

Ji-dong ZHAI, Wen-guang CHEN. 2018. A vision of post-exascale programming.
Frontiers of Information Technology & Electronic Engineering, 19(10):1261-1266.
<https://doi.org/10.1631/FITEE.1800442>

A vision of post-exascale programming

Key words: Computing model; Fault-tolerance; Heterogeneous;
Parallelism; Post-exascale

Corresponding author: Ji-dong ZHAI

E-mail: zhaijidong@tsinghua.edu.cn

 ORCID: Ji-dong ZHAI, <http://orcid.org/0000-0002-7656-6428>

Challenges

Three main challenges faced in future post-exascale programming models:

- (1) Supercomputers which are becoming more **heterogeneous**;
- (2) A large amount of **parallelism** in applications which needs to be exploited to fully utilize a large-scale supercomputer efficiently;
- (3) Fault tolerance programming which** will be inevitable in a future post-exascale system.

Heterogeneous

1. One practical and effective approach is to develop a programming model containing both interfaces for complex features called '**low-level interfaces**' and directives for essential features called '**high-level interfaces**':

(1) The low-level interfaces express **enough architecture-specific features**.

(2) The high-level interfaces comprise **a set of directives**.

2. The last requirement for the programming model is that it should have **a set of powerful libraries** for special functions that can run on heterogeneous processors.

Parallelism

1. For a post-exascale system, the emerging **domain-specific language (DSL)** is a potential good solution to address this challenge.
2. On a post-exascale system, we may provide programmers with various DSLs for each specific domain, thus they can write programs with less efforts and higher performance.
3. A **performance-first programming model** is always necessary for expert HPC users and high-level framework developers.

Fault tolerance

1. Hardware errors will become more frequent in post-exascale systems. On a post-exascale system, novel techniques are required to efficiently detect faults.

(1) A post-exascale programming model:

- not only provide the **application programming interfaces** for functionality
- but also provide some APIs for developers to **understand their applications' performance**.

(2) Fault tolerance programming on a post-exascale system is more complex than before. A **co-design** from the underlying system to the programming framework and to the final application is needed.

Conclusions

1. In this study we discuss three significant challenges of programming for a post-exascale system: **heterogeneous, parallelism, and fault tolerance**.
2. The solutions for a post-exascale system must be aware of the large scale and be specific to **architectural features**.
3. Considering the complexity and diversity of underlying architectures, we infer that **programming frameworks** and **domain-specific languages** will be necessary for future post-exascale programming.