

SIGPLAN and SIGSOFT

SIGPLAN and SIGSOFT have not formally adopted any replication process. In practice, however, numerous conference under the umbrella of these two SIGs (as well as conferences in the same areas but not affiliated with ACM) have informally adopted a fairly uniform process that goes by the name **Artifact Evaluation** (AE). Because this process has been described in detail elsewhere ([AECSite](#), [KVCACM](#)), we provide a brief summary of it here.

The AE process has a few salient highlights:

- The Artifact Evaluation Committee (AEC) is separate from the Program Committee (PC). In practice, especially by design, there may be a small overlap in members. However, each is constituted by different (cooperating) chairs for different purposes: evaluating papers versus evaluating artifacts.
- The AE process begins only after the PC has made its decisions. This firewall ensures that the AE process can have no impact—in particular, no detrimental impact—on a paper’s decision. This was initially put in place to encourage artifact submission, but has largely come to be seen as a positive design decision by the community. As a result, a paper may be accepted but may fail to pass muster with the AEC. What happens when a paper’s artifact is found by the AEC to be in egregious violation has intentionally been left unspecified; it is left to the PC chair and the authors to arrive at a reasonable decision.
- The AE process results in a badge that authors can put on papers that pass muster. For this reason, the AEC must make its decisions

at least a few days before the camera-ready deadline.

- The AEC is predominantly staffed by junior members of the community (mostly graduate students). There are many reasons for this, as described in the longer articles. The two most important are: (a) they are often the ones most familiar with the tools used for artifacts, and (b) it helps shape the value system of future community leaders.
- The AE process does not dictate what can constitute an “artifact”. More often than not, especially in SIGPLAN and SIGSOFT conferences, these will be software tools. However, they could also be data sets, formal models, proofs, architecture diagrams, and so on. The term “artifact” was specifically chosen to be agnostic to the nature of the artifact.
- The most important detail of the AEC process is the evaluation criterion. The AEC does not evaluate the submitted artifact in isolation. Rather, it evaluates the artifact *relative to the expectations set in the paper* (to do which, the AEC is given a copy of the accepted paper). The paper was presumably accepted based on whatever expectations it set, so AEC members merely check that the provided artifacts satisfy those expectations rather than judging it by (presumed) absolute standards. While this does mean a paper with a “weak” artifact can get a stamp of approval, this is not problematic so long as readers remember the evaluation criterion.

Systems Used

The AE process does not mandate any particular systems. Indeed, systems are intentionally left unspecified to accommodate a wide variety of possible artifacts. In practice, as of this writing (2016), virtual machine

images have become increasingly popular ways of bundling software that can be run relatively painlessly.

Process Followed

The AEC Chairs convene an AEC independent of and parallel to the PC. The Chairs typically seek out senior graduate students and post-docs: people who have had experience with publication and who have produced some artifacts of note themselves. Submissions to the AEC occur roughly a week after paper acceptance.

An AEC is structured rather like a PC, with two to three reviewers reviewing each artifact, writing detailed comments, and discussing their views. There may even be a brief rebuttal stage, but this depends on the amount of time available between paper decisions and camera-ready.

AECs have found it very productive to undergo a “kick the tires” round, which corresponds roughly to making sure the submitted artifact can be unpacked and, at a rudimentary level, “run” (e.g., the virtual machine login works and the program can be started; the Makefile successfully builds; etc.). The purpose of this round is not to be punitive: failures do not result in automatic rejection. Rather, this catches minor errors (e.g., missing virtual machine password; hidden filesystem dependency); authors are given a chance to submit a revised artifact.

Changes Made

Not applicable.

Other Comments

None.