# Article information:

OMPT: An OpenMP Tools Application Programming Interface for Performance Analysis | SpringerLink  
<https://link.springer.com/chapter/10.1007/978-3-642-40698-0_13>

# Article summary:

1. The OpenMP Language Committee has developed OMPT, an application programming interface (API) for performance tools in OpenMP.

2. OMPT allows performance tools to gather performance information from applications with low overhead and map it back to a user-level view.

3. OMPT provides runtime state tracking, callbacks and inquiry functions, and additional callback notifications to enable more comprehensive monitoring capabilities.

# Article rating:

Appears moderately imbalanced: The article provides some useful information, but is missing several important points or pieces of evidence that would be required to present the discussed topics in a balanced and reliable way. You are encouraged to seek a more balanced perspective on the presented issues by exploring the provided research topics and looking at different information sources.

# Article analysis:

The article titled "OMPT: An OpenMP Tools Application Programming Interface for Performance Analysis" provides an overview of the OMPT (OpenMP Tools) API, which aims to address the lack of a performance tools API in the OpenMP standard. The authors discuss the capabilities of OMPT and its potential benefits for performance analysis.

One potential bias in this article is that it is written by the developers of OMPT themselves. This could lead to a biased presentation of the tool's capabilities and potential benefits, as they have a vested interest in promoting its adoption. It would have been beneficial to include perspectives from independent researchers or users who have evaluated or used OMPT in practice.

The article does not provide much evidence or examples to support its claims about the usefulness and effectiveness of OMPT. While it mentions that OMPT enables performance tools to gather useful information with low overhead, it does not provide any concrete data or benchmarks to demonstrate this claim. Including such evidence would have strengthened the argument for adopting OMPT.

Additionally, the article does not explore any potential drawbacks or limitations of using OMPT. It would have been valuable to discuss any possible risks or challenges associated with implementing and using OMPT, as well as any trade-offs that users might need to consider.

Furthermore, the article focuses primarily on the benefits of OMPT without adequately addressing alternative approaches or existing tools for performance analysis in OpenMP programs. It would have been helpful to compare and contrast OMPT with other available options, highlighting its unique advantages and disadvantages.

Overall, while this article provides an introduction to OMPT and highlights its potential benefits, it lacks depth and critical analysis. It would benefit from including more evidence, exploring counterarguments, addressing potential limitations, and providing a more balanced perspective on the topic.

# Topics for further research:

* Comparison of OMPT with other performance analysis tools in OpenMP
* Limitations and challenges of implementing and using OMPT
* Independent evaluations of OMPT by researchers or users
* Benchmarks and performance data demonstrating the effectiveness of OMPT
* Trade-offs and considerations when adopting OMPT for performance analysis
* Critiques or alternative perspectives on the benefits of OMPT

# Report location:

<https://www.fullpicture.app/item/35a4a2a1f17daaffbde4f45cd9c727f2>