

Conversion Tools - Feature #1579

review and merge 12/2011 conversion fixes into main P2J code base, regression test, and fix as needed

10/10/2012 10:33 PM - Eric Faulhaber

Status:	Closed	Start date:	10/11/2012
Priority:	Normal	Due date:	
Assignee:	Constantin Asofiei	% Done:	100%
Category:		Estimated time:	24.00 hours
Target version:		version:	
billable:	No		
vendor_id:	GCD		
Description			

History

#1 - 10/10/2012 11:06 PM - Eric Faulhaber

- File *conversion_heavy_preprocessor.png* added

In December 2011, Greg and I made a number of fixes to P2J to enable early work on a large customer project (updates attached). This involved various parser fixes, backing the report generator with the H2 database (the latest version at the time) to prevent out of memory errors, and a handful of other conversion fixes necessary to convert a sample program from the project. While these changes worked for the new project, something in them broke conversion for our pilot project, at least on my development system. These changes need to be reviewed and integrated into the main P2J code base, then regression tested.

#2 - 10/10/2012 11:08 PM - Eric Faulhaber

The following is the text of an email I sent to Greg regarding one of the fixes, which had to do with schema parsing performance. There is no indication that this particular fix is the cause of any problem; it is included only because it is one of the few things recorded about that session of fixes:

Greg,

I eliminated the long pauses caused by sorting the NameNode objects when working with the SchemaDictionary. A fixed NameNode.java and Namespace.java are attached. This is an interim fix which leaves the processing in memory, but eliminates the lazy sorting of the NameNode list in a Namespace by always keeping the nodes in sorted order. This brought down the time to parse a single XXXX program from 26:52 down to 01:38. I have not yet run this through regression testing with YYYY.

I tested this by running full conversion of YYYY locally and comparing the converted results against a previous conversion. There were some unrelated changes with frames and browses (my last conversion run locally was some time in November, and it looks like some UI rule sets have changed since then), but nothing that appeared to be a regression from the NameNode/Namespace change. The time to run did not seem to improve with YYYY. In fact, annotations were considerably slower, but I'm guessing that's related to the frame/browse change, since I got many fewer frame-related warnings compared to my previous run.

I also re-ran F2 on all the XXXX GUI programs to see what effect my changes might have. I compared the resulting cache files with your original run. A lot of them were different, but this appears related to an alert-box message in many of the .w files that specifies the fully qualified file name of the program when the procedure is not RUN PERSISTENT. Since my project directory is not in /home/ges, my results were different. There were too many diffs to check them all, but I spot-checked many and it was always this difference.

The run took around 11.5 hours, but I ran it in an encrypted folder, so it's not an apples/apples comparison with your original run. Although the fix cut down the time early in the process, most of the overall CPU time is spent in java.io.File.listFiles(), called over 110,000 times during preprocessing from com.goldencode.p2j.util.Utils.matchPathCaseInsensitively(). Also attached is an image of the hot spots detected by jvisualvm, which shows over 80% of the overall time spent in that method. We should get a huge win when we optimize this method to be backed by the database (even without multi-threading).

A side effect of this exercise was to remind me how difficult it will be to re-architect SchemaDictionary without breaking all the subtle nuances of behavior we encoded into it.

Thanks,

Eric

Hot Spots - Method	Self time [%]	Self time	Self time (CPU)	Invocations
java.io.File. listFiles ()		29623391 ms (71.7%)	29623391 ms	110933
com.sun.org.apache.xalan.internal.xsltc.trax.TransformerImpl. transform ()		1486623 ms (3.6%)	1486623 ms	13034
javax.xml.parsers.DocumentBuilder. parse ()		1151980 ms (2.8%)	1151980 ms	6827
java.util.HashMap. put ()		686863 ms (1.7%)	686863 ms	6890
java.lang.String. toLowerCase ()		633802 ms (1.5%)	633802 ms	6379
com.sun.org.apache.xerces.internal.dom.ElementImpl. setAttribute ()		573021 ms (1.4%)	573021 ms	5682
com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderFactoryImpl. newDocumentBuilder ()		496189 ms (1.2%)	496189 ms	5016
java.io.FilterReader. read ()		484945 ms (1.2%)	484945 ms	8101
java.io.PrintStream. println ()		362734 ms (0.9%)	362734 ms	3232
java.io.File. getName ()		348842 ms (0.8%)	348842 ms	3596
java.io.PrintStream. print ()		345088 ms (0.8%)	345088 ms	2635
java.lang.StringBuffer. insert ()		257340 ms (0.6%)	257340 ms	2519
com.sun.org.apache.xerces.internal.dom.ElementImpl. getAttribute ()		217153 ms (0.5%)	217153 ms	2195
java.io.File. getCanonicalPath ()		205242 ms (0.5%)	205242 ms	2032
com.goldencode.p2j.uast.KeywordDictionary. addKeyword ()		193002 ms (0.5%)	193002 ms	8813
com.sun.org.apache.xerces.internal.dom.ElementImpl. getNodeName ()		183687 ms (0.4%)	183687 ms	1816
java.lang.Long. toString ()		168230 ms (0.4%)	168230 ms	1670
java.io.File. isDirectory ()		147228 ms (0.4%)	147228 ms	1455
com.goldencode.p2j.pattern.AstWalker. configureResolver ()		134253 ms (0.3%)	134253 ms	1390
com.goldencode.p2j.pattern.AstSymbolResolver. registerTree ()		119944 ms (0.3%)	119944 ms	2337
java.lang.String. getBytes ()		111154 ms (0.3%)	111154 ms	1115
com.goldencode.p2j.uast.ProgressParser. initializeAttributesDictionary ()		109056 ms (0.3%)	109056 ms	1665
java.lang.Class. newInstance ()		105508 ms (0.3%)	105508 ms	1057

#3 - 10/17/2012 04:55 AM - Constantin Asofiei

- Status changed from New to WIP

#4 - 10/17/2012 09:38 AM - Constantin Asofiei

Greg/Eric,

A few questions and findings:

1. By "conversion for the pilot project broke with the update" you mean MAJIC ? Because I've converted the pending MAJIC using the new updates, and the generated sources are the same (the elapsed time was also OK) ... Can you elaborate what part of the project or conversion process broke ?
2. Where can I find info on how to run the conversion reports?
3. I've converted a few test programs (with or without temp tables) and the com.goldencode.p2j.util.Utils.matchPathCaseInsensitively() was not executed at all... can you remember how you concluded that this was the faulty part ?
4. I've finished taking a look at the updates and all the work I see is to finish adding some javadoc and other minor edits/cleanups... is this enough or should I try to isolate and solve the listFiles problem ?

#5 - 10/17/2012 09:52 AM - Constantin Asofiei

Constantin Asofiei wrote:

Greg/Eric,

A few questions and findings:

1. By "conversion for the pilot project broke with the update" you mean MAJIC ? ~~Because I've converted the pending MAJIC using the new updates, and the generated sources are the same (the elapsed time was also OK) ...~~ (sorry, I forgot to rebuild P2J before conversion, I'll re-run it). Can you elaborate what part of the project or conversion process broke ?
2. Where can I find info on how to run the conversion reports?
3. I've converted a few test programs (with or without temp tables) and the `com.goldencode.p2j.util.Utils.matchPathCaseInsensitively()` was not executed at all... can you remember how you concluded that this was the faulty part ?
4. I've finished taking a look at the updates and all the work I see is to finish adding some javadoc and other minor edits/cleanups... is this enough or should I try to isolate and solve the `listFiles` problem ?

#6 - 10/18/2012 04:32 AM - Constantin Asofiei

- File `ca_upd20121018a.zip` added

I've found the problem, it was caused by the `uast/progress.g` changes related to "optional `column_spec` clause" for a define browse statement. The bug was that the `KW_WITH` clause was emitted as a child for the `DEFINE_BROWSE` node instead for the `DISPLAY` node (when the `DISPLAY columns...` clause exists). My problem is that the changes you've made in the `def_browse_stmt` rule (on line 8973) assumes that the `KW_WITH` can appear without the `DISPLAY` clause... currently, the tests I've attached have the `DISPLAY columns...` clause for the `DEFINE_BROWSE` statement, so that `frame_generator.xml` will pass, but please confirm I should go ahead and assume the entire `DISPLAY columns...` clause can be optional.

The `ca_upd20121018a.zip` has a few testcases and the changed `braces.g` (based on the `ges_upd20111213a.zip` version), but my problem is that I think the support for "optional `query_reference` clause" is still incomplete.

I say this because although all tests pass conversion, following have compile problems:

1. has `query_reference`, has `DISPLAY columns...`, has `WITH` clause - pass conversion, pass compile
2. has `query_reference`, has `DISPLAY` but with empty col list, has `WITH` clause - pass conversion, pass compile
3. no `query_reference`, has `DISPLAY columns...`, has `WITH` clause - pass conversion, no compile
4. no `query_reference`, has `DISPLAY` but with empty col list, has `WITH` clause - pass conversion, no compile
5. no `query_reference`, has `DISPLAY` but with empty col list, no `WITH` clause - pass conversion, no compile

The compile fails for all cases with no `query_reference`, as the conversion rules emit an `fFrameFrame.widgetWCustomerBrws().registerQuery` call with only a parameter (it should have emitted null for query reference). Should I fix this too ?

#7 - 10/18/2012 10:30 AM - Eric Faulhaber

Regarding your questions in update 5 above:

1. By "conversion for the pilot project broke with the update" you mean MAJIC ? Because I've converted the pending MAJIC using the new updates, and the generated sources are the same (the elapsed time was also OK) ... (sorry, I forgot to rebuild P2J before conversion, I'll re-run it). Can you elaborate what part of the project or conversion process broke ?

Yes, Majic. The problems you have found match the ones I saw (sorry I didn't respond to your update earlier), and in fact you have gone way further in diagnosing them.

2. Where can I find info on how to run the conversion reports?

See the Reporting chapter of the Conversion Handbook.

3. I've converted a few test programs (with or without temp tables) and the `com.goldencode.p2j.util.Utils.matchPathCaseInsensitively()` was not executed at all... can you remember how you concluded that this was the faulty part ?

Sorry, I've confused the issue by including the above e-mail. That analysis was about an earlier performance problem during preprocessing of the new customer's application, which was fixed with my Namespace/NameNode updates. It was not about the compile issues we saw with Majic later. This code path may be specific to the way the new app uses preprocessing directives.

4. I've finished taking a look at the updates and all the work I see is to finish adding some javadoc and other minor edits/cleanups... is this enough or should I try to isolate and solve the listFiles problem?

If you can think of a fix for the listFiles problem, let's discuss it, but it sounds like you haven't recreated the problem. That is a lower priority performance issue anyway (though we will have to fix it relatively soon), while this task is concerned primarily with integrating and cleaning up the fixes already made, while eliminating the compile failure side effects they produced.

Regarding update 6 above:

Yes, please fix the registerQuery problem as well.

Greg: can you answer the question about the DISPLAY columns?

#8 - 10/19/2012 09:51 AM - Constantin Asofiei

- % Done changed from 0 to 60

Running the tests in 4GL showed the following:

1. entire DISPLAY columns... clause is optional. If it is missing, "size cols by rows" clause is required for the DEFINE BROWSE (4GL compilation error). IMO this does not pose a problem for us, because we assume we convert only valid 4GL code.
2. query clause is optional. But, if query is missing, DISPLAY must be missing too (found no way to compile a DEFINE BROWSE statement without a QUERY clause but with a DISPLAY columns clause).

At this time, I have fixes for both the optional query and optional display columns clauses (basically, in annotation phase I add bogus nodes for the missing query or display column clauses). What remains are to finish fixing some runtime problems, related to how to display a browse with no query or no column list.

#9 - 10/22/2012 08:54 AM - Constantin Asofiei

- % Done changed from 60 to 80

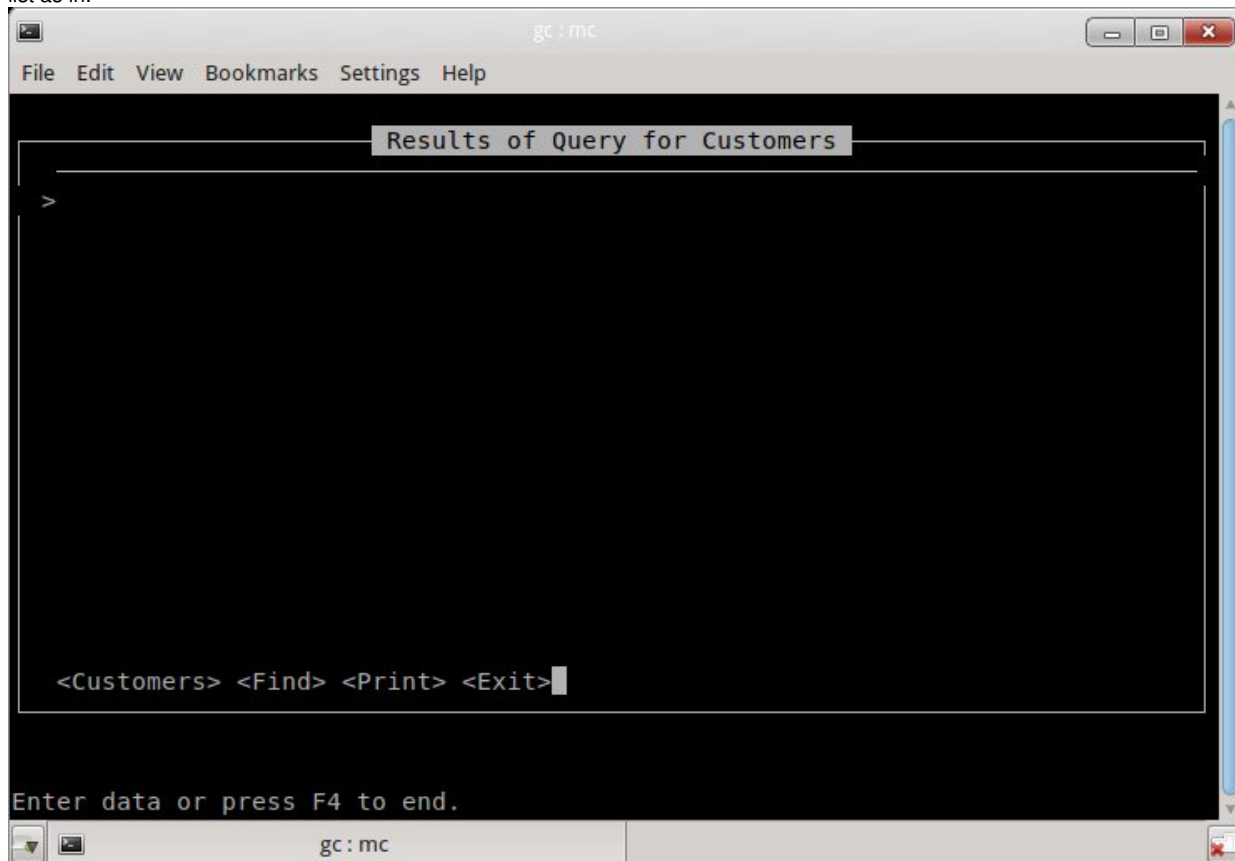
Finished cleaning up all files from the update, I'm reconverting using current staging and I'm going to do regression testing today.

#10 - 10/22/2012 10:35 AM - Constantin Asofiei

- File brws.jpeg added

Greg,

There is a remainder brws runtime issue which I think should be documented and solved later on. Currently, P2J displays a browse with no column list as in:



But, in 4GL, the browse is displayed with an empty content (only the border), with a not-sure-what character on the third column just below the top box line (the char resembles a "1" char, but is not this char, and I didn't find a way to capture the text from a 4GL window yet).

#11 - 10/22/2012 02:38 PM - Greg Shah

Please do add a new Bug issue along with a full testcase, screen shots and any other recreate/description of the problem.

But, in 4GL, the browse is displayed with an empty content (only the border), with a not-sure-what character on the third column just below the top box line (the char resembles a "1" char, but is not this char, and I didn't find a way to capture the text from a 4GL window yet).

I'm not sure what you mean by this. Please provide screen captures of both the P2J and 4GL versions in the bug report.

#12 - 10/24/2012 10:38 AM - Greg Shah

- File *ecf_upd20111207b.zip* added

I was reviewing the files that I posted and I think I was missing a major change made by Eric to StatsWorker.java in *ecf_upd20111207b.zip*. I am posting it now.

Eric: is the version I am posting here the final/best one?

#13 - 10/24/2012 11:38 AM - Eric Faulhaber

Greg Shah wrote:

Eric: is the version I am posting here the final/best one?

My changes in the StatsWorker version in *ecf_upd20111207b.zip* also are included in your version of StatsWorker in *ges_upd20111213a.zip*. So, with respect to my changes only, either version will work. The differences between the two versions are in your code (*setMasterReport*). You will have to compare to see which version is the right one, but I suspect it's the one in your 1213a update.

#14 - 10/24/2012 02:01 PM - Constantin Asofiei

Regression testing has passed. Can't move this to CVS until the [#1445](#) *frame_generator.xml* problem is solved (I'll attach the update here tomorrow).

#15 - 10/25/2012 03:43 AM - Constantin Asofiei

- File *ca_upd20121025a.zip* added

Added merged update, with all fixes.

#16 - 10/25/2012 10:51 AM - Greg Shah

In regard to the StatsWorker version in ecf_upd20111207b.zip, I think the differences with the version in ges_upd20111213a.zip have to do with allowing the MASTER_INDEX_CACHE file to exist between runs of report processing. The idea is that during report processing (while we are still figuring out the parsing, testing new reports...) we often have report failures that abort the report run. If we delete the cache file each time we fail, then at the end we can't even generate a full master index (because each partial run will only have a partial index file). My change to remove the deletion allowed me to get a master index on all reports even though I had to restart the reports multiple times.

Please retain this "feature" by making the cache deletion controlled by a boolean flag (call it partialRuns). Add a method to expose getter/setter for that flag. Make it default to false which means that it DOES delete the cache file each time. If I want to enable partialRuns later, I'll call something to set partialRuns = true.

#17 - 10/25/2012 10:53 AM - Greg Shah

I understand that we can't promote staging until the frame_generator.xml problem is fixed. But it seems like we can go ahead and check-in the update to CVS, unless the update itself increases the problems. As far as I understand it, the frame_generator.xml issue is from svl_upd20120528a.zip, right? This update you have been working on is fully safe?

#18 - 10/25/2012 11:21 AM - Constantin Asofiei

I understand that we can't promote staging until the frame_generator.xml problem is fixed. But it seems like we can go ahead and check-in the update to CVS, unless the update itself increases the problems. As far as I understand it, the frame_generator.xml issue is from svl_upd20120528a.zip, right? This update you have been working on is fully safe?

Looks like the frame_generator.xml was not a real issue after all; see [#1445](#) for details (I'm working on upgrading the baselines to include the correct behavior, as in staging).

#19 - 10/29/2012 04:33 AM - Constantin Asofiei

- File ca_upd20121029a.zip added

Please retain this "feature" by making the cache deletion controlled by a boolean flag (call it partialRuns). Add a method to expose getter/setter for that flag. Make it default to false which means that it DOES delete the cache file each time. If I want to enable partialRuns later, I'll call something to set partialRuns = true.

Greg, I've added these changes to the attached update.

#20 - 10/30/2012 09:48 AM - Constantin Asofiei

- Status changed from WIP to Review

#21 - 10/31/2012 01:18 PM - Greg Shah

- Target version set to Milestone 2

#22 - 11/01/2012 01:11 PM - Constantin Asofiei

- File *ca_upd20121029a.zip* added

I've added the update to CVS. Not applied to staging yet. Needs full P2J rebuild. Does not affect MAJIC's generated Java sources.

The attached version adds the merged *methods_attributes.rules* file.

Note: the update does not upgrade the H2 to the version from *ges_upd20111213a.zip*. Let me know if you want me to upgrade H2 too (this will need another regression cycle to be run).

#23 - 11/01/2012 02:47 PM - Greg Shah

- Status changed from Review to WIP

Yes, please upgrade the H2 version.

#24 - 11/02/2012 09:24 AM - Constantin Asofiei

Regression testing is running. If it passes, I will apply both the *ca_upd20121029a.zip* and the upgraded H2 to staging.

#25 - 11/02/2012 09:36 AM - Eric Faulhaber

Are you testing with the latest available H2 jar as of today? Or the newer one we grabbed back in December 2011 (i.e., from *ges_upd20111213a.zip*)?

#26 - 11/02/2012 09:38 AM - Constantin Asofiei

I'm testing with the latest available H2 jar as of today - Version 1.3.169 (2012-09-09).

#27 - 11/02/2012 09:40 AM - Eric Faulhaber

Good, thanks.

#28 - 11/02/2012 12:17 PM - Constantin Asofiei

- File *ca_upd20121102a.zip* added

Latest H2 has passed regression testing. *ca_upd20121029a.zip* and *ca_upd20121102a.zip* are applied to staging (P2J is rebuilt, no reconversion is needed as it does not affect MAJIC).

ca_upd20121102a.zip was not applied to CVS yet, I'll apply it on Monday.

#29 - 11/02/2012 01:15 PM - Greg Shah

Great! You can also distribute it to everyone at that time.

Based on the changes, please make any needed changes to our documentation. Let me know if there are any questions.

#30 - 11/05/2012 03:13 AM - Constantin Asofiei

Update ca_upd20121102a.zip is applied to CVS.

#31 - 11/08/2012 09:16 AM - Constantin Asofiei

Some questions/info about the reporting changes:

- master_report_from_cache.xml - non-destructive point-in-time master report index generation from a cache file: the StatsWorker.Library is missing the generateMasterReportFromCache method.
- should the Reporting chapter in Conversion Handbook be updated to mention the database support and the "non-destructive point-in-time master report index generation from a cache file" ?

#32 - 11/08/2012 10:17 AM - Greg Shah

Both of these can be left out for now. This is not a feature I expect to be included long term.

#33 - 11/09/2012 08:25 AM - Constantin Asofiei

- Status changed from WIP to Review

- % Done changed from 80 to 90

All changed docs are in bazaar (search for [CA] tag):
handbook: other_customization.odt running_conversion.odt
reference: control_flow.odt display.odt frames.odt preprocessor.odt queries.odt

#34 - 11/12/2012 11:14 AM - Greg Shah

- % Done changed from 90 to 100

- Status changed from Review to Closed

#35 - 11/16/2016 09:02 AM - Greg Shah

- Target version deleted (Milestone 2)

Files

ges_upd20111213a.zip	1.61 MB	10/11/2012	Eric Faulhaber
ecf_upd20111218a.zip	10.2 KB	10/11/2012	Eric Faulhaber
conversion_heavy_preprocessor.png	53.5 KB	10/11/2012	Eric Faulhaber
ca_upd20121018a.zip	268 KB	10/18/2012	Constantin Asofiei
brws.jpeg	34.4 KB	10/22/2012	Constantin Asofiei
ecf_upd20111207b.zip	27.6 KB	10/24/2012	Greg Shah
ca_upd20121025a.zip	582 KB	10/25/2012	Constantin Asofiei
ca_upd20121029a.zip	582 KB	10/29/2012	Constantin Asofiei
ca_upd20121029a.zip	582 KB	11/01/2012	Constantin Asofiei
ca_upd20121102a.zip	1.25 MB	11/02/2012	Constantin Asofiei