Jazz Team Server Monitor (JTSMon)
Release Notes 5.0.2

December 5, 2014
IBM Rational Performance Engineering Team: Dave Schlegel

Version 5.0.2

JTSMon: jazz.net RM webservice counter reports HTTP 401 Unauthorized (338908)

- **ISSUE:** In 5.0.2, the Requirements Management (RM) application now supports web service counter reports and works with JTSMon in some conditions. Problems with monitoring have been found have a few hours in which the connection times out and is not successfully reestablished.
    - **SOLUTION: (Still open issue)**

JTSMon: Support limited analysis data range option (326156)

- **ISSUE:** JTSMon normally analyzes all of the data recorded by monitoring although it may skip samples when the sampling rate is less than the monitoring rate. However sometimes the user wants to analyze a subset of the data, such as a specific day or week.
    - **SOLUTION:** Modify the analyzer to recognize an existing property ANALYSIS_DATA_RANGE to specify a comma separated range of data snapshot identifiers to identify a subset of the data to analyze. For example, if 100 samples are taken snapshot files for each server being monitored are named CounterContentServer1.html.zip to CounterContentServer100.html.zip. Adding this property to jm.properties

ANALYSIS_DATA_RANGE=10,50

will cause analysis to only read the files CounterContentServer10.html.zip to CounterContentServer50.html.zip.

JTSMon should show better breakdown of RQM system components (326237)

- **ISSUE:** JTSMon normally clips off the common prefix “com.ibm.team” and interprets the next service name element as the system “component”, i.e. the functional area that the service is part of. The RQM application uses “com.ibm.rqm” and so all webservices are presented as part of the “com” system component which is incorrect and unhelpful.
    - **SOLUTION:** Recognize com.ibm.rqm specially and replace it with “rqm_” so that RQM web services will be presented as members of a component name like rqm_planning, rqm_execution, etc. which provides more useful information.

JTSMon: Runtime error 1004 on creating workbook with Excel 2010 and above (326340)

- **ISSUE:** The JTSMon Visualizer is a legacy format “xls” file that can work with Excel 2003 and above. By default when it creates a new spreadsheet it is in the .xls format (limited to 255 columns) and so it is unable to manage larger analysis data sets even when the user has Excel 2010 or higher which natively support this and the Visualizer saves it as an .xlsx file which should handle larger data sets.
    - **SOLUTION:** Modified the Visualizer to save, close, and reopen the output spreadsheet in order for Excel to recognize the larger capacity data set.

Investigate how JTSMon can read VVC counters (333411)

- **ISSUE:** The web service counter report files may have new tables providing data on VVC services (Version(s) Variant(s) and Configuration(s) capabilities) which may be used to support configuration management capabilities for various Jazz applications at some future date.
  - **SOLUTION:** Provide support for analyzing and graphing the new VVC tables when they are present in the data. The three different services tables (VVC Configuration Cache Service, VVC Internal Configuration Cache Implementation Service, and VVC Internal Configuration Cache Service) are combined into a single analysis table looking at counts, average time, and total time.

Adapt JTSMon to tolerate scenario based counter reports (333962)

- **ISSUE:** The web service counter report files may have a new column in existing tables identifying a *scenario* (client side usage context label). Data may be reported under both individual scenarios as well as in summary. This format change breaks existing JTSMon data parsing, leading to data duplication
  - **SOLUTION:** Provide support for recognizing the scenario data column when it is present in the data. Ignore all scenarios but the summary scenario (“*”) providing the equivalent of existing pre-scenario data content.

Version 5.0

Provide JTSMon command line options to compress or decompress directories of counter service reports (305913)

- **ISSUE:** The new capability of JTSMon to write out web service counter reports in zip format saves a lot of space for storing new data and speeds up reading it back for analysis. But existing users may have a lot of historical data that could benefit from being compressed after its already been collected.
  - **SOLUTION:** Provide a new pair of JTSMon commands that compress or decompress existing files. The new “zip” command will compress any uncompressed webservice counter files found within a specified directory tree, converting *.html files into *.html.zip files that consume up to 98% less space. The “unzip “ command will decompress html.zip files back into uncompressed html files. Example:

```
java –jar JTSMon.jar zip sampleDirectory
```

Enable basic JTSMon monitor feature using REST calls to avoid version checks (304097)

- **ISSUE:** JTSMon sometimes fails to connect to a Jazz server with a ServerVersionCheckException. Depending on which API version JTSMon is linked with, a strict version check during login will not allow to access repositories running significantly older or newer product versions. This API login is needed only for repository reports which aren’t used very often, but isn’t needed for the web service counter reports that most users need.
  - **SOLUTION:** Add a new property (JTSMON_AVOID_VERSIONCHECK) that tells JTSMon to only log into the Team repository API when repository reports are needed. This will avoid version checks across major releases, facilitating continued use during new product development and allowing JTSMon to continue working with older Jazz product releases for web service counter reports. This is “false” by default so you need to set it to “true” to avoid the version check.

JTSMon Data Visualizer saves output in 2003 compatibility mode limiting allowed number of columns (298447)

- **ISSUE:** The Data Visualizer generates spreadsheets as .xls format files for maximum compatibility (MS Office 2003 and above) but that limits the number of columns to the restrictions imposed by Excel 2003 (no more than 255 columns).
  - **SOLUTION:** The Visualizer will now save output as .xlsx format when it is run in Excel 2007 or more recent versions to significantly increase the number of columns it can work with. If there is a compelling need to produce 2003 compatible output using more recent versions of Excel, the easiest solution is to edit the Visualizer macro code to remove the following line from the Init method in the macro:

```
xlsExt = ".xlsx"
```

JTSMon - ANALYSIS_SAMPLE_TIME property does not work correctly (298142)

- **ISSUE:** The ANALYSIS_SAMPLE_TIME property allows the user to analyze collected data at a different rate than the original data was sampled at, i.e. data might be collected at 10 minute intervals but an hourly analysis is desired so ANALYSIS_SAMPLE_TIME=60 will only look at data snapshots every hour, ignoring intermediate snapshots. There was a miscalculation in the code that tried to tolerate some variability in snapshot timestamps leading to errors.
    - **SOLUTION:** The error has been fixed. The new behavior adapts to how frequently snapshots are taken. If the original snapshots are taken at less than 5 minute intervals, the tolerance is set to 30 seconds. If snapshots are taken at longer intervals, the tolerance is set to 2 minutes.

Modify JTSMon to save counter service reports compressed and read back compressed files as needed (299067)

- **ISSUE:** Web service counter reports take up a lot of space (often 1 megabyte or more, especially for JTS servers). This is a tremendous waste of disk space for files that will only be read for data analysis, usually at the end of the run.
    - **SOLUTION:** Provide a new property that enables web service counter reports to be written out in ZIP format automatically (PARM_COUNTER_COMPRESS_DATA, enabled by default). Each report is written into its own .zip file and then decompressed on the fly during data analysis. Files now take up to 98% less disk space.

JTSMon default data directory is set incorrectly (294226)

- **ISSUE:** By default JTSMon was looking for analysis reference data files inside the original development source code structure instead of the JTSMon Data directory. The problem was that the mechanism used to flag user initiated property changes was incorrectly flagging default adjustments made for RTC as user changes. JTSMon then thought a user had changed the ANALYSIS_DATADIR property and it didn't override the default correctly.
    - **SOLUTION:** Disabled the flagging mechanism for any changes before starting to read the property file.

Version 4.0

Rebrand JazzMon as Jazz Team Server Monitor (274597)

- **ISSUE:** Rename JazzMon as "Jazz Team Server Monitor" (JTSMon) and realign version number to synchronize with Jazz product names and releases. JTSMon 4.0 is based on RTC 3.0 GA which is upward compatible with RTC 3.x and RTC 4.x releases. Support for RTC 2.x is being discontinued at this time.

JazzMon no longer works with V4.0.3 and Tomcat 70.0.40.0 (279639)

- **ISSUE:** Some users have reported the following problem connecting to TomcatV70.0.40.x based Jazz servers:

  ```
  Exception during Task(ServerInfoRestCall): SSLHandshakeException, Message:
  javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake failure
  ```

    - **SOLUTION:** This issue is to do with a change that happened in V7.0 of tomcat, and a bug that was fixed in V70.039.0 of tomcat. There are two ways to configure your Tomcat server to work around this issue:
    - Add this parameter to the server.xml of a Jazz V4.0.3 system using V70.0.40.0 of Tomcat:

      ```
      sslEnabledprotcol="TLSv1,SSLv3"
      ```

    - Add this property when you invoke JazzMon or JTSMon

      ```
      java -Dhttps.protocols="TLSv1,SSLv3" -jar JTSMon.jar monitor file=jm.jazzxxx.properties
      ```

JazzMon password encoding doesn't work with long passwords (281180)

- **ISSUE:** Some passwords encoded by the "password" command have failed at runtime due to decoding problems.
    - **SOLUTION:** The problems were fixed and tested to ensure passwords of up to 128 characters using any of the initial 128 ASCII characters were encoded and decoded successfully.
    - **NOTE:** Non-ASCII characters may work but are not fully supported at this time.

JazzMon should accept http URLs as well as https (261645)

- **ISSUE:** JTSMon supports abbreviated server URLs and applies some rules to figure out the full URL (e.g. “host” becomes https://host:9443/ccm). This functionality assumed that all server URLs would start with https:// and it added this in front of URLs not using SSL so http://host would erroneously become “https:http://host”, etc.
    - **SOLUTION:** The parsing code has been amended to properly recognize http:// and without changing it.
    - **SOLUTION:** A new optional prefix (<url>) may be used to disable any attempt at “fixing” the URL to avoid problems with patterns JTSMon has not seen before, i.e. “<url>http://myhost.com/app/service” will be left untouched.

Exception Error: Skipping folder that doesn't match expected name or contents pattern in JazzMon (279879)

- **ISSUE:** JTSMon was failing to recognize counter service data in directories if the ServerInfo.xml data was not present as expected, causing a crash when no data sets were recognized.
    - **SOLUTION:** An additional check was made looking for the first data file in case the ServerInfo.xml file is not found and a check for a null pointer during data output was added to prevent the crash.

JazzMon: Unexpected server response during monitoring causes analyzer to choke (247807)

- **ISSUE:** Users have reported intermittent issues during monitoring involving counter service reports containing errors that caused problems during analysis (https://jazz.net/forum/questions/86687/data-point-dropout-during-execution)
    - **SOLUTION:** JTSMon now checks for valid reports before saving them to file. When valid data is unavailable, it saves a copy of the previous report and sets a flag indicating there was an error with the sample and shows the error in the column of output tables using "(?)" to show the data was suspect.

RPP CompareRuns incorrectly computes interval rates and JazzMon AnalyzeData shows non-zero averages for unused counters in counter comparisons (275435)

- **ISSUE:** The JazzMon AnalyzeData function is used in computing baselines by comparing two counter service reports to generate an average for a specified period. When the counts were the same between the periods, it was incorrectly showing a non-zero average based on the ending value for the period, when it should show a zero since no activity took place during that period.
    - **SOLUTION:** JTSMon now shows a zero average for inactive service counters when making baselines.

Prior JazzMon releases

Version 1.4.0

Provide Excel macro workbook like NMON to automatically format and chart all the key data tables easily (232367)

- **ISSUE:** Creating charts by manually reading in CSV files and using the Chart Wizard is too time consuming
    - **SOLUTION:** Provide JazzMon_Visualizer.xls, an Excel macro file that produces a set of interlinked workbooks automatically charting nearly all available CSV files as standard charts. See Chapter 4 of the manual for more information.

Streamline JazzMon monitor-gather-analyze process (241716)

- **ISSUE:** Address user feedback to make JazzMon easier to use. The monitor-gather-analyze cycle is overly complicated and leads to confusion when users forget to gather their data before analysis and duplicates storage for large data folders. The JazzMon manual is comprehensive but imposing for new users. The jmTemplate.properties file is extensive but confusing for the basic cases most users need.
    - **SOLUTION:** Provide a new property ANALYSIS_IN_PLACE (true by default) to enable a simpler data analysis pathway. When enabled, the *monitor* command will automatically analyze the data when monitoring is complete, *gather* is unnecessary, and *analyze* and *baseline* know to use the original data monitoring folder for their operations.
    - **SOLUTION:** Provide a JazzMonQuickStart guide which reduces the instructions to a single page of simple steps to follow – install, configure, monitor (analyze), and visualize (use JazzMon Visualizer).
    - **SOLUTION:** Provide a minimal default jm.properties file that just requires server URLs and login information that summarizes default behavior that can be changed by referencing jmTemplate.properties.

JazzMon average of interval averages is inaccurate because it includes intervals with no activity to compute average (236829)

- **ISSUE:** JazzMon monitoring is unreliable in some customer environments because the HTTP connections it is using didn't have short duration timeouts set. This has lead to unusually long times between samples producing data analysis artifacts using the irregular sampling times.
    - **SOLUTION:** Provide a new property SERVER_HTTP_CONNECTION_TIMEOUT providing the number of seconds (default 120 seconds) to force monitor sample calls to timeout when there are communications problems.

JazzMon average/baseline computed wrong in service_etAvg (240969)

- **ISSUE:** The average of interval averages fixed in JazzMon 1.3.1 but the Average / Baseline computation was still using the original miscomputed value leading to incorrect ratio values. In addition, the service_etAvg table and others like it were including the Total Times and Total Counts values as literals twice – in the average computation and as separate data columns
    - **SOLUTION:** The CSV file generation process was changed to allow cross-references to the other cells by name instead of duplicating the values or computations.

Version 1.3.1

JazzMon average of interval averages is inaccurate because it includes intervals with no activity to compute average (236829)

- **ISSUE:** Computing the overall average response time by including all intervals produces an incorrect response time. Intervals with no activity have zero response times, so they reduce the average artificially.
    - **SOLUTION:** Compute a weighted average response time for the entire run by dividing the total elapsed time spent during the monitoring period by the total number of counts to produce a true average response time.

  - **NOTE:** The TotalTime and TotalCounts used for this computation are now included as additional columns in the *_etAvg data tables to allow their use as additional sorting columns. First sorting in descending order by TotalCounts and then hiding or filtering out the low frequency web services allows the analyst to ignore calls that may have high response times but that are not used often enough to contribute to significant performance problems.

Version 1.3.0

Reduce effort to import JazzMon data files into Excel and make data sortable (230720)

- **ISSUE:** Tab-separated .txt files require manual intervention to import into Excel or Lotus Symphony
  - **SOLUTION:** Convert JazzMon data output to comma separated CSV files that can automatically launch the preferred spreadsheet application or be directly read without using the Import Wizard
- **ISSUE:** Sorting data tables to find the most interesting data requires relying on the last data sample column, which is not reliable, or manually adding computed totals or averages column to provide reliable sort keys.
  - **SOLUTION:** Embed spreadsheet formulas in the last columns to provide Totals, Max, Averages and computed ratio of data Average to baseline Average to provide sorting keys to help highlight or filter data without manual interaction.
- **ISSUE:** Spreadsheet formulas are not completely compatible between Microsoft Excel and Lotus Symphony because of a conflict in commas or semicolons as parameter separators. This causes formula evaluation errors.
  - **SOLUTION:** Add a new property (ANALYSIS_TARGET) to select between Excel (default) and Symphony output patterns
- **ISSUE:** Secondary files for grand totals and all fields confuse users about which files to use
  - **SOLUTION:** Remove extraneous files for *Totals* and *allfields and *sorted (now easier using sort keys)

Compute interval averages rather than reporting long cumulative averages that hide performance extremes (233260)

- **ISSUE:** Average response times (service_etAvg, async_etAvg, etc) for time trend intervals were based on cumulative averages reported in the original server reports at the end of each interval. Extreme response times were averaged into cumulative averages for frequently used operations hiding them and reducing the value of the response time data.
  - **SOLUTION**: Compute interval averaged based on dividing the total time spent during the interval by the total count during that interval. This eliminates the confusing situation where the average time would remain constant even when no calls were made to the web service during the interval. The etAvg reports now show each interval's average instead of the cumulative average which allows you to see response time variations over the workday.

JazzMon defaults should be more in line with recommendations and FAQ (232365)

- **ISSUE:** The JazzMon FAQ (Frequently Asked Questions) recommended property settings at odds with the property default values.
  - **SOLUTION:** Adjust the defaults to be in line with the FAQ
    - SEQ_RUN_LENGTH_ARG default is now 7d (7 days)
    - PARM_COUNTER_RATE_MINS default is now 60 minutes
    - PARM_REPOREPORT_RATE_MINS default is now 480 (8 hours)

JazzMon baseline selection fails with server application names like jazz02 (232620)

- **ISSUE:** Baseline comparison data was selected based on an exact match to the application suffix, i.e. “jazz” or “ccm” but some servers use an extended suffix such as “jazz02” which doesn’t fit this pattern.
  - **SOLUTION**: Search for baseline matches if the server suffix *contains* one of the available baseline suffixes.

JazzMon formatting precision rounding off data in totals and component-based reports (234358)

- **ISSUE:** Output of elapsed time totals fields (i.e. “service_etTot.csv”) and component summary data for times and average response times (i.e. “servicecomponents_etAvg.csv”) were using formatting that rounded values to whole numbers when finer precision is important.
  - **SOLUTION:** Fix formatting to show three decimal places in line with other fields like average times.

Version 1.2.2

JazzMon passwords with special characters don't work (221027)

- **ISSUE:** Passwords that were multiples of 8 characters long with special characters were causing exceptions in the decoder
    - **SOLUTION:** Alter the decode password handler.

Version 1.2.1

Documentation changes only

- **New download location for both IBM field teams and customers:** https://jazz.net/wiki/bin/view/Main/JazzMon.
- **New support process:** Please ask support questions on the Jazz.net forums (https://jazz.net/forum) , using the tag "jazzmon".
- **New section 3.1.3:** Description of Web Service Components to provide more context for interpreting web service reports

Version 1.2.0

Assorted JazzMon fixes or improvements (212194)

- **ISSUE:** JazzMon table column headings use unconventional date format (mm_dd_yyyy, etc)
    - **SOLUTION:** Changed to use conventional format (mm/dd/yyyy hh::ss::ms)
- **ISSUE:** Excel chart wizard sometimes displays numbers for X-Axis instead of date headers
    - **SOLUTION:** Added Excel instructions to JazzMon manual for problems with X-axis showing numbers
- **ISSUE:** Password command sometimes produces encrypted passwords with special characters that have sometimes produced run-time errors
    - **SOLUTION:** Fixed password generation to try to avoid embedding special characters that cause runtime problems, displaying a warning if unable to avoid using +, -, / chars

JazzMon baselines not dividing to get per hour rate correctly (214365)

- **ISSUE**: Baseline data files are not adjusting counts or total times data values to get hourly rates
- **SOLUTION:** Fixed the data analysis file pattern matching to cover this situation.

Test and evaluate Jazzmon for general usability and other issues we can address before next release (214009)

- **ISSUE:** The password command output left some ambiguity of how to use the encrypted password as a property
    - **SOLUTION:** Improve password output by printing actual property line.
- **IMPROVEMENT:** Rename command line option so that "properties=<propfile>" is clearer as "file=<propFile>". The original option is still recognized to minimize confusion on existing users.

Convert Std Deviation code to use the math.commons 1.2 apache library. (136112)

- **ISSUE:** Internal calculation of running standard deviation calculation using textbook algorithms
- **SOLUTION:** Change to use existing Apache library to avoid legal distribution and provenance issues.

AnalyzeServerMonitor crashes working with unchanged (zero) QM async data (205270)

- **ISSUE:** Empty data from a non-functioning application or server causes crashes during analysis
- **SOLUTION:** Avoid crashing on empty data sets for whatever reason.

Add Jazzmon support for new Distributed ObjectGrid Cache table in webservice reports (207938)

- **ISSUE:** 4.0 Clustering support adds new table of cross-node synchronization that appears in two configurations based on cluster.
- **SOLUTION:** When new property ANALYSIS_CLUSTER is enabled, generate new object grid trend and totals output from Distributed Object Grid data in counter service report. Check the webservice data reports to determine cluster configuration.
- **NOTE:** Output file names were also standardized so that web service trend_* tables were renamed to service_*

JazzMon doesn't parse mixed of commas and periods correctly. (208156)

- **ISSUE:** Internationalization issue parsing data using period as thousands separator and comma as decimal point isn't parsed correctly in US locale. Should only occur when analyzing in different locale than server
- **SOLUTION:** Set new property ANALYSIS_EURO_LOCALE to true to reverse incoming period and commas

Version 1.1.0

JazzMon package needs version identification and release notes (191405)

- **ISSUE:** Hard to distinguish original release from updated versions
- **SOLUTION:** Add version number to zip package for distribution and provide new version command line option to display current version and compatibility
- Include JazzMon_ReleaseNotes.doc to document changes and defects addressed in each new release

JazzMon: Compute floating license trends when possible (197798)

- **FEATURE:** Floating license usage information is presented in a new trend table (licence_flVal.txt) in the JTS application directory. This provides an indication of active users but is subject to some caveats.
- License information is not yet aggregated or totaled across servers.

AnalyzeServerMonitor should total up the async tasks as it currently does for web services (199831)

- **FEATURE:** Compute totals for async total time and counts across multiple servers or nodes. This can serve as a guide to multiple servers or applications to identify site wide time or traffic.

Data collection glitches cause gaps in analysis (190512)

- **ISSUE:** JazzMon monitoring keeps going even if it fails to get occasional snapshots because the server is down or inaccessible but wasn't saving any data to indicate the gaps in reliable data. This could lead to confusion later on when the timeline gets skewed when compared to other servers or leads to unexplained spikes or dips in the output.
- **SOLUTION:** JazzMon will now write out copies of the last reliable data to act as place holders and record the errors matching those samples. Analysis output for the place holder time samples is highlighted with a leading marker on the column name "(?)". Documented more fully in updates to main documentation

Integrate server monitoring aggregation feature into standard processing (190884)

- **ISSUE:** Counter service reports from main proxy entry point for cluster get individual node reports randomly.
- **SOLUTION:** Monitor cluster by monitoring individual nodes then enable aggregation to get an accurate site-wide report. Documented in updates to main documentation
- Includes 2 new properties documented in jmTemplate.properties and main documentation
  - ANALYSIS_AGGREGATE_LIST
  - ANALYSIS_AGGREGATE_ZERO_BASIS

Getting password handling exception using Jazz monitor (3.0 bundle version) (197965)

- **ISSUE:** Passwords with length divisible by 8 were incorrectly handled as if it were encrypted