

## Shashank Tiwari's Position Statement on the JCP Executive Committee (EC)

Election for JCP EC members is in progress and by next week a new committee will be in place. So it's an important time to understand where the JCP stands and what the EC members can possibly do to make things better for the JCP and the Java community.

Simply stated, JCP is a member driven organization to create standards for the Java language and the platform. Ideally, it intends to be the common aggregation point for all the voices in the community. Unfortunately though, it's still far away from realizing this dream. There are over 10 million programmers and thousands of companies that actively use Java to create their products and deliver their services. However, there are less than 1500 JCP members as it stands today. In addition, only a handful of these 1500 are active in proposing JSR(s), participating in Expert Groups or providing active feedback on the specifications. Therefore, JCP hardly represents a majority of the community.

Why is it important that a majority of the community take active part in the JCP? Standards make sense only when they are adopted by a large majority. In the case of Java it means,

- companies that make Java tools and products need to make their products and offerings comply with the standards and
- developers and service providers who use Java in creating applications need to adopt and accept it.

The current gap is evident from the fact that many JCP created standards are hardly in use. For example JSR 69 (Java OLAP Interface), which was approved back in June 2004, never had a "final release" and is hardly supported by the OLAP vendors or developers today. There may be a small group still using it but alternative standards have rendered it useless from the time it was still being created.

Things are improving though! Over the last few months we have seen an increased participation from all corners. This is making specifications more relevant and meaningful. However, it's not enough yet and a lot more participation from the community is required.

Apart from less participation, the JCP process has additional shortcomings, which are as follows:

- Too many JSR(s) have not reached completion and remain in limbo for over 3 years now. These JSR(s) either need to be taken to completion or officially abandoned. In some cases it may make sense to start new JSR(s) to address the needs that the earlier JSR may have decided to address.
- Many JSR(s) try and solve the same problem. It makes sense to merge a few specifications where they overlap. It may also make sense to refactor related JSR(s) sometimes.

- Many JSR(s) need to be drastically simplified. Enterprise Java is complicated further will the addition of complicated JSR(s). The work of simplifying the EJB specification is a good success story to emulate.
- Many JSR(s) need to be stalled. Sometimes the standards body has proactively tried to create standards in an area, which is still volatile and therefore has not seen much success. No point starting too early. Standards are not about “early adopters” or “bleeding edge technology”.

Now that we are aware of some of areas to work on, let’s see where the EC could potentially contribute. Firstly, the role of the EC is not to manage each of the JSR(s). The JSR spec lead and expert group members run and manage the affairs of their respective JSR. The EC’s role is to manage the JCP process itself. Its duty is to provide checks and balances by voting for or against proposed JSR(s) and facilitate the workings of the JCP.

Therefore, the most significant contribution the EC can make can be summarized as follows:

- Evangelism – Encourage all members of the JCP to participate and even take on the additional responsibility of spreading the word to the larger community. Also work on standards adoption once they are established.
- Process Democratization – Work to mitigate the imbalance between corporate influence and individual members. Allow a few newer processes – for example, allow change of spec lead (not necessarily from the same company) mid-term if required, especially if things are currently going awry in that specific JSR.
- Active Collaboration – Propose and support collaboration among related JSR(s). Work on refactoring existing JSR(s) where required.
- Future Direction – Actively identify areas where standardization will help Java take a smooth course into the future and encourage participation from members working in such areas. For example JSR 292 (Supporting Dynamically Typed Languages on the Java Platform) is a good initiative to facilitate the evolution of Java.

By now you have a sense of what I am thinking. Much needs to be said and done, but I will stop here and start doing my bit to make JCP a more effective organization.

Let me start by appealing to all of you to come take part in the JCP. For those who are not yet its members, please come join in and make your mark. For those who are already members, please cast your vote ([https://www.jcpelection2008.org/jcp/election\\_ballot](https://www.jcpelection2008.org/jcp/election_ballot)) and make your presence felt.