

OpenJDK Scorecard

This scorecard helps assess goals set by the OpenJDK Community and Governing Board so we can track progress over time. The Community Scorecard covers the areas of Infrastructure, Governance and IP. The Project Scorecard, focuses on Project-specific goals, and covers Visibility and Technical matters.

OpenJDK Scorecard Survey Results

Survey ran for a week from September 19-25. Responses on the survey were gathered, and a first round of scores proposed based on the Scorecard draft. Over 80 people responded to the survey.

- Less than half completed the entire survey (not unexpected, the Scorecard is detailed, as intended).
- There was a pretty even distribution amongst Roles and Experience level in the Survey.
- 34% of respondents were not Contributors OpenJDK, just “Interested Participants”. 12% Contribute on their own time. 17% Contribute as part of their job, but at less than half their time, and 37% of respondents Contribute full time as their job.
- Key Takeaways from Survey:
 - Scoring system (0-5 with comments) is well received with three suggested changes:
 - Clarify “Minor” versus “Major” inconveniences
 - Change description for “5” to be “Meets **or exceeds** expectations”
 - Add an NA/Don’t Know to future online surveys
 - There was an almost perfect correlation to areas I believe we are strong or weak and what the survey respondents believe.
 - Suggestion to include a “Learn More” link in some of the goals to help people understand background.
- Some “Thank You” comments were received (a) for doing the survey itself and (b) for supporting OpenJDK in general.

OpenJDK Scorecard Scoring Methodology:

Score	Meaning
0	Not Existent, no workarounds
1	Workarounds exist, but are highly problematic.
2	OK, with major inconveniences. (Only specific people are able to work around the issue, or specialized tools not widely available are required, or an inordinate amount of time is required).
3	OK, with minor inconveniences. (Any participant could work around the issue given available information/tools within a reasonable amount of time).
4	Good, but improvements sought.
5	Meets or exceeds expectations.

Comments field will be used to justify the Score, and describe how to improve upon the score for the next round (if not clear by improving the issues noted in the comment).

Community Scorecard

Proposed Score: Suggested score, proposed to the GB based on feedback from Oracle, and analysis of survey results. Note that it would be imprudent to only consider the average of the survey results too seriously given the spread of opinions, and those who might skew the survey up or down with ill considered clicks of 5's or 0's. Therefore, we look carefully at the median score, and the most popular score in the results as well.

Survey results X,Y,Z: X is the average weighted score, and Y is the median, Z is the most popular score. So "3.97, 4, 5" means the average score was 3.97, that the median was 4, but that 5 was selected more than any other

Infrastructure	Proposed Score	Survey Results	Comments
Project-Specific Infrastructure			
Project source code and history are easily accessible.	5	3.97, 4, 5	Projects are able to commit and push code, public is able to read and consume code. Code archives are maintained. Mercurial is generally well liked, and even detractors generally admit it's tolerable. It can be a challenge to find the right Mercurial forests without asking.
Projects are able to manage their SCM easily.	3	3.68, 4, 4	Projects are able to create new forests, add new Committers and manage repos, but usually require interacting with ops@openjdk.java.net at Oracle to perform these tasks. Lead times to make changes and have requests processed can vary significantly.
An issue tracker is available to the entire community.	1	1.66, 1, 1	An issue tracker is available. The public is able to submit issues, and read most issues. At the current time, only Oracle staff are able to perform queries, manage bugs and the bug lifecycle. This has required projects lead by non-Oracle Committers to seek alternative locations to manage bug fixes.

Public Code Review System is available to the entire community.	2	2.16, 2, 1	cr.openjdk.java.net used for code display, approval handled by email. More features would be possible if a newer technology could be used.
Group and Project			
Mailing List infrastructure is available.	4	4.34, 5, 5	Easy to use and generally well known mail tools available. Archives are not easily searchable, and attachments can be problematic. It would be helpful to better describe and group the various lists for people new to OpenJDK.
Project and Group members can easily edit web pages.	2	2.6, 3, 4	A process exists for Groups and Projects to have a web page created. Unfortunately, only Oracle staff may currently edit OpenJDK web pages, so non-Oracle staff must make requests to ops@openjdk.java.net. This is problematic for Groups and Projects that may not have Oracle Committers.
Project and Group members can easily edit wiki pages.	3	2.77, 3, 4	A wiki is available, but under different TOU than the rest of OpenJDK, which is of concern to some participants.
Blog Aggregator is available.	4	2.93, 4, 5	A blog aggregator is in use. Technology used and process for adding or removing bloggers could be improved.
Infrastructure to manage voting is available.	4	2.76, 3, 4	Voting occurs via email, which is a popular method in a number of communities. One issue that rises frequently is ineligible people tend to vote (not realizing they are ineligible), which can cause some confusion when tabulating results.
It's easy for a newcomer to determine the vitality of a particular Group or Project.	3	2.27, 2, 3	It is generally possible by monitoring mailing lists and other project activity – or simply asking a Project or Group lead on the relevant mailing list. However, there isn't a consistent manner across groups and projects to determine liveness. Having a better issue tracking system, and better enabling Projects to update their web assets (webpage, wiki, etc) will help in this area.

Governance	Proposed Score	Survey Results	Comments
Groups			
Votes are transparent.	5	4.46, 5, 5	As per the bylaws, elections have been run transparently. If anything were to be improved, a summary of votes and results could be posted somewhere other than email archives.
Quarterly Reports are published.	0	1.84, 2, 0	To date, quarterly reports from Groups have not been published, with rare exceptions (Build group, for example).
Governing Board			
Elected Seats are filled as per bylaws.	5	4.23, 5, 5	Elections have taken place on schedule, and candidates have been nominated. Moreover, there are Observers, and the process for adding observers is working as expected.
Meetings occur with regular frequency as per bylaws.	5	2.86, 3, 5	The board meets more frequently, but at least once per quarter as required.
Governing Board meeting results are transparent.	3	3.04, 4, 5	Minutes are being published, but there are frequently delays of several months between the meetings and the minutes being posted.
Open Meetings, or other venues for timely community discussions, besides email, are happening.	2	2.56, 4, 0	To date, no open meetings of the Governing Board have been held. However, the appointed and elected board are easy to find and approachable. There are several general "OpenJDK BOF" events throughout the year with different OpenJDK Governing Board Members participating where the public could attend and provide feedback.
Annual Review is completed.	0	2.11, 2, 0	It has only just been one year since the ratification of the OpenJDK Bylaws. The Governing Board is currently surveying the landscape and expects to do an annual review after JavaOne 2012.

Meritocracy - Progression of Roles			
Each Role is attracting new Participants.	3	2.96, 3, 4	<p>In this case, we use a score of “2012 == 3”, and will adjust this score up, or down, over time as we believe growth is increasing or decreasing.</p> <p>Given the current infrastructure and resources, we feel there is a good number of new Participants and Projects joining the ecosystem, and that individuals are free to evolve their Roles. However, if successful, we should see increasing activity from the current baseline.</p>
New Participants can easily see what Roles and progressions are available.	3	2.90, 3, 4	Aside from the Bylaws, limited resources are available explaining the various roles and progressions.
Participants are progressing to the appropriate Roles.	3	2.60, 2, 2	Given the limited visibility into what the various Roles are, feedback for this goal is mixed. As with the initial goal in the subsection, we will use “2012 == 3” as a baseline, and adjust up or down based on progress of people being able to transition roles YoY.
Votes			
Voting Process is clear and well understood.	4	4.15, 4, 4	Voting processes are described in the Bylaws and it’s easy to get quick answers to voting related questions. An FAQ or similar resources would be all that’s required to improve this area.
Public Voting on issues with results published, as per the bylaws.	5	3.97, 5, 5	Votes are happening in public, and transparently, as required. A posted summary of vote results could be helpful to avoid needing to search mail archives for results.

Intellectual Property	Proposed Score	Survey Results	Comments
Licenses			
All projects use FSF or OSI License.	5	3.96, 5, 5	Working as expected. Some concerns were raised that it's possible for Oracle and related Licensees to distribute commercial distributions based on OpenJDK sources. However, that is intended. It is consistent with other communities, and OS licenses that commercially licensed software be possible.
Trademark License exists and is easily accessible.	5	3.75, 4, 5	OpenJDK Trademark license was recently updated. Some concerns were raised that Oracle sole steward of the OpenJDK trademark, however, that is intended.
Terms of Use for Infrastructure are consistent.	3	3.55, 4, 5	Progress has been made in this area, but some preferred technologies (such as the Wiki) are still not consistent with OpenJDK.
Policies for legal notices in source code are documented.	1	2.95, 4, 5	Policies exist, but are not documented and require asking around.
Contributor Agreement			
Path for new Contributors to complete Contributor Agreement is clear and working.	3	3.52, 4, 4	The Contributor Agreement process is well described and relatively straight forward. However, processing times can vary and may take up to a month, unless escalated by a Project Committer.
It is easy to identify existing Contributors (people covered by a Contributor Agreement).	4	3.21, 3, 3	Contributors are listed on a publicly available on Signatories List page. Project and Group Role affiliations are listed on the OpenJDK Census page. However, it is not always clear to Committers where this list is, and how to use it. Also, the information can be sparse and requires follow up and clarification.

Ability for Contributors to handle Change in Employment is clear and working.	4	3.10, 3, 4	Notwithstanding the issue about processing time, it is easy for Individuals to continue their Role in OpenJDK should they wish to do so, and their employer changes. However, this fact is not readily clear and there may be confusion by Committers when it happens.
IP Processes			
Process for incorporating third-party libraries is clear.	1	2.09, 1, 1	Currently, there is no process or method for Projects to add third-party libraries to their code. When third party dependencies must be added, only Oracle staff are able to do so, and the process for them doing so is completely internal.
Process for asking IP-Related questions is clear.	1	1.91, 1, 1	Currently these discussions can only take place on relevant project or discuss mailing lists. No guidelines or specific venues exists.

JDK 7u & JDK 8 Project Scorecards

Proposed Score: Suggested score, proposed to the GB based on feedback from Oracle, and analysis of survey results. Note that it would be imprudent to only consider the average of the survey results too seriously given the spread of opinions, and those who might skew the survey up or down with random clicks of 5's or 0's. Therefore, we look carefully at the median score, and the most popular score in the results as well.

Survey results X,Y,Z: X is the average weighted score, and Y is the median, Z is the most popular score. So "3.97, 4, 5" means the average score was 3.97, that the median was 4, but that 5 was selected more than any other

*** NOTE – Survey results for JDK 7u include only 3 completed scorecards so was not a big factor in determining proposed scores. The JDK 8 project had 15 completed scorecards, and feedback was more relevant.

Per-Project Scorecard	Proposed Score (JDK 7u / JDK 8)	Survey Results (JDK 7u / JDK 8)	Notes
Visibility			
JDK Enhancement Process (JEP) is helping identify interesting features.	2 / 3	1.33, 1, 1 / 3.00, 3, 3	The JEP process is in place. We seek ways to improve quality of feedback, status of particular JEPs, and to respond faster to proposed JEPs.
Project Planning is publicly available and observable.	2 / 3	1.33, 1, 1 / 2.67, 3, 3	At this time, people who actively watch mailing lists are usually able find and track progress and compare to schedules. However, there is generally no single-source summary location for this information, or agreed time interval for reporting.
Design Decisions are publicly available and observable.	2 / 3	1.67, 1, 1 / 3.2, 3, 3	Design decisions are often spread across various resources, and not always publicly visible. In some cases, JEPs and feedback on JEPs, besides being able to observe JSR Expert Groups helps somewhat with visibility.
Projects are providing information on their roadmaps, milestones, build, integration, and release schedules.	2 / 3	3.00, 2, 2 / 2.90, 3, 3	There have been improvements in JDK 8 with publishing milestone, release and roadmap schedules but more improvements are needed.

Relevant documentation is available and up to date.	4 / 4	2.67, 3, 3 / 2.80, 3, 4	Documentation is available, but goes stale, more attention is need to keep things current.
Identifying Project Leadership and determining how to ask questions is easy.	3 / 3	3.33, 3, 3 / 3.70, 4, 4	The OpenJDK Census page helps with this goal, but is not well known or understood. A different view into the Census (listing Projects, Groups and who fills the various Roles) may help improve this score.
Votes are transparent as per the bylaws.	5 / 5	3.33, 4, 4 / 3.80, 4, 4	Votes are happening transparently according to the bylaws. A summary of vote results may save people from having to mine mailing lists for results.
Technical Matters			
Project is easy to build.	2 / 3	3.33, 3, 3 / 3.18, 3, 3	Several community participants have demonstrated it's possible to do custom builds of OpenJDK. However, the knowledge and resources are scattered and difficult to find. Related tools for doing continuous build and integration testing do not exist. The Infrastructure Project (sometimes referred to as the "new build" Project) has offered improvements, but still has a way to go to help a broader set of Participants.
Project is easy to test.	2 / 3	3.67, 4, 4 / 2.5, 3, 3	There are still internal tests and frameworks. This can cause issues and delays with some projects when 3rd party contributions must first be verified before going into a mainline release. Some spec code requires additional licenses – their availability and ease of access is scored separately.

Contributing new test cases is easy.	2 / 2	3.00, 4, 4 / 2.8, 3, 3	It is currently possible to contribute tests, but only to a subset of the overall testing that is required to ensure a stable build. Also, there are no guidelines for submitting tests, and it is currently very difficult for the Contributors to participate in this area.
Submitting a patch is easy.	3 / 3	2.67, 4, 4 / 3.55, 4, 4	Submitting a patch can be fairly straightforward to Participants, but can be daunting to people new to OpenJDK. A guide, or other documentation, with an eye to new Participants / Contributors, would be helpful.
Making a complete fork of the project is easy.	3 / 3	3.00, 4, 4 / 3.00, 3, 3	It has been demonstrated that it is technically easy to migrate code from OpenJDK to various forges. The inconvenience remaining is forkers are on their own to define what “build and test” means for the fork. Better build and test systems will help fork-ability.
If applicable, API Specification is available and easy to find.	4 / 4	3.33, 4, 4 / 3.78, 4, 4	The Javadoc for API for JDK 7 and JDK 8 are straightforward and easy to find. What is still a challenge, though, are finding information about things like Command line switches, that “shall-not-break” during updates.
Release and Update Projects (only)			
Process for getting TCK Access is clear.	4 / 0	2.67, 3, 3 / Not Applicable	It is possible to receive the Java Compatibility Kit by completing the “OpenJDK Community TCK License Agreement” (OCTLA). Processing times may vary. The OCTLA for JDK 8 is not yet available.
Finding list of OCTLA Signatories is easy.	2 / 0	2.00, 3, 3 / Not Applicable	A list of OCTLA Signatories exists, however, it is currently out of date and difficult to find. The OCTLA for JDK 8 is not yet available.