Contribution ID: **297** Type: **Poster**

Pythonization API for Cppyy

Thursday 13 October 2016 16:30 (15 minutes)

Cppyy provides fully automatic Python/C++ language bindings and so doing covers a vast number of use cases. Use of conventions and known common patterns in C++ (such as smart pointers, STL iterators, etc.) allow us to make these C++ constructs more "pythonistic." We call these treatments "pythonizations", as the strictly bound C++ code is turned into bound code that has a Python "feel." However, there are always a few corner cases that can be improved with manual intervention. Historically, this was done with helpers or wrapper code on the C++ or Python side.
In this paper, we present the new pythonization API that standardizes these manual tasks, covering the common use cases and in so doing improving scalability and interoperability. This API has been provided for both CPython and PyPy. We describe the fundamental abstractions that it covers, how it can be used to resolve conflicts across packages, and its performance.

Tertiary Keyword (Optional)

Secondary Keyword (Optional)

Primary Keyword (Mandatory)

Analysis tools and techniques

Primary authors: Mr ST CLERE SMITHE, Toby (Warwick University); LAVRIJSEN, Wim (Lawrence Berkeley National Lab. (US))

Presenter: LAVRIJSEN, Wim (Lawrence Berkeley National Lab. (US))

Session Classification: Posters B / Break

Track Classification: Track 5: Software Development