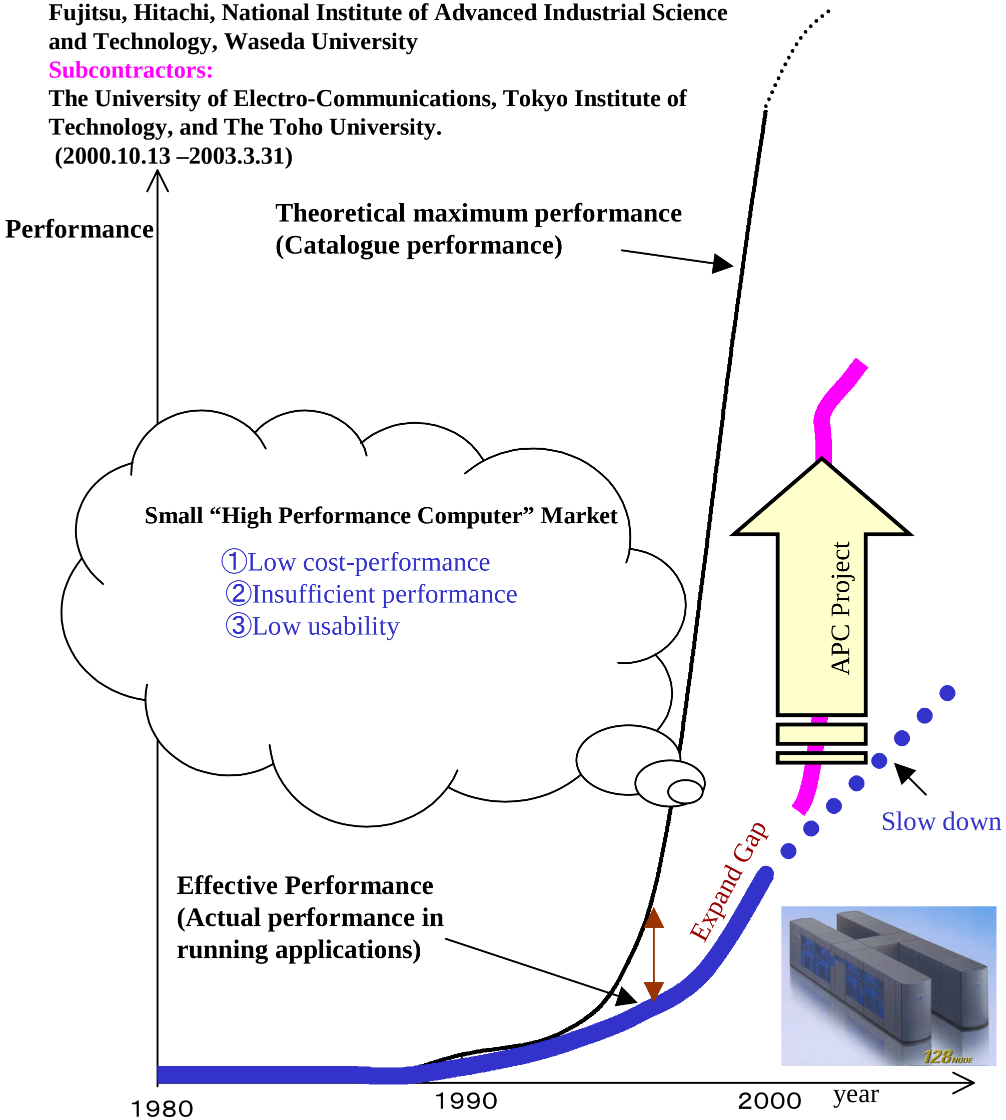
Overview of the Advanced Parallelizing Compiler Technology R&D Project

Project Leader
Hironori Kasahara
Waseda University, faculty of science and engineering, department of electric-electron information engineering
kasahara@oscar.elec.waseda.ac.jp
http://www.kasahara.elec.waseda.ac.jp

Advanced Parallelizing Compiler Technology

Contractor: **JIPDEC**
Collaborating research facilities:
Fujitsu, Hitachi, National Institute of Advanced Industrial Science and Technology, Waseda University
Subcontractors:
The University of Electro-Communications, Tokyo Institute of Technology, and The Toho University.
(2000.10.13 –2003.3.31)



Change of theoretical maximum performance and effective performance of HPC

Background and Problems

①Adoption of parallel processing as a core technology on PC to HPC
②Increase of importance of software technology on IT
③Need for improvement of cost-performance and usability

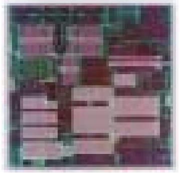
Contents of Research and Development

①Development of advanced parallelizing compiler
②Development of performance evaluation technology for parallelizing compilers

Goal: **Double the effective performance**

Ripple Effect

①Development of internationally competitive next generation PC and HPC
②Putting the innovative automatic parallelizing compiler technology to practical use
③Development and market acquisition of next generation single-chip multiprocessors
④Boosting R&D in the following many fields:



Information technology, Bio-technology, Device technology, Earth environment, Next-generation VLSI design, Financial technology, Weather forecast technology, Energy technology, Space development, Automobile technology, Electric Commerce

The Need for an Automatic Parallelizing Compiler (APC)

- Spread of muliprocessor architectures, from single-chip processors to high-performance computers
- As the number of processors increases:
  - The gap between maximum hardware performance and effective performance widens
  - Cost performance deteriorates
  - Optimization requiring advanced specialized knowledge is needed

→ Small market

- Advanced Parallelizing Compiler
  - Needed to boost effective performance, cost performance and ease of use

Current Status of APC Research and Development

- Automatic Loop Parallelization

(University of Illinois, University of Maryland, Stanford University)

  - Data-dependent analysis
    - GCD, Banerjee’s inexact/exact, Omega, symbolic analysis, inter-procedural analysis (IPA), symbolic analysis, execution-time analysis, etc.
  - Loop restructuring (automatic rewriting of programs)
    - Unrolling, strip-mining, loop merging, unimodular conversion (skewing, reversing, interchange), array privatization, tiling, etc.

University of Illinois: Polaris, Parafrase2, Promise (partly in collaboration with University of California at Irvine)

Stanford University: SUIF, National Compiler Infrastructure (NCI)

- These approaches have reached a mature phase and further significant improvements in performance will be hard to obtain (Amdahl principle)
  - Data-sensitivity and control-sensitivity between complex iterations create the presence of sequential loops
  - Hierarchical parallelization is needed where the number of loop repetitions is smaller than the number of processors

The Need for Multigrain Parallelization

- HPC: The limits of loop parallelization
  - The technology is reaching a mature phase and some loops are difficult to parallelize
  - As the number of processors increases, previously ignored sequential loops have an adverse impact on the improvement of performance (Amdahl principle)
- Microprocessors: Limits of instruction-level parallelization
  - Only command level parallelization is used for mainstream super-scholar/VLIW
  - As the level of integration increases, hardware becomes more complex (increasing number of instructions issued and functional units, speculative execution, branch prediction, etc.), but further improvement in scalability with respect to the number of transistors used becomes difficult (a multiple of 2-3 time is the limit).
- As the number of processors and transistors rises, architecture and compiling technology capable of scalability performance improvements is needed.
- Multigrain automatic parallelization
  - Improved effective performance, cost performance and ease of use from single-chip multiprocessors to HPCs
  - Development of architectures optimized for multigrain compilers leads to development of highly competitive HPCs and microprocessors

Research Trends in Multigrain Parallelizing Compiler Technology

Multilevel parallel processing

The PROMIS compiler project

- University of Illinois: Parafrase2

  (Hierarchical task parallelization using Hierarchical Task Graph (HTG) technology)

- University of California at Irvine: EVE (VLIW fine-grain parallelization)

  The prototypes were developed under joint research; currently, the University of Illinois’ project has developed a practical compiler.

NANOS compiler (NthLIB, extended Open MP)

- Technical University of Catalunya (UPC): Development of a run-time library that parallelizes nested tasks based on the HTG output by Parafrase2
- OpenMP was extended to enable hierarchical thread generation. The effort on attempts to introduce the extended OpenMP as a standard is tangled.

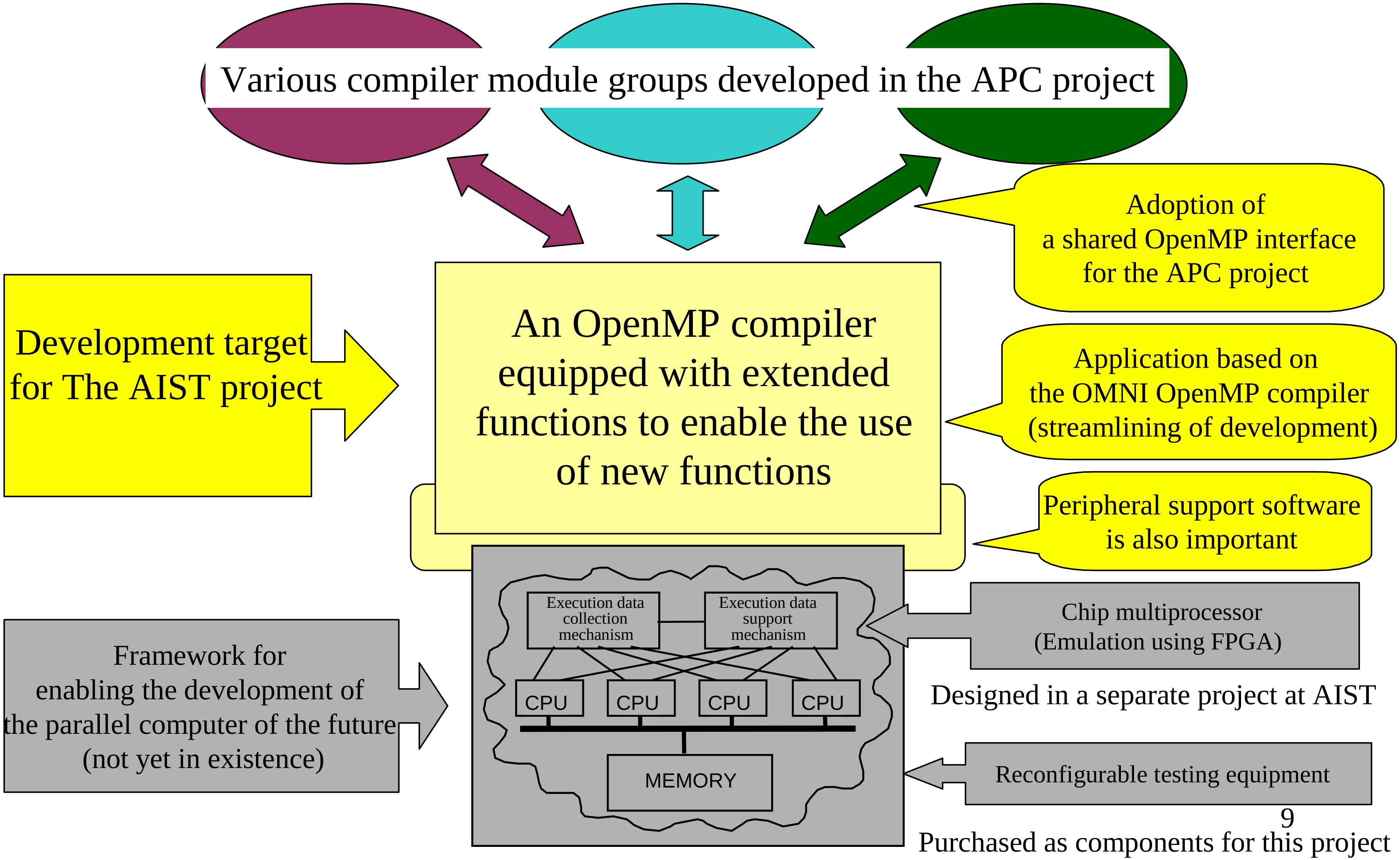
Multigrain Parallelization in the OSCAR Parallelizing Compiler

- Coarse-grain task parallelization
  - Parallelization between subroutines, loops and basic blocks
  - Parallels in each level are used to separate static scheduling during compiling from dynamic scheduling during execution, to schedule tasks to a processor or processor group.
- Loop parallelization
  - Parallelization between loop iterations
  - Iteration sets are scheduled to processors
- (Near-) fine-grain parallelization
  - Parallels between assignment statements within basic blocks
  - Static scheduling to processors
- Use of parallels through hierarchical organization

R&D Technology in the APC Project

- Development of automatic multigrain parallelizing technology
  - Multigrain parallel extraction technology(Loop parallels + coarse-grain task parallels + (near-) fine-grain parallels)
  - Data-dependent analysis technology (analysis between procedures, analysis during execution)
  - Automatic data distribution technology (effective use of distributed cache and distributed shared memory)
  - Speculative execution technology (improvement of the rate of hardware use)
  - Scheduling technology(Minimization of data transmission, reduction of concealed overhead costs from overlap between processing and data transmission)
  - Preparation of an extended parallel description language(Used as an intermediate language between OpenMP compiler modules)
- Development of parallelizing tuning technology
  - Technology to render programs visible and technology for dynamic use of information
- Performance evaluation technology
  - Development of methods for evaluation of individual functions
  - Development of methods for evaluation of general performance

R&D in chip-multiprocessor parallelizing compilers equipped with dynamic information collection mechanisms and speculative execution support mechanisms
National Institute of Advanced Industrial Science and Technology



Various compiler module groups developed in the APC project

An OpenMP compiler equipped with extended functions to enable the use of new functions

Development target for The AIST project

Adoption of a shared OpenMP interface for the APC project

Application based on the OMNI OpenMP compiler (streamlining of development)

Peripheral support software is also important

Framework for enabling the development of the parallel computer of the future (not yet in existence)

Execution data collection mechanism
Execution data support mechanism
CPU
CPU
CPU
CPU
MEMORY

Chip multiprocessor (Emulation using FPGA)

Designed in a separate project at AIST

Reconfigurable testing equipment

Purchased as components for this project

Overview of the Activities of the 2000 APC Committees

(Not including the Development Promotion Committee, Technology Committee and Network Council)

■Development Promotion Committee Met twice, in the weeks of October 13 and April 23, 2001, to discuss approaches to R&D

■Technology Committee Met three times: December 14, 2000, January 31, 2001 and February 20, 2001

Setting of targets, research plans, approaches of the research association, year-end reports, intellectual property rights, methods of publication of results, budgets

International Advisory Committee

- Coordination with overseas bodies
- Judging appropriateness of targets
- Overseas publication of results
- Globally renowned professors evaluate our activities as advice for evaluation committee

✧Prof. Padua, University of Illinois, global authority on parallelizing compilers (Parafrase, Cedar, Polaris)
✧Prof. Lam, Stanford University, specialist in Suif, NCI, Suif Explorer compilers and tuning tools
✧Prof. Eigenmann, Purdue University, specialist in Polaris and SPEC HPC compilers and performance evaluation
✧Prof. Irigoin, Ecole des Mines de Paris, specialist in inter-procedural analysis in the PIPS Automatic Parallelizer

Invited overseas researchers

✧Prof. Eigenmann, Purdue University, October18, 2001, “Performance evaluation in the Polaris compiler”
✧Prof. Gao, University of Delaware, March 22-23, 2001, Coordination between compiler and architecture”
Founder of the IEEE ACM International Conference on Parallel Architectures and Compilation Techniques (PACT) and Advisor to China’s Parallelizing Compiler Research Committee

Overview of APC Committee Activities in the 2000 Fiscal Year

(Does not include Technology Forums, Administrative Liaison Forums, Intellectual Property and OpenMP Forums or the Network Council)

■Technology forums held 10 times a year

October 24 and 31, November 15 and 22, and December 6 and 19, 2000; January 10 and 31, February 20 and March 21, 2001 (reporting session for the 2000 fiscal year)

Research plans, research reports, discussions, overseas survey reports

SC2000, ASPLOS-IX, Cluster2000, MICRO33, HiPC2000, HPCA, MICRO

■Administrative Liaison Forums

Held twice in 2000 (October 23, December 27)

Budgets, external assignment, contracting of clerical and administrative work, interim inspections, etc.

■Intellectual Property Rights Examination Committee

Once in fiscal 2000: October 12 (also discussed in the Technology Committee)

Rules on handling of intellectual property rights of APC technology research bodies

■OpenMP Examination Committee

Once in fiscal 2000: February/20/2001

Examination of extended specifications for OpenMP

Clarifying Targets in the Basic Plan

(Technology Committee, January 31, 2001)

Extract from Targets in the Basic Plan

(1) Development of automatic multigrain parallelizing technology
(2) Development of parallelizing tuning technology

(1) The project aims to establish a platform-free automatic multigrain parallelizing technology that can automatically extract and effectively process parallels from each level of a program.

(2) The project aims to establish an interactive parallelizing tuning technology that performs dynamic analysis using profile and other data.

(3) To evaluate the technologies developed as described in 2 above, the extended parallel description language that is output will be run on a multiprocessor computer system, such as will serve as the platform for this technology. Using the evaluation method described in R&D item [2], “Development of a technology for evaluating the performance of parallelizing compilers,” this performance will be compared with the performance of automatic parallelizing compilers in extracting parallels from existing single grains in various SMP systems. The target is to achieve improvement of 100% or better.

(1) Development of methods for evaluation of individual functions
(2) Development of general performance evaluation methods

By carrying out the evaluation of R&D item [1], “Development of an Advanced Parallelizing Compiler Technology,” using the methods developed, the project aims to achieve a technology for objective evaluation of the performance of parallelizing compilers using SMP systems.

<Clarification plan>

- This project aims to roughly double the processing performance (halve the processing time) of SMP systems in comparison with single-grain parallelizing compilers commercially available at the time of the start of this project. The number of processors used in this comparison will be whatever number minimizes the processing time for each compiler.

· To ensure the objectivity of the program used in the general evaluation, the number of processors will be selected using a benchmark program. The reason for the selection will be clearly stated at the time of selection.