle/contribution/pull/484 and https://github.com/checkstyle/checkstyle/pull/8320 from Richard .

Nick Mancuso   16:02

EOD: made progress on text blocks grammar; I should be able to finish it up tomorrow and begin work on unit tests. I plan to have PR updated with new grammar, unit tests, and check regression reports by Thursday, and mark it ready for review by the weekend.

Erik Silkensen   17:57

Awesome, thanks for all your hard work Nick, and I hope you were able to enjoy some time with your family yesterday.

Nick Mancuso   18:11

@esilkensen thanks, we did! Thanks for the input/ explanation on the text blocks issue. I sort of lost sight that logic for checks should come first, before the appearance of the printed AST.

24 June 2020

Nick Mancuso   04:44

Plan for today: continue refining text blocks grammar and writing tests. I will likely be able to run all regression reports so that the PR can be updated tomorrow by EOD.

Nick Mancuso

jdk14

Nick Mancuso   05:01

NM

Plan for today: finish up text blocks PR and mark it ready for review.

Roman Ivanov   07:55

RI

https://github.com/checkstyle/checkstyle/pull/8294#pullrequestreview-437574268 - All, please review.

Nick Mancuso   11:17

NM

@pbludov is the code that is on the screen here:https://youtu.be/qurG_J81_Cs?t=1065 valid java code right now? I have watched the video, but the auto-translations are not so good.

Pavel Bludov   13:40

PB

In reply to this message

No, in this video, Tagir tells us some details about pattern matching, what difficulties there are now and what is possible to develop in the future. Note that this is a presentation from 2018. Some of the plans have already materialized. Something continues to develop in https://openjdk.java.net/projects/amber/

Nick Mancuso   14:14

NM

Thank you for explaining. I was going to add it as a test case if it was, but I tried it and it wouldn't compile. On another note, is it ok to start a review on an item in my own PR? I have a question about the best way to reuse existing code in AstRegressionTest.java.

Nick Mancuso   15:43

NM

Can anyone explain why I am getting this: https://teamcity.jetbrains.com/viewLog.html?buildId=2990971&tab=Inspection&buildTypeId=Checkstyle_IdeaInspectionsPullRequest ? I copied the existing "AssertGeneratedJavaLexer" class and tests (testImpossibleExceptions and testImpossibleValid), which do not cause these warnings. I understand what the warning means, I do not understand why the existing class and tests don't cause them.

Nick Mancuso   18:16

NM

EOD here: https://groups.google.com/g/checkstyle-devel/c/-_se7-Pjrag/m/zNQlAeKWAgAJ

Check regression reports posted to text blocks PR; I had to upload them 18:18 to my s3 bucket since github didn't like the size of this report, so it took longer than usual. AST regression report will be posted as soon as it finishes tonight or first thing tomorrow morning.

I will catch up on all the latest PR's today and this weekend, but re: starting a review of your own PR: absolutely, I think in general it's a good practice to put up a PR and do a self review of it first thing even, check things over, add comments to help guide reviewers, provide context, call out things you're not sure of, etc.

(If I am understanding your question right - sorry, it is the end of a long week 😴 you might already be doing everything I just said!)

Nick Mancuso   04:16

Plan for today: rework TokenTypes.java in all PRs to new specification, determine if openjdk14 has any usage of enhanced instanceof, create input update for instanceof relating to checks that use VARIABLE_DEF.

@romanivanovjr for the enhanced instanceof input updates (for checks), 04:19 should I simply add to the existing input files, or create new inputs and tests? Should this be in a separate commit? Also:
https://github.com/checkstyle/checkstyle/pull/8294#discussion_r445638524

Nick Mancuso   10:16

While searching through openjdk14 for folders that we can uncomment in `projects-to-test-on.properties` , I found some more interesting enhanced instanceof usage. Is it ok to add them to the test inputs for instanceof PR even though two reviewers have approved already?

Nick Mancuso   11:08

Ok, so I tried to simply add test cases to the existing files, but (of course) now they won't compile. I am unsure how to proceed. should I create a new test suite, using all naming checks and others related to VARIABLE_DEF? For example, make one new java class with all naming related checks, in test/java.../naming/, then place the input file in a noncompilable directory, like the other java14 inputs?

Nick Mancuso   13:53

EOD report: all PR's updated with new TokenTypes documentation, researched instanceof usage further in openjdk, found some more input examples for testing, addressed all items in PR's, and begin creating input and setting up configs for instanceof check input updates.

Also, I began review on   13:54
https://github.com/checkstyle/checkstyle/pull/8327

27 June 2020

Erik Silkensen   02:48

In reply to this message

re: making changes to PR's after approval - especially for something like adding additional test cases, that's likely a good change to make - and you can

RI   **Roman Ivanov**   11:02

> Is it ok to add them to the test inputs for instanceof PR even though two reviewers have approved already?

It is always good to add more test cases and test inputs, the more the better :)

> should I create a new test suite, using all naming checks and others   11:07
related to VARIABLE_DEF?

New input files in no compilable folders, location of test methods are in existing test classes.

Please provide list of Checks that are affected by new variable_def   11:12
location

We need to think is it possible to update Checks later on?, will it be false   11:14
positive or negative ?

NM   **Nick Mancuso**   11:20

I will provide a list on Monday, but I have looked through almost all of them, and about a third of them crash checkstyle, so we will have to update them now.

NM   **Nick Mancuso**   11:35

Most of the problem is that since the modifiers AST is null, many of the checks throw a null pointer exception

Since this pattern definition is always final, could we make a "final"   11:37
modifiers AST by default?

RI   **Roman Ivanov**   11:38

I do not understand about final, please rephrase and extend

NM   **Nick Mancuso**   11:39

> **Photo**
> Not included, change data exporting settings to download.
> 506x78, 8.3 KB

In this rule, could we construct a modifiers AST with a "final" token by   11:40
default, since a pattern definition is always final?

ES   **Erik Silkensen**   11:42

Maybe a dumb question, don't have the code in front of me: could modifiers be an empty array instead of null? Or is that not the right type, has to be a tree?

NM   **Nick Mancuso**   11:43

It is an AST.

Erik Silkensen

Nick Mancuso   11:46

Some of the checks look at the modifiers to see if the variable is final, that's why I thought maybe we should construct the "final" modifier AST by default.

Erik Silkensen   11:48

Yeah, it makes sense, though I wonder about fabricating syntax like that, if it's possible to update the checks instead even if harder, maybe that's better

Nick Mancuso   11:53

Yeah, I feel like it is cheating a bit.

Plus, that would appear in the printed tree, and it doesn't really exist.   11:54

Roman Ivanov   11:57

We can not put in AST that doesn't exist in code, if no final modifier in code, there should not be it in AST

Nick Mancuso   11:59

Ok, I agree. I will provide a list of affected checks on Monday, and begin creating tests and inputs. Should this work be a separate commit?

Roman Ivanov   12:12

In general we do not allow to changes that break any functionality. But situation is not general. Without new instanceof grammar support we just throw parser exception, so it is blocker to use checkstyle at all. Ones we merge grammar change, exception is. moved from parser to Checks. Original problem is more clear signal that it is not supported, exceptions in Checks looks like bugs by users.

Let's see your report first on what and how it failing. There should be   12:14
exception when such new code is used.

There will be relate tomorrow, so we can merge PR after release   12:17

But before grammar change, we should have a plan of what Checks are 12:20
affected, and create issues on each of them. To fix before next release (end on July)

Nick Mancuso   12:37

Sounds good. Does the checkstyle_checks.xml config have ALL checks in it? I was thinking I could use that to test all checks if so.

Richard Veach   12:42

yes, it does. we also have the regression checks in contribution. Those shouldn't have any checks suppressed while ours could.

28 June 2020

Nick Mancuso

jdk14

Nick Mancuso   05:34

Plan for today: make sure all CI is green in all PRs, address reviews in text blocks PR, raise issues for instanceof check exceptions, and post in email thread.

Roman Ivanov   07:55

@Nmancuso , please explain in your comment with table of affected Check at https://github.com/checkstyle/checkstyle/pull/8294 how you get this list. Is is just regression testing? There will be more problem in other checks but not as exception and more like validation malfunction, We need to have a list of Checks that use VARIABle_DEF , there should not alot, and we need to review them to see if they are affected.

please put priority on instanceof more than work on other jdk14 features 07:55

there will be a lot of PRs to make it work, so lets start sooner, review is not quick process.   07:56

Nick Mancuso   07:58

Something that I did not think about/understand when I first ran the check regression report for instanceof, was that the test input file was not merged; so that code was not tested, and these exceptions were not found. I ran check regression after that input was merged, and found these exceptions, which I have confirmed by running checkstyle_checks.xml on the input file, then each check separately.

I raised issues for each check   07:59

Roman Ivanov   08:00

please focus on completion of grammar update on instanceof, it is block for all other udpates, all other updates could be done in parallel

if CI start to fail on our Inputs of instance of - please exclude them till all updates to Checks are merged.   08:01

Erik Silkensen   08:03

In reply to this message

Catching up on the threads - are there planned changes for instanceof grammar in addition to/before updating checks to take it into account as a new kind of variable definition?

Nick Mancuso   08:03

All CI for instanceof grammar is green, but wercker is still running. What else do I need to do so that we can get it merged?

Roman Ivanov   08:09

wercker is restarted, it was sonar, violation, I masked it as waontfix

jdk14

Nick Mancuso   09:09

Should tests for possibly affected checks (that don't cause exception) be added to instanceof PR?

Roman Ivanov   11:16

No, update for grammar should has only grammar update and related inputs for parser tests

All other be addressed in later on PRs, module by module.   11:16

Nick Mancuso   11:17

Should I make issues for them, too?

Roman Ivanov   11:17

Yes, we need merge grammar update and have clear plan on what is affected and what needs to be fixed

Nick Mancuso   11:19

Ok, I was working on fixing and making inputs for HiddenFieldCheck (for instanceof), I will stop and make all issues (issues with exceptions are made already) is there anything else I should be working on today?

Also, should I make one "master" issue for all instanceof check updates,   11:21
like we had with the pitest issue, and link to individual issues?

Nick Mancuso   15:20

EOD report: I have created issues for all checks that utilize the VARIABLE_DEF token, identified those that are causing exceptions, added inputs to instanceof PR, and made CI happy for instanceof PR. I am posting in email thread now. Tomorrow, I plan to provide more information/context in the issues I've posted today. Please let me know what exactly I can provide to make review of these issues as quick and easy as possible. As it stands, I plan to compare the grammar outlined in JLS/JEP to determine if certain checks should be impacted by the addition of instanceof grammar, then target specific usages of instanceof to see what should be caught by checks. I have provided the normal bug report information for those checks that throw exceptions already.

Nick Mancuso   17:40

Email posted: https://groups.google.com/g/checkstyle-devel/c/-_se7-Pjrag/m/zNQlAeKWAgAJ

30 June 2020

Nick Mancuso   05:07

(hopefully) we can get #820 merged sooner, I think it is good to have all issues contained in one place. After this is done, I will begin to post additional information and context to each issue.

Nick Mancuso 05:39

Main tracker issue for validation of checks related to #8294:
https://github.com/checkstyle/checkstyle/issues/8391

I can begin work on the checks that throw a NPE as soon as their issues 05:40
are approved

Roman Ivanov 06:29

https://github.com/checkstyle/checkstyle/pull/8294 - is approved and moved to last review to Richard. @Nmancuso , please recheck generated reports to make sure all still works fine even we changed AST a bit. It would be awesome if you send PR to print generation data-time in diff reports.

Nick Mancuso 06:31

I did not change the implementation of the grammar, it has been the same since I submitted the PR. I just ran reports two days ago, and would be happy to share them or run them again if you would like.

Roman Ivanov 06:32

Lets rerun them and be safe.

Nick Mancuso 06:32

Of course, I will start them now, and head over to contribution.

Please note, that checkstyle "project" for check regression will throw NPE 06:33

Roman Ivanov 06:34

add extra exclude for several files

that has patterned instanceof 06:34

Nick Mancuso 06:34

Will that work for check regression? I thought excludes were ignored for regression.

Roman Ivanov 06:34

oh, yes, you ar right

ok, wec will be aware of NPEs in chekstyle 06:35

Nick Mancuso 06:36

Roman, can you take quick look at
https://github.com/checkstyle/checkstyle/issues/8364 and let me know if this is all the information that we need? If so, this is what I will provide for all related issues

and prepare for massive PRs review. @Nmancuso , please focus on Exceptions first. There might be no updates in Check required, if so, just prove this in issue and let mentors close it without changes. My review is not required merge all that PRs, @esilkensen and @pbludov review (2 reviews) will be enough. I need to go back to other projects, as they are not in good state.

Erik Silkensen   08:02

In reply to this message

Thanks Roman, that all sounds good

Nick Mancuso   10:44

I think I've finished up everything in contribution; what is the best way for me to proceed with fixing checks that throw NPE, since instanceof PR is not merged? Make a new branch from instanceof PR branch, and base work off of that?

Nick Mancuso   11:21

EOD report: since I got an early start today, I'm wrapping up now. I have opened two PR's in contribution today: https://github.com/checkstyle/contribution/pull/489 and https://github.com/checkstyle/contribution/pull/488, made a parent issue for all instanceof check validations https://github.com/checkstyle/checkstyle/issues/8391, and updated several of the child issues, namely https://github.com/checkstyle/checkstyle/issues/8364 , https://github.com/checkstyle/checkstyle/issues/8366 , and https://github.com/checkstyle/checkstyle/issues/8367. New reports for instanceof PR are still running, I will post them in PR as soon as they are ready.

@romanivanovjr I'm not sure if this was intentional or you just ran out of   11:23
time, but I still need approval on
https://github.com/checkstyle/checkstyle/issues/8362 ,
https://github.com/checkstyle/checkstyle/issues/8361 ,
https://github.com/checkstyle/checkstyle/issues/8360 , and
https://github.com/checkstyle/checkstyle/issues/8359. Thank you.

Roman Ivanov   13:06

Approved.

>Make a new branch from instanceof PR branch, and base work off of   13:08
that?

Yes, in PR description just mention that first commit is from another PR

Nick Mancuso   15:25

Re-running reports, they failed with: `Exception in thread "main" java.lang.Error: Error was thrown while processing /home/nick/development/contribution/checkstyle-tester/src/main/java/checkstyle/src/test/resources/com/puppycra`

`.processExpression(ParenPadCheck.java:325)`

Unusual, I've never had this issue before. I will update instanceof PR with new reports as soon as they are done.

1 July 2020

Nick Mancuso   04:51

Plan for today: begin working on all checks affected by instanceof that throw NPE. Since I had to restart reports for instanceof PR, they are still running, but should be done by EOD today.

Nick Mancuso   08:40

Richard https://github.com/checkstyle/checkstyle/issues/7290#issuecomment-638979390

Richard Veach   08:40

Thanks.

I am trying to review the PRs now.   08:41

Nick Mancuso   08:45

I have new check regression reports running, I've been having some random weird problems.

The grammar has not changed since the ones that I have posted,   08:45
though.

Richard Veach   08:47

It depends on what you have been seeing. It is possible some checks could not be fully deterministic and have an ounce of randomness to them. I would keep track of which checks and issues you have to see if one comes to lite. Let us know if we can help further.

Nick Mancuso   08:47

https://travis-ci.org/github/checkstyle/checkstyle/builds/703921603 could these have anything to do with openjdk being down?

Twice it was failure to delete a directory, once was what I noted above.   08:48

Richard Veach   08:49

https://travis-ci.org/github/checkstyle/checkstyle/jobs/703921612 To me, they seem unrelated as it just lost output from the computer. We don't know what happened that it lost that connection.

I see they are all JDK14 builds. That is strange. I tried restarting them.   08:51

Nick Mancuso   08:53

Thank you. I just re-confirmed that all jdk14 files I added DO compile

This is down, too: https://hg.openjdk.java.net/jdk/jdk   08:58

RV

if you already downloaded HG OpenJDK, you can change it to a folder for regression to prevent regression from trying to poll the server

NM

Nick Mancuso   09:00

I nuke the repo directory after every run to avoid issues. Can I just use the read-only github mirror?

RV

Richard Veach   09:00

thats fine too.

What issues do you have that you have to nuke the directory?   09:01

NM

Nick Mancuso   09:02

None now, but I was going back and forth between my fork of openjdk (I deleted the deeply nested lambda files) and the regular one, and was too lazy to rename the project :)

jdk14, that is   09:02

So I just added a "clean" type of operation to my script, figured it would help eliminate any random stuff.   09:03

I have 1000Mbps connection, so it really doesn't slow things down   09:03

RV

Richard Veach   09:04

of course, just will degrade your HD. :)

NM

Nick Mancuso   09:04

Shoot, didn't think about that, good point.

RV

Richard Veach   09:06

If memory serves me, I don't think the groovy scripts poll the repo after the first download. It copies the files over during regression, so the main repo should remain unchange after any regression run.

Those JDK14 builds failed again, I would assume it is related to the JDK 09:12
servers being down.

NM

Nick Mancuso   09:12

No versions lower than 11u are on github...

This is where it keeps failing: `Removing non referenced xref files` 09:21
`in report ...`
`[delete] Deleting directory`
`/home/nick/development/contribution/checkstyle-`
`tester/src/main/java/openjdk9`
`[move] Moving 17362 files to`
`/home/nick/development/contribution/checkstyle-`
`tester/reports/openjdk9`

What generates this part of the output?

The next line should be: `openjdk9 is synchronized` but it just hangs   09:22

Richard Veach   09:23

https://github.com/checkstyle/contribution/blob/master/checkstyle-tester/launch.groovy#L261

its probably hanging in this method,   09:23
https://github.com/checkstyle/contribution/blob/master/checkstyle-tester/launch.groovy#L125

that is the clone utility where it brings in a brand new repo.   09:24

Nick Mancuso   09:25

Ok, so that at least clears up why everything is failing right now. I thought I broke everything!

I am going to comment out the mercurial java projects, and continue with   09:27
the remaining reports. I had about half done before all this started happening.

for instanceof, that is.   09:28

Nick Mancuso   17:55

EOD report: just submitted latest check regression reports to #8294 (instanceof), updated PR. Submitted a PR for https://github.com/checkstyle/checkstyle/pull/8392, submitted a patch in contribution, and added some information to several of the instanceof related issues. I had a personal matter to attend to, and openjdk site was down, so today was not as productive as I had hoped. I'll be back on it tomorrow. Could a mentor please restart the CI for https://github.com/checkstyle/checkstyle/pull/8392 ? I think openjdk is stable now.

I'm running the missing projects (openjdk) from earlier right now.   18:01

Can a mentor take a look at this: 18:10
https://github.com/checkstyle/checkstyle/pull/8392/commits/20a60fdacaf5f50e7d016158d43ca8248a11c409 and let me know if this is ok? I will be making my other branches from this commit for the remaining instanceof check issues, and will be using this utility in many of them.

2 July 2020

Erik Silkensen   02:26

In reply to this message

Restarted JDK 14 build, also edited your PR description to reference the Issue # for the stacked commit so the grep check passes, all green now

In reply to this message   02:27

Sounds like a good plan to me - I left a question in the PR

And post in email thread   05:25

Nick Mancuso   09:31

[Richard https://checkstyle-reports-java14.s3.amazonaws.com/reports/instanceof_report/index.html](https://checkstyle-reports-java14.s3.amazonaws.com/reports/instanceof_report/index.html)

If this is too crowded, or you need anything else, let me know   09:31

Richard Veach   11:35

I don't think it's that crowded. I notice some variables are missing like try resource, lambda parameters, ...

Nick Mancuso   12:08

I'm on it

Nick Mancuso   13:02

@esilkensen hold up on reviewing any of those instanceof PRs, we might be making some changes I think.

Richard report is updated   13:14

Richard Veach   13:18

@ni

@Nmancuso What about other variables like method parameters? We   13:19
are looking of a complete list of everything considered a variable.

Class fields is another one.   13:20

Sorry see the method parameter now.   13:21

Nick Mancuso   13:21

Ok, I must've misunderstood what we needed. I thought we just needed to look at variables/parameters that are declared in a construct of some sort, not in a block.

Declared, and then used in the block of the construct.   13:22

Richard Veach   13:22

@romanivanovjr Wanted a list, I didn't see what he wrote, so maybe he can confirm the exact scope.

We were talking about pattern variables, and it delved from there as we 13:23
discuss how we handle other variables in the grammar

Nick Mancuso   13:23

Roman's message: "we need you review AST for all cases where variable is declared in non block `{ }` but in expression or some special contruction (for, try, catch, ...)"

RV

ok, I didn't realize he wanted statements/expressions. So that rules out fields.

If thats everything, then thats what we want. We just wanted to be sure   13:25
we looked at everything.

Nick Mancuso   13:26

NM

Thanks for the lambda and try with resources suggestion, I was on the fence
about including lambdas, but I guess more information is always better.

Richard Veach   13:28

RV

try with resource was in our talks, so yea, I would assume everything and
anything that can be written inside a method.

Nick Mancuso   13:33

NM

On one hand, when comparing all of the others, the "PATTERN_DEF" node
does seem extraneous/ out of place. If anything besides just making it a
"VARIABLE_DEF" only, I think a "PATTERN_VARIABLE_DEF" replacing the
"PATTERN_DEF" and "VARIABLE_DEF" nodes would be good.

From JLS: A variable declared in a pattern is known as a pattern   13:34
variable...

Just my two cents.   13:34

Since records is green and approved, and we seem to be in a holding   13:45
pattern for instanceof, I am going to continue work on text blocks unless any
mentors object.

Nick Mancuso   15:51

NM

EOD report: https://groups.google.com/g/checkstyle-devel/c/-_se7-
Pjrag/m/JgiCredqAQAJ

Nick Mancuso   17:49

NM

I have converted all PRs except initial record grammar and enhanced
instanceof to draft status, since all is subject to change, and not worth
reviewing at this point. Tomorrow I will keep adding more info to instanceof
issues that don't rely on grammar/ implementation (non-NPE checks) and keep
working on text blocks.

3 July 2020

Nick Mancuso   05:16

NM

Plan for today: try to figure out where I need to add a suppression to make
TeamCity happy for text blocks PR (I got it working locally), dig a little deeper
into out of order first token for text blocks, and add some more info to a few of
the instanceof check issues (those that aren't related to implementation).

Nick Mancuso   05:36

NM

jdk14

Nick Mancuso 11:30

@esilkensen I just wanted to let you know that it seems like we are in a holding pattern for instanceof, that's why I didn't update the PR yet to fix Travis. I didn't want to waste CI time just for that, when I might have to rewrite much of the PR anyway. Should I mark it draft for now, too? I don't want you to waste time reviewing things that might change anyway.

Erik Silkensen 11:36

In reply to this message

All right, thanks - sorry, what is the holding pattern / are you blocked on instanceof? I see Roman's comment from yesterday about adding additional test inputs.

Nick Mancuso 11:41

No, basically, we may have to completely rehash the grammar, so I'm just waiting to hear what the final decision is. We may not be using the VARIABLE_DEF token anymore.

Erik Silkensen 11:47

Sounds good, so just checking I am up to date / not missing a thread: are Roman and Richard discussing this offline (not in a PR or this group) and will decide what they want to do?

Nick Mancuso 11:52

That's correct, it was in PR too, but I think that conversation got marked resolved.

If you have time this weekend, if you could help sift through any of these 11:55
that would be awesome: https://github.com/checkstyle/checkstyle/issues/8391.
Not implementation details, since that's up in the air, but if they actually apply. I just want a second/ third opinion for any before I close them.

That's really the only type of "reviewing" we need right now. Records is 11:57
green and lots of approvals, and text blocks is green too, but I'm not happy with implementation, so it is still marked draft.

Erik Silkensen 12:10

I wasn't able to find discussion in the PR besides Richard saying "only thing I really question is new VARIABLE_DEF being created from PATTERN_DEF" and then Roman asking about additional test input "after discussion with Richard."

Based on that, my understanding would have been: everyone was basically in agreement about the grammar, go ahead and add more test input, and let's move forward with addressing the impacted checks.

said, I will check them out from my computer.)

Anyway, I don't mean to throw a wrench into anything here - if you are good, and Roman and Richard are good, then it's all good on my end too of course - and I will be happy to help out however I can with whatever path we go down...

Nick Mancuso   12:21

Yes, there was some private discussion, I think implementation is subject to change right now. Regarding the checks, I should clarify; which checks should and shouldn't apply. I thought that maybe, just based on syntax, we can determine whether certain checks will/ will not apply, right? For instance, we were able to rule out https://github.com/checkstyle/checkstyle/issues/8367 regardless of implementation.

I just got my review from first evaluation, thank you for all the positive   12:27
comments and encouragement.

Erik Silkensen   12:29

In reply to this message

Ok, yeah just wanted to make sure you weren't waiting on something unnecessarily - it sounds like you are set for now! I will check out the list

Roman Ivanov   12:30

https://github.com/checkstyle/checkstyle/pull/8294 please do add inputs, it is not blocking, inputs do not depends on AST structure

Nick Mancuso   12:31

Ok, I will do that now.

Roman Ivanov   12:33

Please update inputs all affected Checks and even write expected behavior, no matter how AST will be Check behavior is be the same

You can start the same activity for other grammar changes :), you are   12:35
almost never blocked

We found all affected Checks, no matter what we will deside on how AST 12:49
should looks like, Checks behavior must not change.

We can prepare inputs, noncompilable inputs, we create even UT   12:52
methods that declare expected behavior. While grammar is not ready , we
keep such methods ignored by annotation. We merge all that. During grammar
update we can simply remove ignore annotation and that will be prove that it
works

Nick Mancuso   12:53

I agree, however, to find checks affected due to existing token usage, we will need to decide based upon implementation. For instance, I would not want to close an issue on a check that could cause a false positive/ negative based on

right?

Roman Ivanov   12:55

By reuse of tokens, we found all Checks that are affected, even we use different tokens, all that Checks need to be updated to be aware that variable can be declared even here. There will be no exception but we should prevent false positives.

While you're blocked, you create inputs and expected behavior declaration. Beauty of our tests that they do not depends on implementation at all.   12:57

Nick Mancuso   12:58

I think that this has been a good exercise, maybe it is better to make a new token to avoid unintended side effects.

Roman Ivanov   13:00

New tokens is good to do less work :) , but ones users start to use new syntax, they will get numerous false positives

Nick Mancuso   13:01

"The developer team of checkstyle is really a lazy bunch of people" :)

Roman Ivanov   13:02

Even we start to use new tokens, we can delay fixed in Checks, yes. But at least we know all affected :), much better situation than waiting for users bug reports

Nick Mancuso   13:11

What annotation should I use to ignore unit tests?

Roman Ivanov   13:12

`@Ignore`

There might be complains from checkstyle on usage of it, I am ok to suppress violation with comment "until .... issue... "   13:14

@esilkensen @pbludov , do you understand what I am proposing ? You will not need me to merge such stuff, it will speed up process.   13:17

Erik Silkensen   13:32

In reply to this message

Yes, just reading back to check I am understanding: we want PR's that add test cases for affected checks, testing those checks on new instanceof syntax, and they will be @Ignore'd for now - these are the PR's to merge without needing your approval, right?

Roman Ivanov

5 July 2020

Nick Mancuso   05:47

instanceof PR is ready, all reports posted and CI is green:

https://github.com/checkstyle/checkstyle/pull/8294

Nick Mancuso   06:05

Wait, here was a problem with the check regression reports, I'm going to run
them again.

AST regression is still good, master branch was not good for check   06:18
regression

Roman Ivanov   07:39

@Nmancuso , please explin why you selected this structure

```
`--LITERAL_INSTANCEOF (instanceof)
    |--IDENT (obj)
    `--PATTERN_DEF
        `--PATTERN_VARIABLE_DEF
```

what if we do:

```
`--LITERAL_INSTANCEOF (instanceof)
    |--IDENT (obj)
    `--PATTERN_VARIABLE_DEF
```

Nick Mancuso   07:46

Your comment in the "instanceof AST" thread: " so question is: How to
structure AST?
1) "PATTERN_DEF / VARIABLE_DEF"
2) "PATTERN_DEF / PATTERN_VAR_DEF"" led me to believe that in either
case,"PATTERN_DEF" was to remain the parent node of either
"VARIABLE_DEF" or "PATTERN_VAR/VARIABLE_DEF"

Nick Mancuso   17:49

Closed old pr, new one is here:

https://github.com/checkstyle/checkstyle/pull/8401

6 July 2020

Nick Mancuso   05:05

Plan for today: get check UTs squared away for instanceof, get reports posted
to instanceof, post in email thread. @romanivanovjr can you create a new
channel for full records support AST?

Nick Mancuso   06:43

Mentors, I have raised new issues for the effected checks that we identified in
the old instanceof PR that caused NPE (they do not cause NPE anymore), and
closed the old ones. I need approval on these last eight issues in main tracker
issue: https://github.com/checkstyle/checkstyle/issues/8391

base them off of instanceof PR again?

Nick Mancuso   08:15

I have to add a few suppressions across multiple config files, then go back and make all new PRs to un-suppress everything. It would save time not to do this. Couldn't we just add "blocked" tag to PR for check UT?

Roman Ivanov   08:20

> can you create a new channel for full records support AST?

lets keep discussion in PR for now.

Roman Ivanov   08:35

https://github.com/checkstyle/checkstyle/issues/8409 - should we make special Check for join FinalLocalVariableName ?

Naming Checks are not affected when new token is introduced and we   08:39
almost agree to make new Check on this. There are more problems in Checks that use VARIABLE_DEF in logic to find something.

Erik Silkensen   08:40

In reply to this message

Agreed, this sounds good to me

Roman Ivanov   08:40

> do I still need to @Ignore UTs, or can I just base them off of instanceof PR again?

you can base of PR with grammar fix, as we close to merge.

Nick Mancuso   08:42

Awesome. My reports are almost all uploaded to S3, but openjdk is/was down again, so the last few reports are missing openjdk projects. I will post what I have as soon as they are uploaded.

Nick Mancuso   09:37

Reports are posted:
https://github.com/checkstyle/checkstyle/pull/8401#issuecomment-654011201

Nick Mancuso   16:20

Kinda EOD report: remaining check regression reports (java projects for instanceof PR) are almost complete, I will post tonight. I have addressed each issue raised in reference to check support for pattern variables. I would like at at least one mentor approval before closing any check related issues, and I have mentioned @esilkensen and @pbludov in many of them for a second opinion. I submitted a PR for AbbreviationAsWordInNameCheck: https://github.com/checkstyle/checkstyle/pull/8412 and CI is green.

My plan for the week is to resolve all checks related to instanceof, and submit a draft PR for full records support so that we have lots of time to discuss AST structure.   16:23

Nick Mancuso   19:57

Email in thread: https://groups.google.com/g/checkstyle-devel/c/-_se7-Pjrag/m/32Uc_KyyAgAJ

All missing java project reports posted to instanceof PR: https://github.com/checkstyle/checkstyle/pull/8401   19:57

7 July 2020

Nick Mancuso   04:53

Plan for today: work on checks for instanceof. I need to run out for a couple of hours mid-day, in case I don't respond right away. @romanivanovjr and anyone else, please add thoughts/ and discussion tag to this: https://github.com/checkstyle/checkstyle/issues/8414

Roman Ivanov   05:59

Issue is closed with comment.

Roman Ivanov   06:42

I reopened several issues on Checks that were affected by old instanceof implementation, @Nmancuso , please review them and provide prove that no false-positives and false-negatives.

Erik Silkensen   07:51

In reply to this message

Thanks Roman, I had made a first pass on those last night with my best guess and appreciate the second set of eyes.

I wasn't sure I understood your comment on reopening the unused import check - it seemed what Nick posted was the correct behavior, but certainly that is a check we want to make sure is working correctly.

Nick Mancuso   08:13

Wouldn't we only really know if the import is "used" at runtime?

Should we check for usage in the IF block?   08:14

Erik Silkensen   08:19

I think for the purposes of that check "used" means an import is referenced somewhere in the file, regardless of whether it may possibly be executed or not. So we would want to confirm that if an import is only used in a pattern variable definition, the check detects that as a used import. Similarly if an import is not used we would want to make sure a pattern variable can't interfere and suppress the violation.

Nick Mancuso   08:21

```
 import java.awt.Component;
class FooBar {
private Object Component; // a bad practice in my opinion
...
}
```

@esilkensen from the check documentation ^^^   08:21

But this example is ok, and will not be flagged.   08:23

Well not ok, but not detectable by the check.   08:29

Erik Silkensen   08:31

Yes, that makes sense - that Component is just a variable name and not a reference to the imported class.

Looking at the check, I think the question is does it already do what we want for pattern variables, so picking up the type name ident and including that in its list of references

Nick Mancuso   08:34

From my example, that shows that it works as intended, and yes, the check itself is looking at IDENTs. To cause a violation, I should just import something else and not use it?

                                                                    08:36

Photo
Not included, change data exporting settings to download.
478x106, 10.0 KB

It's definitely hitting that IDENT.   08:37

Erik Silkensen   08:44

Oh sorry, I didn't read close enough - I would have assumed the check was smart enough to detect that the Component variable above was NOT an import reference and so there should be a violation.

In which case, never mind my comments about needing to check   08:46
whether an ident is part of a type name or not... and as far as I can tell this issue isn't impacted by your PR and can be closed, so I defer to Roman and others to continue...

Roman Ivanov   08:57

For reopened issues, just share prove by CLI

Nick Mancuso

jdk14

Roman Ivanov   13:33

Use type only in pattern, Check should not complain on unused import

Erik Silkensen   13:42

In reply to this message

I think that is what Nick was showing here
https://github.com/checkstyle/checkstyle/issues/8379#issuecomment-654548538

Nick Mancuso   16:40

EOD report: I have just posted a new PR for PatternVariableNameCheck :
https://github.com/checkstyle/checkstyle/pull/8417. Today, I have addressed all
check issues raised by instanceof PR, and determined whether or not action
must be taken.
Over the next few days, I plan to resolve all issues, and update all checks, to
be completed by Friday. For the existing instanceof check PRs, I will drop the
(now conflicting) previous instanceof PR commit, and rebase (I'll do this
tomorrow so we can merge sooner). I expect to have Full records support draft
PR up by Friday afternoon.
Please note that I am taking my planned vacation next week, but I will still plan
on having some limited interaction/ coding later in the week.

Mentors, what is your feeling about my progress so far? Do you think I   16:42
will be able to successfully complete my proposal on time?

Roman Ivanov   18:04

@Nmancuso , please try to focus on other grammar changes before vacation
to provide us bunch of information to think of (AST strucuture and .....) while
you are out

Nick Mancuso   18:04

Ok, text blocks is up, the only one left after full records is switch, right?

Roman Ivanov   18:04

It is better to worok on Checks updates when you are blocked by us to make
decision on how to make it

Nick Mancuso   18:05

Ok, no problem. I thought we wanted to make sure we had full support for
instanceof before the next release, that is why I had other plan.

Roman Ivanov   18:06

nobody use instaceof .... in code so no user for Checks :)

Nick Mancuso   18:06

Then: change of plan. I will have draft PRs for all remaining grammar changes
by Friday. Checks will wait, except I have PatternVariableNameCheck almost

and not that much people use latest chekcstyle version

> what is your feeling about my progress so far?   18:09

we are ok. I still think we will be able to complete on time. First grammar update showed us all details that we need to keep in mind, so all others will have less pause and we know what to expect from update.

Nick Mancuso   18:13

NM

For records, would grepping for checks that use the CLASS_DEF token be a good approach to figuring out what is effected?

Roman Ivanov   18:22

RI

CLASS, ENUM, INTERFACE ... all type declarations related Checks might be affected. We need to grep all and review case by case. For example https://checkstyle.sourceforge.io/config_design.html#InnerTypeLast might result in false negatives, but it is minor priority. As High priority we need to resolve Checks that are fail with exceptions. As major priority resolve false-positives, last priority false-negatives.

Nick Mancuso   18:26

NM

For text blocks, all string-related checks? final variable check, what else?

Roman Ivanov   18:30

RI

I can not find quickly affected, we do not validate string content that much

Nick Mancuso   18:31

NM

ok, I will post a proposed list this week, so we have time to think on it.

Nick Mancuso   19:32

NM

Hey guys, I made a mistake, my vacation is the week after next (week of the 19th), as stated in my proposal. My apologies.

8 July 2020

Nick Mancuso   04:43

NM

Plan for today: work on full records grammar.

Nick Mancuso   12:01

NM

Let's discuss: https://github.com/checkstyle/checkstyle/pull/8422

Nick Mancuso   16:12

NM

EOD report: submitted PR for full records support: https://github.com/checkstyle/checkstyle/pull/8422, submitted PR for https://github.com/checkstyle/checkstyle/pull/8421, rebased and dropped instanceof commit for https://github.com/checkstyle/checkstyle/pull/8412 and https://github.com/checkstyle/checkstyle/pull/8413 , and have reports running

9 July 2020

Nick Mancuso   03:47

Plan for today: post in email thread, work on enhanced switch grammar, post reports to full records PR.

Nick Mancuso   05:01

All reports posted to full records PR:
https://github.com/checkstyle/checkstyle/pull/8422#issuecomment-655698342

Nick Mancuso   06:29

Mentors, can one of you relay this failure?
https://app.wercker.com/checkstyle/checkstyle/runs/build/5f06f1d5dac5630008af18a0

Roman Ivanov   07:11

This project is in good enough state for now, I need to focus on patchfilter project (it in very bad state), so I will be less active here. Mentors, please step in.

Wercker is restarted, some download problem   07:13

Nick Mancuso   16:26

EOD posted in email thread: https://groups.google.com/g/checkstyle-devel/c/-_se7-Pjrag/m/EumLcfbFBgAJ

Put about four hours into the switch expression grammar today, it ended   16:27
up being more complicated than I thought it would be. I hope to make some
headway tomorrow. All reports are posted to full records PR.

10 July 2020

Nick Mancuso   04:56

Plan for today: keep working on switch expression, add more info to records PR.

Roman Ivanov   07:08

Mentors, I remind that when we create new Input we put in it config of check as comment above Class and trailing comments in line with violation - https://github.com/checkstyle/checkstyle/pull/8412/commits/58619d5fa74d372405f182195b7cc7180f0a0c96#r452865287 . One day in future we will read config and violations from Input file. @Nmancuso , is it possible to reuse some Check to enforce it ?

Nick Mancuso   07:12

Add a test to check all input files for a block comment at the top of the file regarding config of check?

446435202 is possible to implement, and we will relax on this, build will check it

Nick Mancuso   07:25

Are there any examples of something similar? I'm not sure what the best way to do this is.

Roman Ivanov   07:32

I instructed Nick in private message on how to reuse RegexpMultiline, I hope it will work.

Nick Mancuso   12:33

@pbludov sorry about the permissions, S3 is weird. I am setting all to public again, it should be done in about five minutes or less

@pbludov let me know if it is still not working   12:43

Pavel Bludov   13:18

In reply to this message

Works now, thank you.

Actually no. Some of them are accessible, others are not. For example: 13:21
https://checkstyle-reports-java14.s3.amazonaws.com/reports/full-records-support_check_diff_reports_2020_07_08/diff_part1/checkstyle-with-excludes/xref/home/nick/development/contribution/checkstyle-tester/repositories/checkstyle-with-excludes/src/test/resources-noncompilable/com/puppycrawl/tools/checkstyle/grammar/java14/InputJava14Records.java.html#L14
Maybe we should wait longer.

Nick Mancuso   13:22

Ok, either way, I was able to get rid of the instanceof and initial record reports from github.io, so I just pushed them to github. I'll post the links for them right now.

Links to github.io are posted now too, but it seems to be taking along   13:33
time for them to populate.

Nick Mancuso   13:50

https://travis-ci.org/github/checkstyle/checkstyle/jobs/707008420 can someone restart CI for https://github.com/checkstyle/checkstyle/pull/8435

Nick Mancuso   14:33

https://codecov.io/gh/checkstyle/checkstyle/compare/aa41a0f7c4c1393ec02adc7cfd7f9194a1006759...e759a60f74f65c163feb25fe2155ad0bc6c742a7 is this new?

Nick Mancuso

the grammar is still stumbling with the typecast lambda assignment
statements.

Erik Silkensen   16:49

Sounds good — I know I have several things in my queue that I haven't got to
yet, I am planning to catch up this weekend.

11 July 2020

Nick Mancuso   11:45

Mentors, if you have any tine this weekend, please provide some feedback on
the text blocks PR:https://github.com/checkstyle/checkstyle/pull/8327

I think that everything is working great, other than the   11:46
TEXT_BLOCK_BEGIN token being out of order; I'm not sure if that is a deal
breaker or not, please let me know.

I am going to keep trying on switch expression grammar until Monday   11:50
afternoon EST, if I can't figure out what causing the problems with typecasting
lambdas, I will create a detailed report to provide so that I can get some
suggestions.

Roman Ivanov   12:09

Where can I read on reason to have endofline in text blocks ?

Nick Mancuso   12:22

https://groups.google.com/g/checkstyle-devel/c/-_se7-Pjrag/m/32Uc_KyyAgAJ
, https://groups.google.com/g/checkstyle-devel/c/-_se7-Pjrag/m/JgiCredqAQAJ
, https://stackoverflow.com/questions/62705997/antlr-2-token-stream-
multiplexing-why-are-my-tokens-out-of-order ,
https://groups.google.com/g/antlr-discussion/c/MdTqaDv0ZA4

                                                                      12:25

Photo
Not included, change data exporting settings to download.
538x238, 21.5 KB

Basically, everything works according to the guide here:
https://www.antlr2.org/doc/streams.html, except our "DOC_OPEN"
(TEXT_BLOCK_BEGIN) is coming after our "DOC_CLOSE"
(TEXT_BLOCK_END), not the beginning, like in this diagram:

I have traced though the execution of this for many hours, and can't   12:30
understand why this seemingly simple operation isn't working as intended.

I'm really out of ideas at this point, honestly, and what I've done works   12:32
great tracking the line numbers, the location of the delimiters, etc, except for
this one unexpected behavior.

The downfalls of the three other implementations I tried were the   12:36
difficulty/inconsistency in tracking line numbers and the locations (columns) of

can you point me to your antlr that is having problems?

Nick Mancuso 12:38

https://github.com/checkstyle/checkstyle/pull/8327/files#r446170918

Richard Veach 12:39

I see parsing gets content, then end and then begin again

why isn"t it begin, content, end? 12:40

Nick Mancuso 12:40

That's what I don't understand.

That's what order the tokens should be in, of course, but the 12:41
TEXT_BLOCK_BEGIN token is being returned after the second lexer is done

Richard Veach 12:42

what is the "!" for?

Nick Mancuso 12:42

to not attach the AST then, to attach it in the action

So that I can fix the order 12:42

Richard Veach 12:43

so your saying if you move line 1657 to 1655, the parsing fails?

Nick Mancuso 12:43

Thats correct

the tokens are out of order, and I cannot figure out why 12:44

This has been very frustrating, because everything else about this 12:44
implementation work perfectly...

Richard Veach 12:45

I would need to look into this more to help understand it

why is the text block grammar split into a new file? 12:45

also "\r\n" isn't evil dos. Its Windows line endings 12:46

Window users may take offense to that 12:46

<--- Windows user 12:46

Nick Mancuso 12:46

Because to parse only the content, and return a separate token, the regular
lexer can't handle that.

I meant no offense, I copied it verbatim from our regular grammar file 12:47

Nick Mancuso 12:47

I don't care what kind of line endings you use :)

I'm curious, if everything else works great, is this a dealbreaker? 12:49

the one out of order token, that is. 12:49

Richard Veach 12:50

can you point to me where the example output tree is?

nm found it 12:50

it was collapsed 12:50

https://github.com/checkstyle/checkstyle/pull/8327/files#diff-005f4a4ab29fcdadc426ab80fbcda6c3R106 12:51

I am not seeing anything out of order 12:51

Nick Mancuso 12:51

Right!

What I did in the grammar does produce the intended AST 12:51

Richard Veach 12:53

the ouput is the most important imo, if it is right, then a weird limitation of an old code is fine to me

antlr is using version 2 instead of 4 12:53

Nick Mancuso 12:54

That is great to hear. The way I did this with the second lexer, IMO, is the only way to handle all the escapes and everything that can be in a text block, and differentiate between a typed '\n' and a real control character

Richard Veach 12:55

we can make a note/issue to try to understand why the grammar coding looks weird, but correct output shouldn't stop you

Nick Mancuso 12:55

Yeah, even in antlr 3, this would have been a trivial task.

Richard Veach 12:55

doesn't our grammar handle control characters in normal strings? why does this not work in text block then

thats the only reason why I asked why it was split into new file 12:56

Nick Mancuso 12:56

Because there is real control characters in the too, now

RV

https://github.com/checkstyle/checkstyle/pull/8327/files#diff-005f4a4ab29fcdadc426ab80fbcda6c3R107

you mean like how this starts with information outside the string, like `\n'   12:57

Nick Mancuso   12:58

what line number? the link isn't taking me to a line

Richard Veach   12:58

107 in InputJava14TextBlocks.txt

its the extremely long text block   12:58

I haven't been involved in all discussions, so I may be missing alot, but   13:00
TEXT_BLOCK_CONTENT doesn't look very friendly to follow. I question why
everything isn't a separate token

are we trying to duplicate Javadoc text?   13:00

Nick Mancuso   13:01

Let me find the conversation, hang on

https://github.com/checkstyle/checkstyle/issues/7103#issuecomment-   13:01
645028044

I goes on through several posts   13:01

Richard Veach   13:03

I am looking at the original java too, its interesting what can be put outside the
string blocks

I see now why I don't think 1 grammar would be enough   13:03

Nick Mancuso   13:04

I also felt that this was something that could be built on, so having the extra
lexer now makes any additions trivial.

                                                                            13:04

Photo
Not included, change data exporting settings to download.
780x242, 31.2 KB

Example: change to this is one line of code...

Richard Veach   13:07

I don't see @romani involved in the tree discussion

this does seem very similar to javadoc printing   13:07

new lines in the file (not control characters) are handled by a separate   13:08
token

NM

Exactly.

Richard Veach   13:09

no where else in our Java grammar do we have a token like that

its slightly weird to add it just for text blocks   13:09

there is a complete separation between java and javadoc   13:09

your text block will be fully in java with no separation, thats the only thing   13:10
I think should be discussed

but I am not saying to stop your current PR   13:10

if we add a NEWLINE token to Java, it will have to be clear its only for   13:10
text block

maybe TEXT_BLOCK_NEWLINE   13:11

unless we will add new lines for everything, but that would be a huge   13:11
change

Nick Mancuso   13:12

NM

It's not in the PR, I've removed the NEWLINE node, to just have the single line
of content

I was just showing you a possibility some you mentioned that it was   13:13
difficult to read

*since   13:13

Richard Veach   13:13

RV

yep, I was talking about if we move forward with your possibility, which I like
more than the current raw view

13:14

https://github.com/checkstyle/checkstyle/blob/463d86026a0241f07b4218fe029
c0269baa40afe/src/test/resources-
noncompilable/com/puppycrawl/tools/checkstyle/grammar/java14/InputJava14
TextBlocks.java#L38

Nick Mancuso   13:15

NM

Ah, I see. Well, the NEWLINE token only lives in the text block lexer, if that
make it any better

Richard Veach   13:15

RV

would this be parsed as "{Number} {Char} {Char} {Char}"?

it has to be added to our TokenTypes class. Everything Java lives there.   13:15

would be no separation between different lexers   13:16

Nick Mancuso

Or esc char   13:19

Richard Veach   13:19

i meant in your future one where you separated things out better

doesn't really matter was just curious. I should look at the JLS   13:21

Nick Mancuso   13:27

Oh yes, we could do that

Even make an HTML node for the example above   13:28

Richard Veach   13:41

I believe it is out of order because of your push

if you look inside `mTEXT_BLOCK_LITERAL_BEGIN` inside generated lexer, 13:42
you can see you are doing the push first and then creating the token for the
block begin

content and end are inside the new grammar file. It is probably   13:43
processing those first before it gets the chance to return the begin back

but we already matched the begin, so the new grammar starts after it   13:44

I am not able to see where it is exactly happening, but that is my best   13:47
guess. This pushing is new to me since we have always used 1 grammar file

Nick Mancuso   13:50

You are right, but this isn't the behavior described by the documentation, which
is why I'm confused. I even tried not sharing the lexer state, and it still did not
work as intended. In fact, after digging into the antlr code, I don't know how this
would work as described in the documentation

Roman Ivanov   20:08

https://github.com/checkstyle/checkstyle/pull/8422 assigned to Richard , I need
to focus on patchfilter project for a bit

12 July 2020

Nick Mancuso   22:25

@esilkensen can you restart travis for
https://github.com/checkstyle/checkstyle/pull/8427#pullrequestreview-
446987298 ? It's been hung up for a few days now.

Erik Silkensen   22:26

In reply to this message

strange, yeah I will restart it... looks like it's all green there, maybe the
webhook got dropped on the way to github

NM

Plan for today: resolve review items, work on switch expression grammar.

NM

Nick Mancuso   08:41

Any mentors available to restart CI for
https://github.com/checkstyle/checkstyle/pull/8417 ?

and https://github.com/checkstyle/checkstyle/pull/8437   08:43

For the second one, my master was too old, but I rebased and pushed,   08:44
and travis didn't run again, plus some tests just hung up

ES

Erik Silkensen   08:45

In reply to this message

Restarted, looks like it's green now

NM

Nick Mancuso   08:53

Thanks, looks like this one hung, too
:https://github.com/checkstyle/checkstyle/pull/8427

ES

Erik Silkensen   08:54

In reply to this message

Restarted and looks like all is green except for the pr-age check, but I can see
that your branch is based off the latest, so I'm not sure what's up there - but it
doesn't seem like that should be a blocker for your PR in the meantime...
maybe someone else has an idea/can explain though

NM

Nick Mancuso   08:55

I just changed an input string, and pushed again

We'll see if that works   08:55

NM

Nick Mancuso   13:52

Almost EOD: I have addressed all items in PRs, and everything is green
except for those with the new CI. I have began to gather information for my
report on switch expressions, but I want to put a few more hours in tomorrow
before posting a report and asking for assistance. I have spent quite a bit of
time today reading JLS and studying other (PMD) grammar in an effort to find a
solution. Apparently, PMD thought this was ugly, too
https://github.com/pmd/pmd/blob/1ca84d56c15818d048c9fa922f82c11093773
1bf/pmd-java/etc/grammar/Java.jjt#L1639. The real challenge is going to be
minimizing the side affects that this grammar change creates; adding the
switch expression to unaryExpression generates changes throughout the
entire generated parser.

NM

Nick Mancuso   15:17

I have pushed my branch to github, in case anyone has time to check it out
and maybe offer some advice. Here is the offending rule, everything else works
great:

Even with x2 lookahead (a`la PMD) this is still a problem. I have factored 15:19 out the casting logic, to introduce the castExpression rule, and tomorrow will continue to try and isolate the problematic grammar so that I can observe exactly what's happening.

I've traced through side by side with the master branch while parsing the 15:21 offending structure, numerous times, but the

`unaryExpressionNotPlusMinus`

rule was very complex, which is why i am beginning to factor out other parts.

14 July 2020

Nick Mancuso   05:30

NM

plan for today : put in some more work on switch expression, address all PRs. I also need to update the text blocks PR to make the printed AST in TokenTypes.java look correct.

Nick Mancuso   09:31

NM

switch expression PR is up:https://github.com/checkstyle/checkstyle/pull/8449

There is still one issue with the grammar remaining; I have made this PR 09:32 draft status because of it, and will provide exact details of the issue in the second comment of the PR.

Nick Mancuso   09:55

NM

If any mentors have a minute, can you take a look at this post: https://github.com/checkstyle/checkstyle/pull/8422#issuecomment-655698638

I will be opening all the issues related to full records support affected 09:56 checks today, so please let me know if there is any that we can rule out before I make an issue for it.

Nick Mancuso   14:39

NM

I have made a detailed post about the issues I'm running into with the switch expression grammar; I am running out of ideas at this point, please take a look and let's try to figure this out: https://github.com/checkstyle/checkstyle/pull/8449#issuecomment-658278145

Erik Silkensen   15:33

ES

In reply to this message

Thanks Nick, I will check out your post tonight!

Nick Mancuso   19:28

NM

Sounds great. I have begun opening issues for records checks, should have it completed tomorrow.

15 July 2020

Erik Silkensen

My initial thought: I wonder about taking a step back and seeing if we can break this down into a couple of incremental steps. The JEP (looks like the latest is https://openjdk.java.net/jeps/361) summarizes:

"Extend switch so it can be used as either a statement or an expression, and so that both forms can use either traditional case ... : labels (with fall through) or new case ... -> labels (with no fall through), with a further new statement for yielding a value from a switch expression."

That makes me think of a possible sequence of PR's, like

- Extend switch so it can be used as either a statement or an expression
- Add support for the new case ... -> labels
- Add support for the new yield statement

where each could be implemented and tested on its own.

It sounds like the new case labels will be the hardest part - and I know this by itself isn't helping you get around the current issue - but let me know what you think: Does "switch expressions with traditional case labels" sound like a worthwhile first step to you? Is it too trivial to bother breaking off?

If you'd rather tackle it all at once, could you walk me through where you're at starting from a little higher level? e.g. your overall strategy, what examples your grammar is parsing currently, is it only hung up on casts/lambdas, etc.

I can tell you have already made some great progress - and I know by now you know a lot more ANTLR than me - but I will do my best to try and at least be a sounding board for you and understand what paths you've already been down, and hopefully help generate some new ideas

Nick Mancuso   04:29

NM

>f you'd rather tackle it all at once, could you walk me through where you're at starting from a little higher level?
Ok, where I am at right now is that the grammar can successfully parse all switch expression code that I can find, including lambdas in the expression, deeply nested switch expressions, etc. no problem. This change has effects throughout the generated file (as expected, expressions are everywhere), and one effect was to change the token sets that antlr generates, such that when parsing:

```
 public class TestClass {
void indexed() {
IntConsumer consumer = (IntConsumer) i -> {
};
}
}
```

The grammar now stumbles on the IDENT that is being typecasted, which we were able to parse before. The problem is that now that the tokenset has

Basically, we can still parse anything else, like the other two code snippets in the second post of the PR. We have lost the ability to parse the above expression only. So, by adding the optional "id" to the lambdaParameters, we can parse this code, but not the original way that is was parsed. 04:36

All attemps to "close off" the postfixExpression rule when parsing the typecast with lambdas have broken Checkstyle (400+ errors), or just not worked. 04:37

All attempts to use a ridiculous amount of lookahead have still failed. 04:39

Actually, I am going through the CI for switch PR, and see that this fails to parse now, too: 04:47
https://github.com/google/guava/blob/32f2d770f7117015385083ec6e755336ddf18f21/android/guava/src/com/google/common/io/BaseEncoding.java#L455
So, this is a bigger problem than I thought. I am going to take a break from this for a day and work on getting issues up for records related checks and tie up some loose ends.

@esilkensen I think your idea of breaking this into three parts is what we should do, but let's change the order a bit. statement or expression -> yield -> new labels ? 04:49

I think each part needs to be thoroughly tested, so this makes sense to break it up. The addition of switch expressions has effects throughout the entire generated parser, we should not take this lightly. 04:52

Do I just reference the same "Issue #6615:..." for each? What about input files? The PR for the input file has every sort of usage I could find, should I just amend the existing PR to be input for statement or expression, then make two more input file PRs? 04:59

Nick Mancuso   14:50

Almost EOD: I plan to continue going through all the check issues related to full records support, and complete creating all issues by tomorrow at some point. Mentors, I have rebased text blocks to remove conflicts, and fixed the printed AST in the TokenTypes.java javadoc, so it is ready for review. If we could get full record support merged this week, I can get all the check updates taken care of the week after next (next week really is my vacation). Please let me know the best way to proceed with breaking the switch expression PRs up, and I should be able to have the first two up before I go on vacation.

Nick Mancuso   16:02

About 30 issues opened, and about 40 to go:
https://github.com/checkstyle/checkstyle/issues/8452

Erik Silkensen   17:27

Awesome, that all sounds good - I am just reading your posts now and will read deeper later tonight

For breaking up the switch changes, Roman/Richard/Pavel know best but I'd

IMO could make sense to as you go expand or add to a test input so it includes just what you have implemented for so far, targeting that final one at the end that you created that exercises everything.

In reply to this message   17:28

Or I meant to say: I think it makes sense to include the input file along with the grammar change PR for switch

Nick Mancuso   17:29

Yes, I was thinking the same, that makes the most sense in this case. I'll get on this tomorrow.

Hopefully, I'll be done posting issues by noon tomorrow, then I'll get on those PRs.   17:30

Erik Silkensen   17:30

Sounds good. I will review any PR's I haven't yet tonight and see if I can help push on the records PR, and check out the issues you've been creating

Nick Mancuso   17:31

The issues for records checks are much more involved then the instanceof checks were, since we now have a new class like structure, and declarations thereof, so it is taking considerably longer for each one.

The good news is, that since I reused OBJ_BLOCK, maybe about half of them still work well.   17:32

Erik Silkensen   17:32

Makes sense, yeah that sounds good

16 July 2020

Nick Mancuso   06:19

I'm continuing to open issues today for records, and should be able to complete that today, and open at least one PR for the first installment of switch expression.

Nick Mancuso   12:58

Ok, all 68 issues have been submitted:
https://github.com/checkstyle/checkstyle/issues/8452

Maybe about 40% of these we may be able to close, but I really wanted to make sure that we were thorough since this is a major change.   13:01

Some interesting checks came up from my grep of the checks directory, and some that I didn't really think would apply, ended up being impacted. I did not know that you could have a main in a record!   13:02

Nick Mancuso   14:10

jdk14

Nick Mancuso   18:16

NM

EOD: getting ready to post in the email thread, I will post a link here after I do.
Mentors, please take a look at the records related issues that I posted, so that
we can approve the ones that need to be fixed/ supported, and close the
issues that don't apply. Also, it would be great to get full records merged to
make life a bit easier for check updates. Text blocks is also ready for review.

Erik Silkensen   20:19

ES

Thanks Nick — that all sounds good to me. Sorry I haven't got to anything yet,
I'm on call at work this week and things kind of caught on fire.

Did you have any luck with the switch grammar?

I will just reiterate my intention to do what I said I'd do last night, and will
certainly have time tomorrow evening and this weekend.

Thanks for all your work on this project and I hope you are able to enjoy a
vacation next week through these difficult times we are in!

Nick Mancuso   20:20

NM

I think I got it :)

Erik Silkensen   20:20

ES

Awesome

Nick Mancuso   20:22

NM

I completely rewrote the switch rules to be more granular and rely less on
existing rules, then reversed the order of two of the cast expression rules. I
figured out what was likely causing the problem: postfixExpression should
have been in UnaryExpressionNotPlusMinus (as the first rule)

instead, our grammar has it in the cast expressin rule   20:22

but moving it out broke checkstyle with over 400+ errors   20:23

I found all parsing pain points from CI and the other ones that I tested on, 20:23
and added all to the UT

I have one more trick up my sleeve if we run into anymore issues, but I   20:24
feel pretty good about what I did today.

https://groups.google.com/g/checkstyle-   20:33
devel/c/Uc7n5LsVMMw/m/Bn1c6Lz6CAAJ

Nick Mancuso   21:50

NM

Switch is green: https://github.com/checkstyle/checkstyle/pull/8449

Erik Silkensen   21:54

ES

That's awesome, great work @Nmancuso !!

ok to do, or would it make a problem with CI?

Basically, I excluded the input from being parsed, since we couldn't parse 22:07 it yet, then after I added support, I need to delete it. But I don't know if that would mess up CI. Does CI check all commits, or just the last one?

Erik Silkensen   22:27

I think you are good to go - yeah I believe it only runs for the top commit, or for example if multiple commits are pushed or merged in one event, then CI only for the top.

Anyway, but if there's something that needs to be shuffled after all to make CI happy - no problem, it can be done, the hard part is the actual grammar change ☺

Those are the most satisfying times in programming I think, fighting something for days on end and finally coming through, keeps you coming back for more.

17 July 2020

Nick Mancuso   04:27

Indeed! Plan for today: get reports posted to switch expression PR, make sure CI is green for all PRs, work on check updates for instanceof related checks.

Nick Mancuso   07:02

Can a mentor restart travis for
https://github.com/checkstyle/checkstyle/pull/8327

Erik Silkensen   07:44

In reply to this message
Restarted

Nick Mancuso   07:47

Thanks! Do you know what's up with the CI in this PR? I haven't seen this before, and it isn't really giving me any details about what to do to make it green: https://github.com/checkstyle/checkstyle/pull/8427

Erik Silkensen   07:51

In reply to this message

Ah, yeah in my experience code coverage CI checks failing like that can be kind of common unfortunately... IMO nothing that you need to do about it, I imagine the coverage check can be configured to make this green and in the meantime it shouldn't block something like that from merging.

Nick Mancuso   09:21

Ok, thanks. JOSM failed again, can someone restart Travis
https://github.com/checkstyle/checkstyle/pull/8529

by tomorrow afternoon. I have addressed all issues that are approved. I have addressed all PRs, and everything should be green. All grammar changes are submitted and green.

Nick Mancuso   15:37

Can a mentor restart CI for https://github.com/checkstyle/checkstyle/pull/8449 a bunch of jobs failed, I don't think it is anything that I did.

18 July 2020

Nick Mancuso   05:48

Regression reports are posted to switch expression PR: https://github.com/checkstyle/checkstyle/pull/8449#issuecomment-658278229

19 July 2020

Nick Mancuso   12:16

Hey guys, I was just thinking about the switch expression grammar, and realized I made a minor oversight in my logic to check for yield statements; I only checked if we were in a switch rule, and overlooked that we can also have yield statements in a switchlabeled block statement. I will update the PR after my vacation, but for now I will move it to draft status. Only a few lines need to be changed, so feel free to still look it over, since the important parts shouldn't be changing. I will re-run all reports afterwards.

Erik Silkensen   17:13

In reply to this message

Sounds good 👌

24 July 2020

Nick Mancuso   09:45

Hello all, I just wanted to let everyone know that I will be home from vacation tomorrow, and I'm willing to provide any updates we need or rebasing to get the full records support PR merged before Monday so that I can jump right into check updates Monday morning.

26 July 2020

Nick Mancuso   05:19

Ok, switch expression/ yield statement grammar PR is updated with all new reports, and it is ready to review: https://github.com/checkstyle/checkstyle/pull/8449

Nick Mancuso   05:48

@romanivanovjr Richard any chance we can get https://github.com/checkstyle/checkstyle/pull/8422 merged today?

Roman Ivanov 08:50

Richard, please review first (I try assist in other projects). Let me know if you have no time.

Just an idea: as TokenTypes is our most critical API, why we have no 08:53
`@since` in it

Nick Mancuso 09:16

@romanivanovjr
https://github.com/checkstyle/checkstyle/pull/8422#discussion_r460545472

Richard Veach 16:04

@romanivanovjr I was the only one who started since which was for checks and modules. It was only possible due to automation, which won't be the same for token types since it's an internal class.

Roman Ivanov 16:41

It is ok, we will catch up on the go

Nick Mancuso 17:24

@romanivanovjr Richard so, leave it as is, then?

Roman Ivanov 17:40

Keep `since` , you will show good example to others

Nick Mancuso 17:52

Ok, should I submit minor PRs to update the other tokens that have been merged already? I will update other grammar PRs tomorrow.

Actually, I think it's only one token to update: PATTERN_VARIABLE_DEF 17:54

Roman Ivanov 18:21

Please update only one token, that you added

27 July 2020

Nick Mancuso 03:33

Plan for today: begin working on records check updates, create minor PR to add `since` to PATTERN_VARIABLE_DEF token, and also add it to existing grammar PRs.

@esilkensen @pbludov we need review on 03:34
https://github.com/checkstyle/checkstyle/pull/8327 and
https://github.com/checkstyle/checkstyle/pull/8449

@romanivanovjr Richard we still need to get full records support PR 03:35
merged: https://github.com/checkstyle/checkstyle/pull/8422

merged

Roman Ivanov   10:52

contribution PR is merged

Nick Mancuso   11:13

Mentors, please restart Travis for:
https://github.com/checkstyle/checkstyle/pull/8563 ,
https://github.com/checkstyle/checkstyle/pull/8449 ,
https://github.com/checkstyle/checkstyle/pull/8327

Also, is there any way to work around this: https://travis-
ci.org/github/checkstyle/checkstyle/jobs/712214446 ?   11:13

Roman Ivanov   11:16

3 PR builds were restarted

> workaround   11:18

just put in Pr description `Issue #8267` on moment of PR creation. Our
validation is not that smart to detect that you have commit from another PR :).

I updated you PR description

Richard Veach   12:25

record PR merged

Nick Mancuso   18:45

EOD posted in email thread: https://groups.google.com/g/checkstyle-
devel/c/Uc7n5LsVMMw/m/YjoR1d0IBgAJ

Nick Mancuso   19:11

@esilkensen @pbludov I have three check update PRs posted:
https://github.com/checkstyle/checkstyle/pull/8568 ,
https://github.com/checkstyle/checkstyle/pull/8566 ,
https://github.com/checkstyle/checkstyle/pull/8564

Roman Ivanov   21:45

@Nmancuso , please keep PRs with grammar update like
https://github.com/checkstyle/checkstyle/pull/8327 with more priority than
updates in Checks

Please resolve conflict   21:47

Roman Ivanov   22:27

In Checks problems please fix exceptions first, false positives next, false
negative (missed violations) the last

NM

Plan for today: resolve all conflicts caused by records PR merge, then continue with records check updates.

Can someone please restart Travis   04:31
https://github.com/checkstyle/checkstyle/pull/8568

RI

Roman Ivanov   06:42

Restarted

RI

Roman Ivanov   08:12

https://github.com/checkstyle/checkstyle/pull/8327#pullrequestreview-456726516 - comments for text block PR

NM

Nick Mancuso   08:22

There was no official discussion regarding a second grammar file; but I don't know how else we would have the lexer emit the three separate tokens with correct line and col numbers, keep track of newlines, escapes, etc. and not cause ambiguities with the current lexer. Once the current lexer emits the TEXT_BLOCK_BEGIN token, it starts parsing the content of the text block as if it were regular java code without being able to switch to another lexer. I tried to make imaginary tokens, but getting the correct positions was very difficult and unreliable. Richard and I did have a brief conversation on Telegram regarding the second grammar file.

@romanivanovjr I can't link to the conversation about text blocks within   08:30
Telegram, but you can find the first comment that I posted by searching "Mentors, if you have any tine this weekend, please provide some feedback on the text blocks PR:" and go from there

NM

Nick Mancuso   10:54

In reply to this message

@romanivanovjr where does creation of new checks fall in this list?

NM

Nick Mancuso   18:14

Can a mentor restart Travis please?
https://github.com/checkstyle/checkstyle/pull/8574

RI

Roman Ivanov   20:49

Restarted , green.

>where does creation of new checks fall in this list?   20:52

My statement was unrelated to any existing issue. It was general. You can pin https://github.com/checkstyle/checkstyle/issues/8558 for more details, or we will close it as invalid issue

RI

Roman Ivanov   21:09

lexer can't handle that.

As explanation on why second grammar is created, ... please put a bit more details as comment in GitHub issue

Nick Mancuso   21:23

I will take care to provide more details as my first task tomorrow, then I will take a look at RequireThis. No exceptions at all were reported for full records support grammar. Can I ask the user in the issue to try the latest snapshot?

Roman Ivanov   22:30

It will unlikely happen, as he even ca not provide input file

Initial support was intended to be not reliable, so no worries that there   22:32
was some exception

29 July 2020

Nick Mancuso   04:55

Plan for today: first, I am going to update the text blocks PR with more details regarding second grammar, then update all PRs, then continue working on records check updates.

Roman Ivanov   07:08

@pbludov , as you finish with review, please reassign PR to next reviewer, example https://github.com/checkstyle/checkstyle/pull/8574 . PR should be always assigned or merged

Nick Mancuso   08:32

Mentors, we need to determine whether we are making a new check for the number of record components, or adding support to ParameterNumberCheck: https://github.com/checkstyle/checkstyle/issues/8521

This also needs approval and to be added to the project.

We need to add these PRs to project, too: https://github.com/search?   08:42
q=is%3Aopen+is%3Apr+author%3Anmancus1+archived%3Afalse

Nick Mancuso   13:01

Can a mentor restart Travis for https://github.com/checkstyle/checkstyle/pull/8574 and https://github.com/checkstyle/checkstyle/pull/8570 ? Also, please approve and add to project: https://github.com/checkstyle/checkstyle/issues/8577

Nick Mancuso   13:59

Restart Travis one more time please: https://github.com/checkstyle/checkstyle/pull/8570

Nick Mancuso

Nick Mancuso   18:00

EOD report: I have submitted 4 more PRs for records check updates today, and will continue to chip away at those tomorrow. All PR's are updated and items addressed. Please see above comments. @esilkensen I ping'd you for review in a few PR's since no one was assigned, I hope that's ok. Two of them need to have Travis restarted, they should be green after that.

30 July 2020

Nick Mancuso   06:34

Plan for today: address all items in PRs, keep working on check updates

Nick Mancuso   08:10

Can a mentor restart appveyor please:

https://ci.appveyor.com/project/Checkstyle/checkstyle/builds/34382087/job/bc6 16fl39fhcgbrq

Richard Veach   11:03

I restarted it.

Let me know if we need to discuss the escaping issue in the text block   11:04
PR.

Nick Mancuso   11:33

@Richard Thank you for the additional explanation. Forgive me, but I still don't understand the rationale behind further escaping the escapes. Admittedly, I am not an expert regarding text parsing, and probably should know more about the nuances of escape characters, especially as a contributor to a project that relies so heavily on knowledge of such things. My aim when writing the rules that I did write was to match the way that text is represented in our current string literal, just with the additional line terminator. I have added another comment:
https://github.com/checkstyle/checkstyle/pull/8327#discussion_r463192695

If you can provide me some example input and output, I will make the   11:38
changes that need to be made to get this merged; but right now I don't fully understand what our goal is here.

Richard Veach   13:52

If my 2nd explanation which provided some examples isn't enough. Then I create one from your input.

@romanivanovjr feel free to chime in if you understand that I'm talking   13:52
about.

Nick Mancuso   13:56

Thanks for your patience, I think I am having a hard time with this statement: " Imagine if someone put in String s = """\\n""";. As you have your AST

example would be this:

```
 String s = """
\\n""";
```

Richard Veach   14:56

RV

let me install your version of checkstyle and create an example

Nick Mancuso   14:57

NM

Thanks for bearing with me here

Richard Veach   15:07

RV

http://rveach.no-ip.org/checkstyle/checkstyle.php?action=view&config=72fba57f86eaf7df54f418b381c28422&code=e776858e5c3181c138bd97912392df37&checkstyle=checkstyle-nmancus1-pr-selectortextblocksgrammar-2020-07-30-all.jar&printTree=true&printSuppressions=

line 3, it prints 3 `\n\n\n`   15:08

one should be the text `\\n` . You can't tell the difference between the 3   15:08
in the AST print out

Nick Mancuso   15:09

NM

I think I know what you mean, except that two should print like that, since they
are true escapes, right?

There are two real line terminators and one `\n`   15:10

Richard Veach   15:10

RV

yes, 2 are real escapes and should be printed s they are now

my concern was the text (non-escape)   15:10

if I wanted to build my own parser for my own usage, how do I tell text   15:11
from non-text

my owner parser meaning using the printout from the AST tree   15:11

Nick Mancuso   15:13

NM

This is a good point. So you are proposing that we preface every escape
character with an imaginary `\` ?

Richard Veach   15:13

RV

I wouldn't call it an imaginary `\` . Think of it as more printing the string as it
would appear in a Java string

if you did `String s = "\n";` you wouldn't print out `n` if your printed it 15:14

but yea, preface every `\` with another `\`   15:15

NM

This isn't consistent with our treatment of string literals, why deviate?

Richard Veach   15:16

RV

String literals don't contain true line controls like text blocks can, right?

that would be my argument why its different, but @romanivanovjr can   15:16
chime in on his thoughts.

Nick Mancuso   15:19

NM

Ok, so the idea here is that we want to make it easy to parse the
TEXT_BLOCK_CONTENT nodes right?

I just want to make sure that we are on the same page, since there are a   15:29
few different ways to approach this.
1. Make the text block ast easy for humans to read
2. Make it consistent with the current string literal
3. Make it easy to parse

Nick Mancuso   15:50

NM

You are correct about regular string literals not containing true line control
characters, sorry I missed that question.

Nick Mancuso   19:04

NM

EOD report in email thread: https://groups.google.com/g/checkstyle-
devel/c/Uc7n5LsVMMw/m/A16uQX6xAQAJ

31 July 2020

Nick Mancuso   04:01

NM

Plan for today: keep working on records check updates, and address all items
in PR's

Roman Ivanov   07:11

RI

@Nmancuso , please do sooner
https://github.com/checkstyle/checkstyle/pull/8327/files#r463634923

Nick Mancuso   07:12

NM

@romanivanovjr I'm on it

Roman Ivanov   07:21

RI

https://github.com/checkstyle/checkstyle/pull/8327#discussion_r463639758 -
one more comment, please clarify

Nick Mancuso   11:05

NM

https://github.com/checkstyle/checkstyle/pull/8327#discussion_r463733113

Nick Mancuso

know what else we need to get text blocks merged, so that once that is done I
can identify checks and get issues posted. What is a good way to identify
checks affected by merging text blocks? There are obvious ones, like
MultipleStringLiteralsCheck and IllegalTokenText, but what else? Thanks for
the kind words and good feedback in my review!

Roman Ivanov   17:40

https://checkstyle.org/config_misc.html#AvoidEscapedUnicodeCharacters

Search for `TokenTypes.STRING_LITERAL` in Checks   17:42

https://checkstyle.org/config_coding.html#EqualsAvoidNull   17:43

https://checkstyle.org/config_coding.html#StringLiteralEquality 17:47

Nick Mancuso   18:45

Can anyone check the wercker log for me?
https://app.wercker.com/checkstyle/checkstyle/runs/build/5f24b1badac563000
8b8bcdb

Erik Silkensen   19:06

In reply to this message

```
 Exception in thread "main" java.lang.Error: Error was thrown
while processing /pipeline/source/.ci-
temp/contribution/checkstyle-tester/src/main/java/local-
checkstyle/target/test-
classes/com/puppycrawl/tools/checkstyle/detailastimpl/InputDeta
ilAstImplJustToMakeStackoverflowError.java
at
com.puppycrawl.tools.checkstyle.Checker.processFiles(Checker.ja
va:319)
at
com.puppycrawl.tools.checkstyle.Checker.process(Checker.java:22
1)
at
org.apache.maven.plugins.checkstyle.exec.DefaultCheckstyleExecu
tor.executeCheckstyle(DefaultCheckstyleExecutor.java:202)
at
org.apache.maven.plugins.checkstyle.AbstractCheckstyleReport.ex
ecuteReport(AbstractCheckstyleReport.java:533)
at
org.apache.maven.plugins.checkstyle.CheckstyleReport.executeRep
ort(CheckstyleReport.java:57)
at
org.apache.maven.reporting.AbstractMavenReport.generate(Abstrac
tMavenReport.java:255)
at
org.apache.maven.plugins.site.ReportDocumentRenderer.renderDocu
ment(ReportDocumentRenderer.java:219)
```

```
    at
org.apache.maven.doxia.siterenderer.DefaultSiteRenderer.render(
DefaultSiteRenderer.java:135)
    at
org.apache.maven.plugins.site.SiteMojo.renderLocale(SiteMojo.ja
va:175)
    at
org.apache.maven.plugins.site.SiteMojo.execute(SiteMojo.java:13
8)
    at
org.apache.maven.plugin.DefaultBuildPluginManager.executeMojo(D
efaultBuildPluginManager.java:137)
    at
org.apache.maven.lifecycle.internal.MojoExecutor.execute(MojoEx
ecutor.java:210)
    at
org.apache.maven.lifecycle.internal.MojoExecutor.execute(MojoEx
ecutor.java:156)
    at
org.apache.maven.lifecycle.internal.MojoExecutor.execute(MojoEx
ecutor.java:148)
    at
org.apache.maven.lifecycle.internal.LifecycleModuleBuilder.buil
dProject(LifecycleModuleBuilder.java:117)
    at
org.apache.maven.lifecycle.internal.LifecycleModuleBuilder.buil
dProject(LifecycleModuleBuilder.java:81)
    at
org.apache.maven.lifecycle.internal.builder.singlethreaded.Sing
leThreadedBuilder.build(SingleThreadedBuilder.java:56)
    at
org.apache.maven.lifecycle.internal.LifecycleStarter.execute(Li
fecycleStarter.java:128)
    at
org.apache.maven.DefaultMaven.doExecute(DefaultMaven.java:305)
    at
org.apache.maven.DefaultMaven.doExecute(DefaultMaven.java:192)
    at org.apache.maven.DefaultMaven.execute(DefaultMaven.java:105)
    at org.apache.maven.cli.MavenCli.execute(MavenCli.java:957)
    at org.apache.maven.cli.MavenCli.doMain(MavenCli.java:289)
    at org.apache.maven.cli.MavenCli.main(MavenCli.java:193)
    at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0
(Native Method)
    at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(
NativeMethodAccessorImpl.java:62)
    at
```

```
at
org.codehaus.plexus.classworlds.launcher.Launcher.launchEnhance
d(Launcher.java:282)
at
org.codehaus.plexus.classworlds.launcher.Launcher.launch(Launch
er.java:225)
at
org.codehaus.plexus.classworlds.launcher.Launcher.mainWithExitC
ode(Launcher.java:406)
at
org.codehaus.plexus.classworlds.launcher.Launcher.main(Launcher
.java:347)
Caused by: java.lang.StackOverflowError
at
com.puppycrawl.tools.checkstyle.checks.whitespace.ParenPadCheck
.processExpression(ParenPadCheck.java:348)
```

Sorry, on phone, but in case that's helpful ... ParenPadCheck SO error?   19:06

Nick Mancuso   19:14

Yeah, that randomly shows up for me during check regression testing. Can you restart it for me?

Erik Silkensen   19:18

In reply to this message

Restarted, I will keep an eye and can kick it again if necessary

(Also, I wonder if Roman could grant you privileges to restart CI?)   19:19

Roman Ivanov   19:21

I grant permission only if I became confident in no misusage. Simply change of commit and push can restart CI (but will restart all CIs)

Nick Mancuso   19:23

>Simply change of commit and push can restart CI

Is that ok? I didn't want to waste resources.

Erik Silkensen   19:28

I understand, for sure - and I appreciate you trusting me, and I don't mean to be looking to offload responsibilities as a mentor.

Nick, as far as I know it's more a question of your time - more efficient for you to be able to restart a single CI run rather than the whole deal, obviously, but I don't think (I don't know if this is true) it's costing the Checkstyle org $$ to kick off extra CI for example

That said, maybe another thought is - for flaky, intermittent CI, if it's not   19:31
blocking you in some way (i.e. you'd be waiting for a review anyway) I think

Nick Mancuso   19:42

Yes. Probably 1 out of 10 check regression cycles fails on ParenPad with that file. Did you notice the name of the file? :)

Erik Silkensen   19:44

Yes, was confused by the red X since it seems the file is producing the desired effect haha

Roman Ivanov   19:52

> I didn't want to waste resources.

I do not pay for CI :), if you see no high activity in PRs, you can do this

<center>1 August 2020</center>

Nick Mancuso   09:57

Can anyone provide me with the wercker failure please?
https://app.wercker.com/checkstyle/checkstyle/runs/build/5f25997cdac5630008b8d23c

Erik Silkensen   10:13

In reply to this message

```
 A problem occurred evaluating root project 'spring-
integration'.
> Could not resolve all files for configuration ':spring-
integration-stomp:compileClasspath'.
> Could not find io.projectreactor.netty:reactor-netty:.
Required by:
project :spring-integration-stomp
```

Nick Mancuso   10:14

Thanks, I'm not familiar with that, is that something that I did?

Erik Silkensen   10:16

I don't think so, looks like all builds are failing

Roman Ivanov   17:31

@esilkensen and @pbludov , please read
https://github.com/checkstyle/checkstyle/wiki/Post-merge-actions , skipping of this cause master build to fail

Erik Silkensen   17:44

In reply to this message

Thanks @romanivanovjr - sorry about that - yes, I was not doing the first two steps. I will track down the PR's I merged tonight

New token: is either bug (if added to required set) or new features 18:15

It matter a lot for plugins 18:15

Nick Mancuso 18:27

What else do we need to get text blocks merged? I plan to put in a few hours tomorrow, and can take care of it then.

Roman Ivanov 19:12

Richard , please try to find time to do final review of text block

2 August 2020

Nick Mancuso 11:23

@romanivanovjr Richard can we discuss the addition of a new token?
https://github.com/checkstyle/checkstyle/pull/8449#discussion_r462380919

3 August 2020

Nick Mancuso 05:23

Plan for today: address all items in PRs, submit a PR for new check ParameterNumberCheck, and then keep working on records check updates.

I will also post issues related to text blocks, and post in email thread. 05:23
(New one for August.)

Nick Mancuso 07:50

I think I have discovered a potential issue with the full records grammar:

```
 // record declaration, note that you cannot have modifiers in
this case
| recordDefinition[#null]
```

This is creating a problem when checking for the access modifiers. In the 07:54 grammar for a class definition, the mods AST is not built, so an empty AST is passed to the classDefinition rule. Then, in ScopeUtil, the empty modifiers AST is marked as package visibility. With the current records implementation, Scopeutil is throwing a NPE, because instead of the modifiers AST just being "empty" it is null.

You can see the NPE here: https://checkstyle-diff-reports.s3.us-east- 07:55
2.amazonaws.com/full-records-
RecordComponentNumberCheck_2020133606/reports/diff/checkstyle/index.ht
ml

I propose that we create an empty modifiers AST in traditionalStatement 08:06 to pass to the recordDefinition rule.

Should I open an issue in github for further discussion? 08:12

jdk14

Nick Mancuso   08:32

After trying a few things, creating an artificial modifiers AST seems error-prone and problematic. I will just update ScopeUtil unless there are any objections.

Roman Ivanov   08:40

Please create issue and put all details to it

Nick Mancuso   08:40

Ok, I'm on it.

Nick Mancuso   08:58

https://github.com/checkstyle/checkstyle/issues/8598

Roman Ivanov   11:43

se comment

Nick Mancuso   11:45

Ok, I'll just use OuterTypeFilename, since that is affected as well. I didn't know if it was ok to submit a bug report for my new check since it is not merged yet

Nick Mancuso   12:24

I've tried the grammar modification on all affected issues, and it works great. I was not constructing the empty AST properly when I posted the above comment "After trying a few things, creating an artificial modifiers AST... " I have posted the proposed grammar change to the issue.

Roman Ivanov   12:25

please make sure you have example of how it affecting AST structure

Nick Mancuso   12:26

Ok. I will post now.

It just adds an empty "MODIFIERS" node to record definitions in methods 12:27 only. Otherwise, all record definitions already had that node, empty or not.

Nick Mancuso   12:45

Ok, issue is updated. I have a grammar patch ready to submit if we decide to go that route.

Nick Mancuso   16:44

EOD: all reports are running for the records grammar patch, should be done later tonight. I will post ASAP. All PR's are updated, and I have a lot that are just waiting for the patch to get merged. I think that I will be able to have all check updates Pr's for records support posted by Thursday.

Nick Mancuso

to re run reports, do I.

Email posted: https://groups.google.com/g/checkstyle-devel/c/hp05kSVxGJ4   17:30

Erik Silkensen   20:17

ES

In reply to this message

No objections here, that all sounds right to me

4 August 2020

Nick Mancuso   05:36

NM

Plan for today: post issues for all checks related to text blocks. If there are any that cause NPE, I will start on them today, and push off records check updates.

Nick Mancuso   06:17

NM

Hey, I just wanted to let you guys know that there are tornadoes in my area, and the power has been intermittent. If you don't hear from me/ see activity today, I have probably lost power/ taken shelter.

Nick Mancuso   07:50

NM

Guys, my house has sustained a fair amount of damage and my property is devastated. I will be MIA for a few days. We won't have electricity for quite a while most likely

Roman Ivanov   07:51

RI

What happened ?

Nick Mancuso   07:51

NM

Tornado

I have four trees on my house, and lost about 20 on my property. Roof is   07:52
damaged, shingles and siding in some places are gone.

Roman Ivanov   07:57

RI

https://www.wdel.com/news/isaias-upgraded-back-to-hurricane-heavy-rain-tornado-possible-in-delaware-tuesday/article_f3bedf8e-d5cc-11ea-ba2f-4fd9e7648f38.html

Erik Silkensen   08:00

ES

Sorry Nick, stay safe!!

Nick Mancuso   08:04

NM

Thank you, it's pretty ugly but we are safe.

08:08

Photo

08:08

Photo

Not included, change data exporting settings to download.
1280x960, 389.2 KB

RV

Richard Veach   09:07

Didn't realize you were in the us too. No damage here as I am further north.
Mostly just rain. Sorry to hear about your house.

RI

Roman Ivanov   09:39

@pbludov , @esilkensen , please fix
https://github.com/checkstyle/checkstyle/issues/8604 , we leaked stuff , and
master build is red and all new PRs also red

5 August 2020

NM

Nick Mancuso   03:57

Let me know what else I can do to get
https://github.com/checkstyle/checkstyle/pull/8600 merged. Once that is
merged I can update all my PRs for check updates so that they are green.

RI

Roman Ivanov   06:16

minor modification is required, please do.

NM

Nick Mancuso   06:18

Thanks, I'll take care of it tonight. I have to cut trees off the house while the sun
is out.

RI

Roman Ivanov   08:12

@pbludov , please do not forget to put label on issues before you close issue.
Without this action there is master build failure

Such minor build failure can hide bigger problem   08:12

FYI: I fixed 3 CIs today for master. PRs should be rebased   08:14

NM

Nick Mancuso   19:37

Hello all, the power company says that we will have service back by tomorrow,
and all trees are off my house. I will update all PRs tomorrow and hopefully
submit some more. I will also submit all issues related to text blocks .

Everyone, thanks again for bearing with me, and the well wishes.   19:39

I just updated https://github.com/checkstyle/checkstyle/issues/8598 and   19:53
https://github.com/checkstyle/checkstyle/pull/8600

6 August 2020

I need to be sure about   06:29
https://github.com/checkstyle/checkstyle/pull/8600#discussion_r465713737

Nick Mancuso   06:30

I posted in the issue last night:
https://github.com/checkstyle/checkstyle/issues/8598#issuecomment-669652266 is this ok?

Roman Ivanov   06:58

yes, now I see it, thanks a lot. Merged.

Nick Mancuso   08:40

Ok everyone. I have power back now, and it is raining, so no tree cutting. What is priority? Getting pitest coverage back up?

Roman Ivanov   09:15

If grammar update is blocked by mentors, please keep doing Checks updates

Pitest in the last   09:15

Nick Mancuso   09:17

Ok. I will
#1. Update existing PRs
#2. Finish submitting issues for text blocks so that we can identify any checks that cause exceptions
#3. Start working on checks (if any) that cause exceptions from text blocks

Nick Mancuso   09:45

All PRs are updated and rebased. We have two that are hanging:
https://github.com/checkstyle/checkstyle/pull/8575 and
https://github.com/checkstyle/checkstyle/pull/8578. These two check updates are good to merge with review from @esilkensen and @pbludov only, right @romanivanovjr ?

Roman Ivanov   12:02

https://github.com/checkstyle/checkstyle/pull/8575 pitest failure

Nick Mancuso   12:06

Ok, I will take care of it tonight, the rain stopped so I am back outside doing some repairs if that's ok.

Nick Mancuso   19:00

can anyone explain this to me:

https://teamcity.jetbrains.com/buildConfiguration/Checkstyle_IdeaInspecti 19:00
onsPullRequest/3052229?buildTab=Inspection

@romanivanovjr , what is the verdict on   19:05
https://github.com/checkstyle/checkstyle/issues/8513?

Richard Veach   20:17

@Nmancuso I am seeing some single quotes in the property files for the
changes you added. Maybe that is the issue as single quotes need to be
doubled to be printed as normal in property files.

Nick Mancuso   20:19

Great, I will try that! Thank you very much. I don't have a lot of experience
dealing with property files, and googling that specific violation didn't turn up
anything worthwhile.

Richard Veach   20:19

I believe its a false violation from their parser failing on the single quote.

https://www.mscharhag.com/java/resource-bundle-single-quote-escaping 20:20

`d'enregistrement` should be written as `d''enregistrement`   20:21

I tried the changes locally and the violations went away. So that is the   20:23
issue.

If your not familar with IntelliJ, that is basically what TeamCity is running.   20:24
You can run the violations locally.

Nick Mancuso   20:27

It's strange because those violations weren't showing up locally. I tried to set
up the Team City plugin for intellij, in hopes that it would highlight the exact
issue(the TeamCity reports offer to link you to the violations in Intellij), but I
could not figure out how to get it set up.

Richard Veach   20:28

yea, imo IntelliJ is not very friendly

I don't remember having any problems when I followed   20:28
https://checkstyle.org/idea.html

Roman Ivanov   20:30

Please make sure you use the same version on local as in teamcity

Nick Mancuso   20:31

I hadn't considered that. I have just been using the latest build from AUR

I have been having trouble with pitest still, it seems that violations are   20:31
changing between runs

Richard Veach   20:32

I don't use the Team City plugin, running inspection inside IntelliJ does it for
me. IMO, it is too much trouble to keep IntelliJ the same as TC. I haven't had

Roman Ivanov   20:32

Yes, pitest was not stable for me too

Nick Mancuso   20:35

I am going to update the issue that you made. I think we may need to fix this sooner rather than later.

The textBlockLexer coverage, that is.   20:35

Nick Mancuso   20:51

https://github.com/checkstyle/checkstyle/pull/8575 is still failing from
`selector.addInputStream(filter, "filter");`

I know you said to wait on this, but I would like to try and submit a PR to   20:53
fix coverage of `JavaParser.java` tomorrow.

Roman Ivanov   20:58

You are always welcome

https://github.com/checkstyle/checkstyle/issues/8513 is approved   20:58

Nick Mancuso   21:02

In my new PR, I will just make my changes, then undo everything from
https://github.com/checkstyle/checkstyle/pull/8605 ?

Roman Ivanov   21:04

Do simple revert of my commit and separate commit fix what is required

Nick Mancuso   21:05

Ok, I will do that. I will also make sure to run pitest 8 a few times locally before pushing.

7 August 2020

Nick Mancuso   04:22

Plan for today: try to submit PR to resolve pi-test issue, keep working on records check updates

Nick Mancuso   06:34

Since we are getting close to the end of the summer, I just wanted to clarify what remains to be done:
#1. Finish records check updates
#2. Finish text blocks check updates
#3.Get switch/ yield PR merged
#4. Identify affected checks for #3
#5. Fix affected checks from #3
??

please prioritize `Get switch/ yield PR merged` as most important   06:37

Nick Mancuso   06:44

All is green, and two approvals for swtich/ yield

Just need review from @romanivanovjr and Richard   06:44

Nick Mancuso   11:17

Can someone restart Travis please: https://travis-ci.org/github/checkstyle/checkstyle/builds/715879857

Erik Silkensen   11:56

In reply to this message

Sorry, it looks like I no longer have permission to restart CI

Roman Ivanov   12:46

Restarted. @esilkensen , you should have rights, you do have RW access to repo, Travis match permission by code access

Erik Silkensen   12:48

In reply to this message

Photo

Not included, change data exporting settings to download.

591x1280, 69.0 KB

It's strange, something seems to have changed recently - it looks like this 12:48 to me now

Where on my personal repos there is a restart button below the "2 hours 12:49 ago" text for example

Roman Ivanov   12:50

Strange

Nick Mancuso   13:04

EOD report: submitted 5 Pr's for check updates related to records, 2 PRs for bug/ documentation fixes, and finished posting text blocks related issues. Mentors, we need https://github.com/checkstyle/checkstyle/issues/8614 added to our project, please.

I plan to do some more tonight, so I can address any review items then.   13:05 Is it possible to get switch/ yield merged this weekend?

Nick Mancuso   15:46

If anyone has a moment, can you restart Travis: https://travis-ci.org/github/checkstyle/checkstyle/jobs/715900331

jdk14

Richard Veach   18:46

RV

restarted travis job

Nick Mancuso   18:46

NM

Thank you. What about that pitest?