



Development of Advanced Parallelizing Compiler Technology

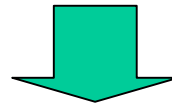
Group Leader
Sumio Kikuchi (Hitachi Ltd.)

Details and Targets of Research and Development

Technological development items

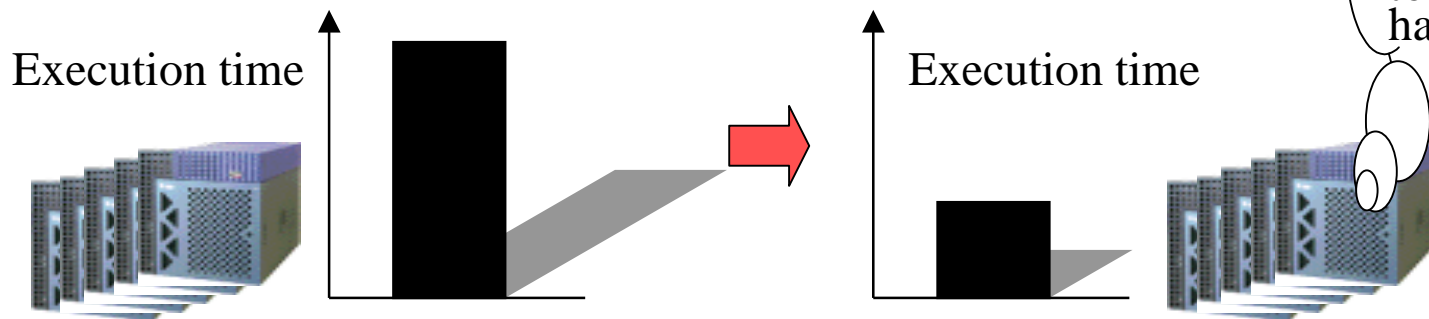
Development of software technology that enables anybody to double their effective performance

- 1) Development of APC technology
- 2) R&D in technologies to evaluate the performance of parallelizing compiler



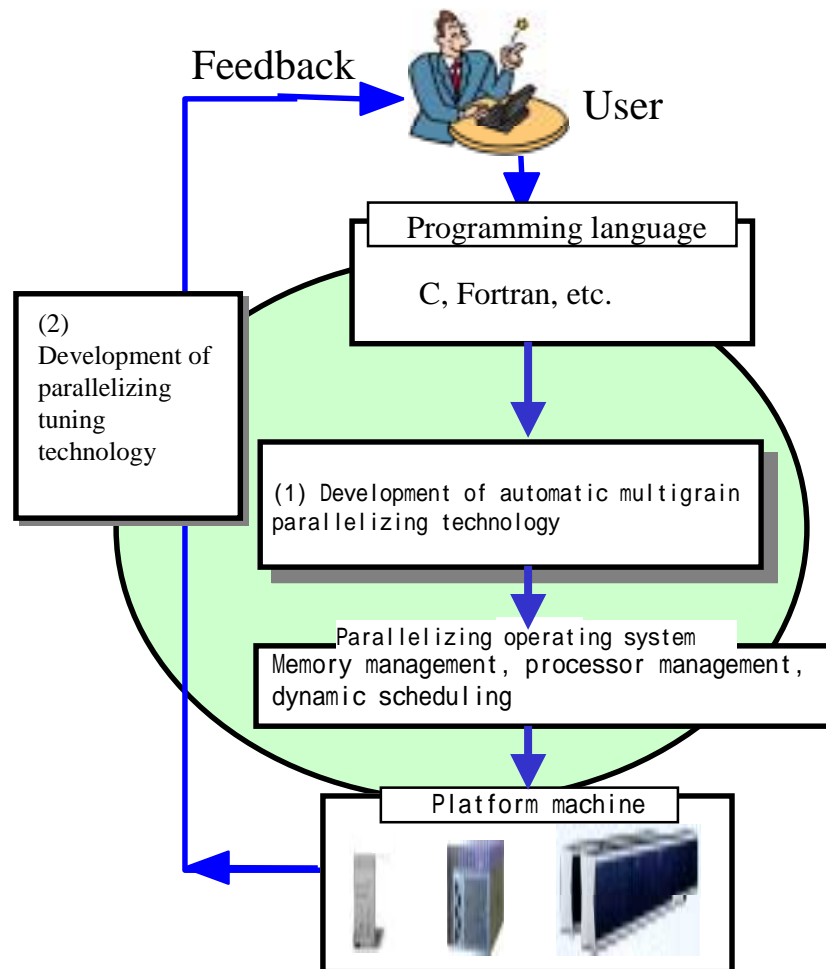
Development target: Doubling of effective performance

Achievement of 50% or better reduction (doubling in cost-performance) in execution time with the same hardware



With the development of this technology, the purchase of a computer with the same performance enables costs to be cut in half.

Details of Research and Development



(1) Development of automatic multigrain parallelizing technology

This development provides automatic extraction of hierarchical parallels using data-dependent analysis technology, speculative execution technology, automatic data distribution technology and scheduling technology.

(2) Development of parallelizing tuning technology

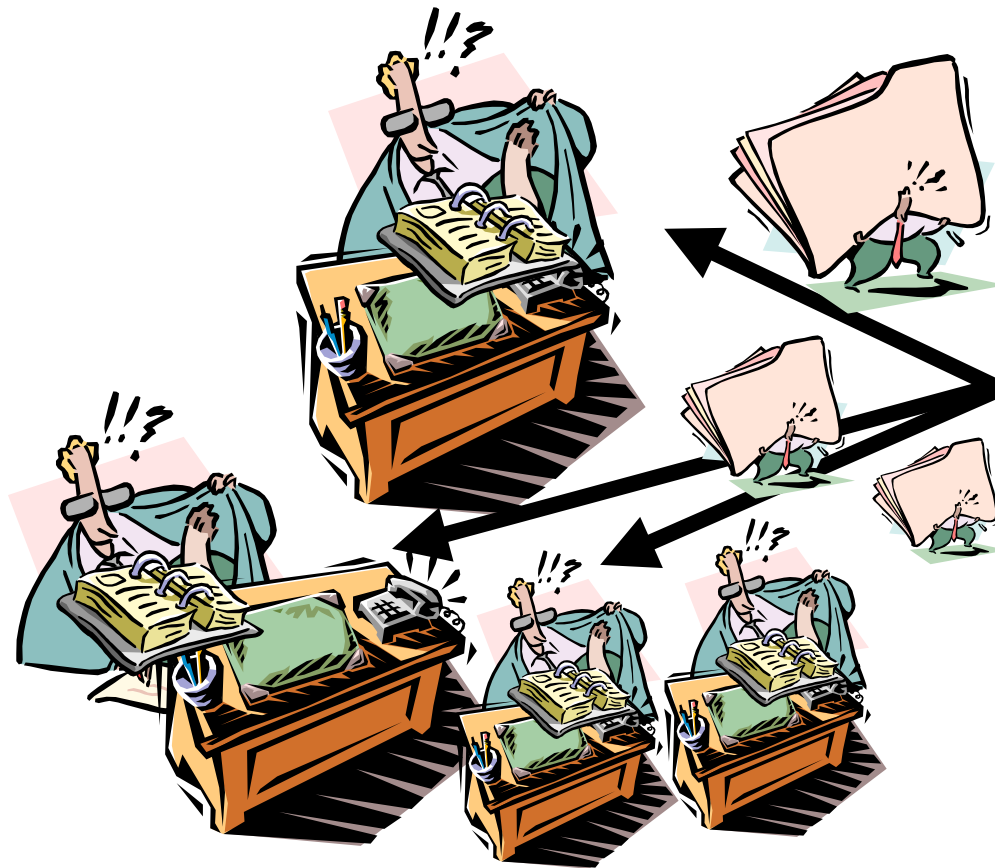
This development provides interactive, platform-free, parallelizing tuning tools that use data unavailable to static analysis, including user data and profile data that outline program behavior.

Key Issues

Technology items	Conventional technology	Key issues
Multigrain parallelization	Use of single-grain parallelization alone at the instruction level, loop level, etc	Distribution and parallelization to the optimum grain, to achieve high parallelization in each part of the program
Speculative execution technology	Achievement in a single grain within the processor	Speculative execution in multiple grains
Automatic data distribution technology	Optimization requires explicit instruction from the user	Tasks are optimally distributed automatically even without explicit instructions from the user
Scheduling technology	Single-grain scheduling technology	Scheduling technology that is tailored to multigrain parallelization (multigrain)
Optimization based on dynamic program behavior data	Use of static data only	Technology of using dynamic information, that is obtained only on time of execution, by compiler

Multigrain Parallelization

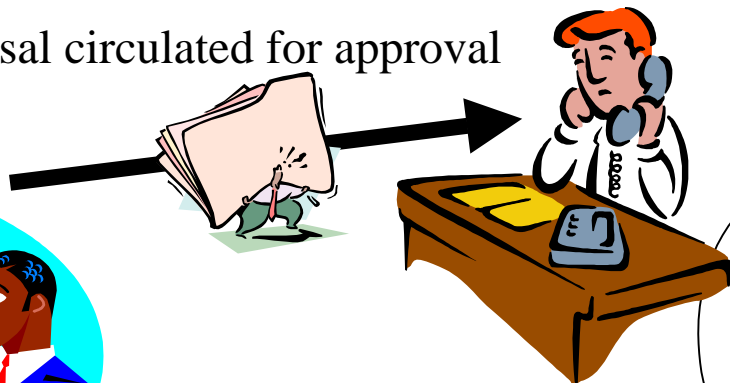
Each person in this illustration represents a processor.



No matter how large the job is or what type of job it is, we can do the job fast because we can divide it skillfully among many people!
(Multigrain parallelization)

Speculative Execution Technology

Draft proposal circulated for approval



Like draft proposals and other materials circulated for approval, data must be circulated (processed) in sequence



This can be a time-consuming process.

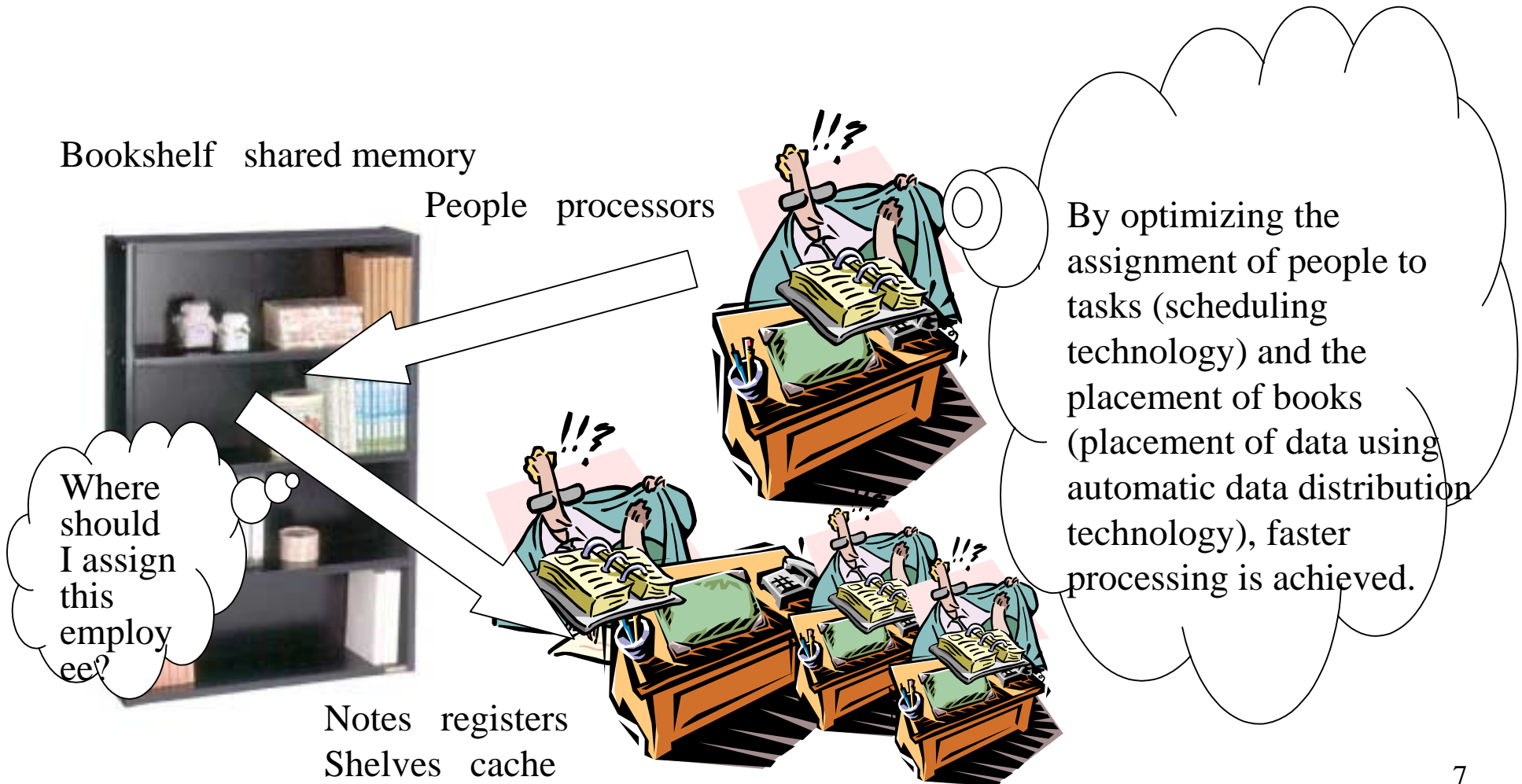


Like office staff anticipating the arrival of a circular, speculative execution software predicts what sort of processing is required and completes it in advance of receiving the data.

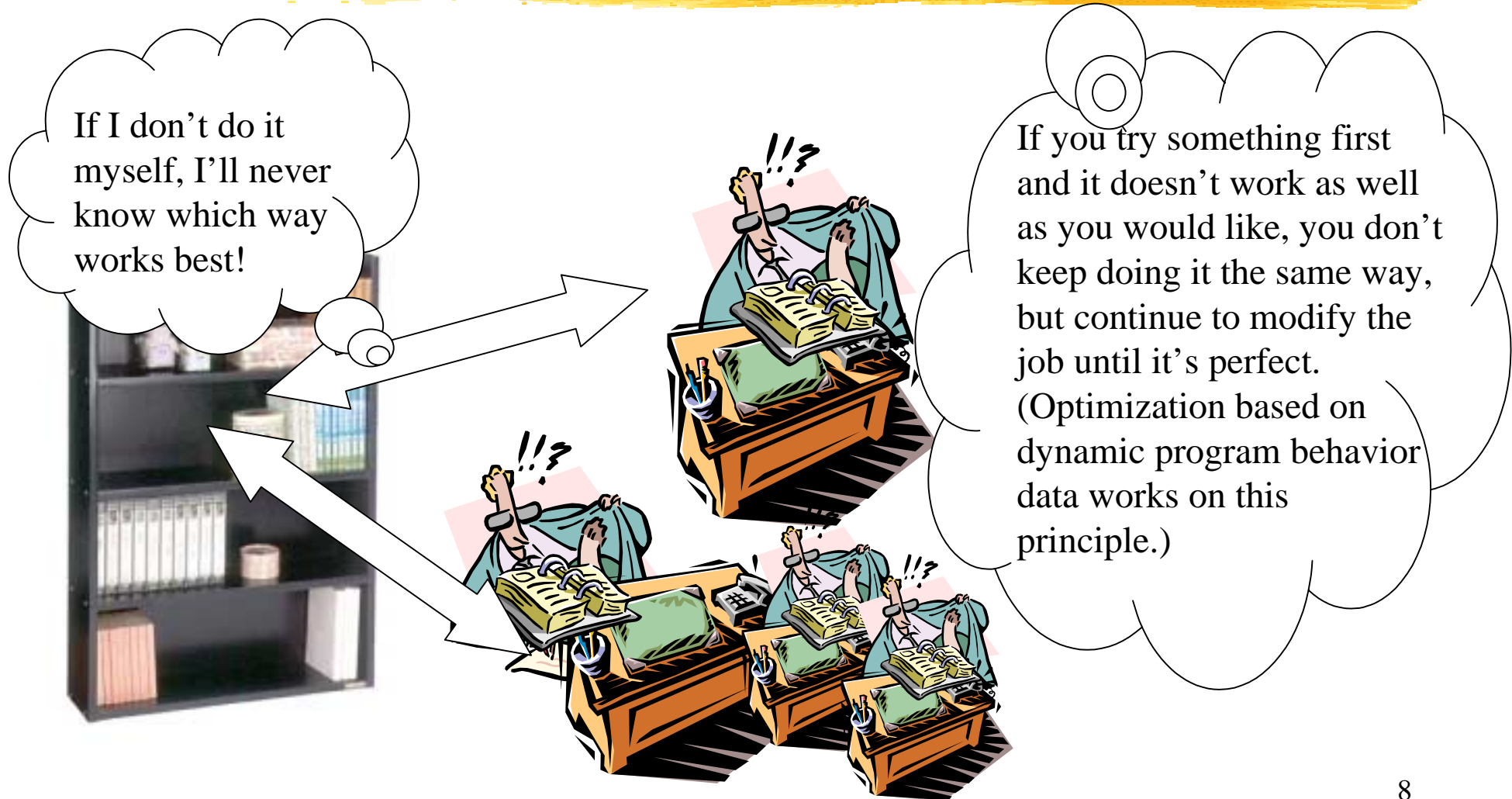


稟議資料

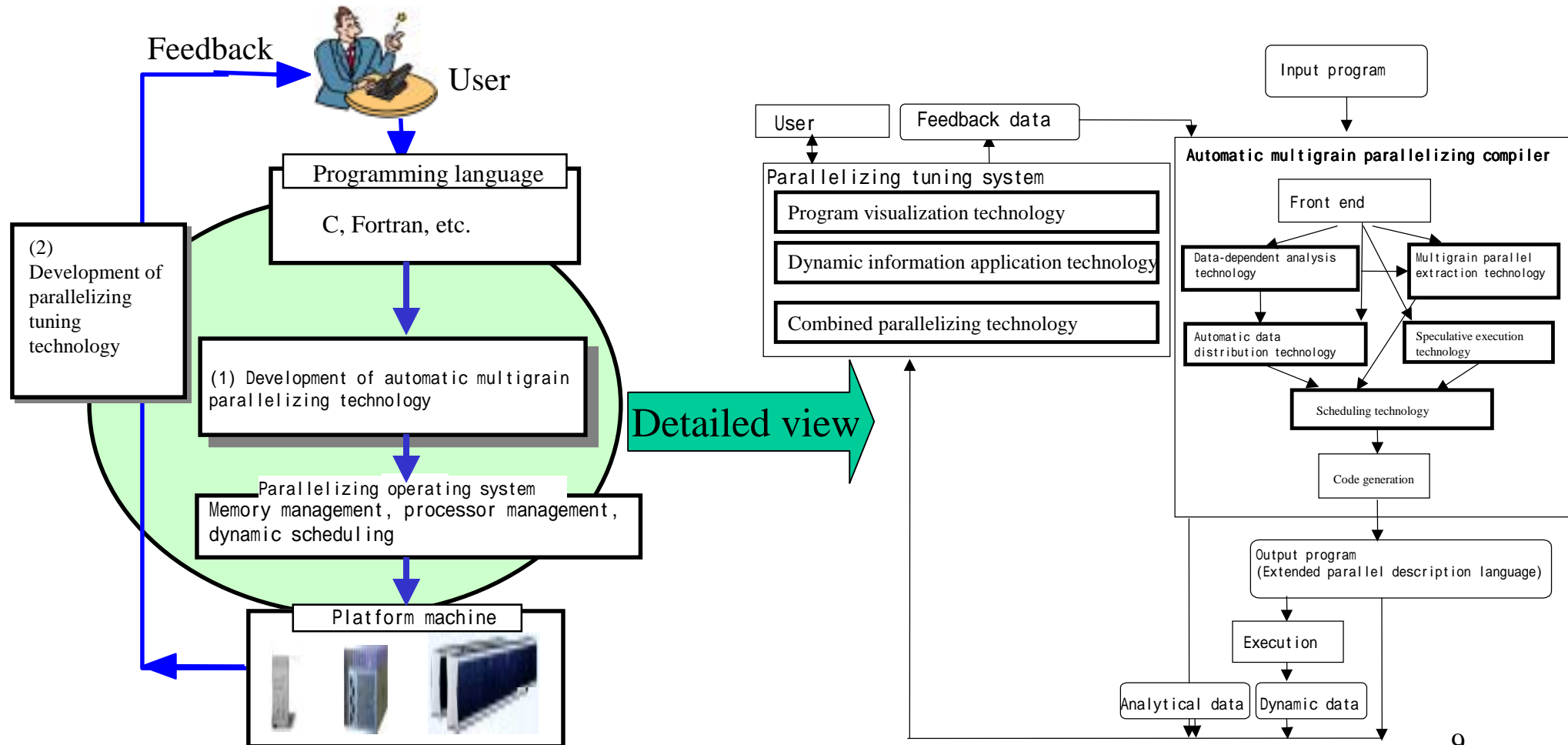
Automatic Data Distribution Technology Scheduling Technology



Optimization Based on Dynamic Program Behavior Data



Configuration of Advanced Parallelizing Compiler



Progress in FY 2000, Plan for FY 2001

Main issues for discussion Technological items	Technology items	Progress in FY 2000	2000 Academic presentations		Plan for 2001
			Overseas	In Japan	
	1) Development of APC technology	Examination of elemental technologies	2	16	Development of elemental technologies
Multigrain parallelization	1) Development of automatic multigrain parallelizing technology				
Speculative execution technology	(a) Multigrain parallel extraction technology	Some prototypes, preliminary evaluation	2	4	Extension and development
Automatic data distribution technology	(b) Data-dependent analysis technology	In process of examination and design			Development
Scheduling technology	(c) Automatic data distribution technology	Some prototypes, preliminary evaluation		3	Extension and development
Optimization based on dynamic program behavior data	(d) Speculative execution technology	In process of examination and design		1	Development
	(e) Scheduling technology	In process of examination and design		5	Development
	(f) Preparation of an extended parallel description language	Prototype is proposed		1	Development of basic components
	(2) Development of parallelizing tuning technology				
	(a) Program visualization technology	Some prototypes		2	Development
	(b) Dynamic information application technology	(Not yet implemented)			Development of basic components
	(c) Combined parallelization technology	(Not yet implemented)			Examination and preliminary evaluation

Multigrain parallel extraction technology



■ Basic technology for extraction of multigrain parallels

- A prototype function for conversion of multigrain parallels to OpenMP was developed in 2000
SPECfp95 achieved a 60% increase over IBM in “swim,” and a 100% increase over IBM in “mgrid.”
- In 2001, work will begin on development of a multigrain parallel extraction technology to enable effective use of cache and distributed shared memory.

■ Extraction of parallels from incomplete nested loops

- A prototype for the basic function was produced in 2000 and features will be extended in 2001 to eliminate some of the restrictions.

■ Extraction of multigrain parallels between procedures

- Algorithms were designed in 2000 and a prototype for basic functions will be prepared in 2001.

Data-dependent Analysis Technology



■ Conditional data flow analysis

- Algorithms were designed in 2000, based on technical discussions
- A prototype of basic functions will be built and evaluated in 2001.

■ Analysis of data flow at time of execution

- Nothing has been started in 2000. In 2001, basic functions will be designed and a prototype built.

■ Fortran90 compatibility

- A prototype for basic functions was built in 2000 and work will begin on the examination of extended functions in 2001.

Automatic Data Distribution Technology

■ Automatic data distribution for use in distributed memory

- An extension for indirect referencing arrays was proposed in 2000 and a preliminary evaluation was conducted .
 - 50% improvement in performance was confirmed in NPB computer graphics
- In 2001, a prototype of the functions proposed in 2000 will be evaluated and work will begin on examination of extended functions.

■ Automatic data distribution for distributed cache

- A prototype of basic functions was completed in 2000 but work is suspended in 2001 due to budget restrictions.

■ Data locality optimization for distributed shared memory

- In 2000, loop matching and distribution methods were developed, and data transmission gain/CP methods were proposed and preliminary evaluations conducted. In 2001, a prototype will be constructed and evaluated.
 - 50% improvement in performance was confirmed in “tomcatv” of SPECfp95.

Speculative Execution Technology



■ Speculative Execution of multigrain

- In 2000, parallelization was subjected to rigorous desktop examination using - SPECint2000.

Algorithms for speculative execution were completed in the first half of the fiscal year.

- In 2001, R&D will be conducted on speculative parallel execution formats for parallel tasks

Scheduling Technology



■ Scheduling to minimize data transmission

- Development of algorithms began in 2000.
 - In addition, standard task-gram sets were proposed for use in evaluating scheduling methods
- In 2001, algorithms will be developed to conceal overhead from data transmission resulting from overlap between coarse-grain task processing and data transmission.

Preparation of an Extended Parallel Description Language



- Proposed extended specifications for a common OpenMP within APC
 - In 2000, extended specifications were proposed and evaluated distributed-memory computers, and the elements of common specifications within APC were prepared.
 - In 2001, functions for generation and input of OpenMP source programs in the C language will be developed.

Parallelizing Tuning Technology



■ Program visualization technology

- Program slicing
 - No work was done in 2000, but a prototype for analytical processing will be developed in 2001.
- Hierarchical display of data-dependent positions
 - A prototype of display processing was developed in 2000. A prototype of analytical processing will be completed in 2001.

■ Dynamic information application technology

- No work was done in 2000, but a prototype of basic functions will be developed in 2001.

■ Combined parallelization technology

- No work was done in 2000, but examination and preliminary evaluation will be conducted in 2001.

Portion of the APC Project under the Responsibility of the National Institute of
Advanced Industrial Science and Technology
Development of a chip-multiprocessor parallelizing compiler equipped with dynamic
information collection mechanisms and speculative execution support mechanisms

Hanpei Koike

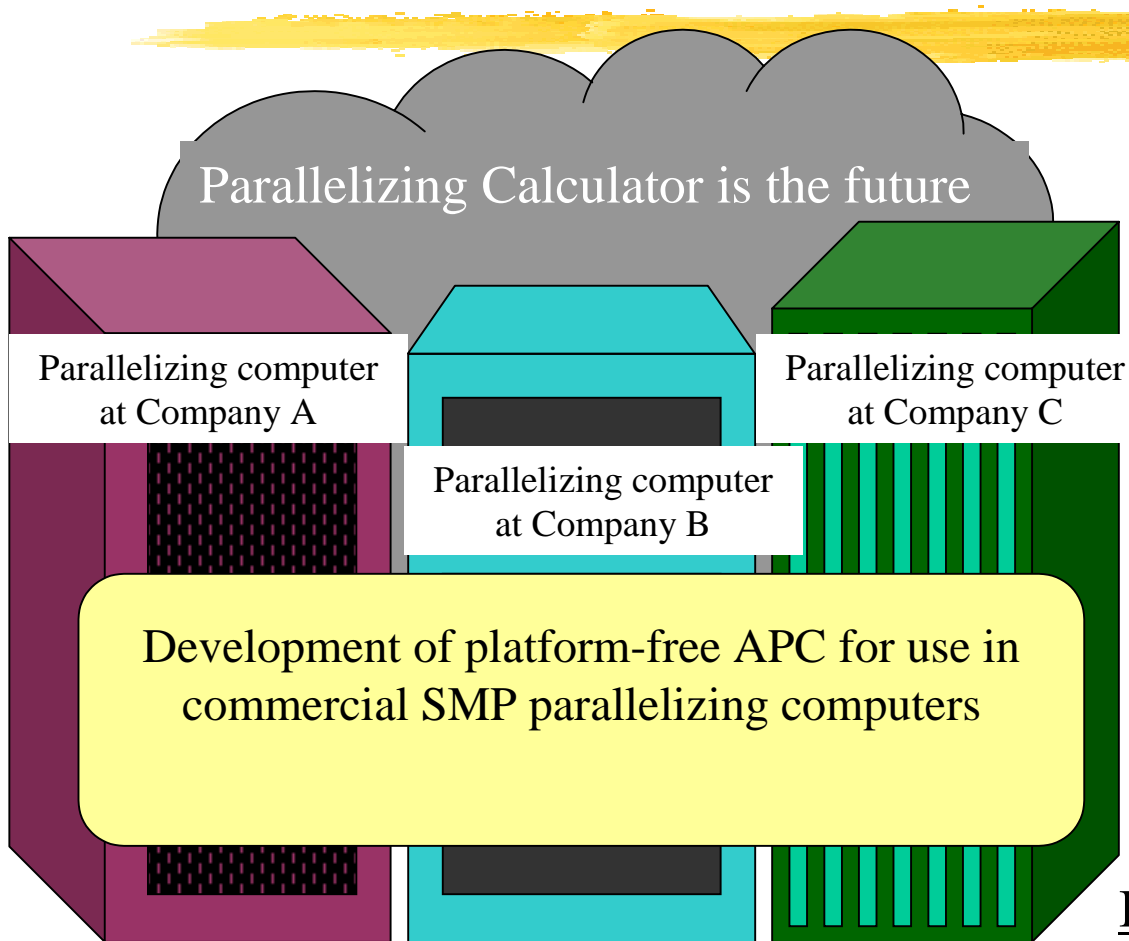
Advanced Parallelizing Compiler Research Body

&

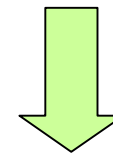
Ministry of Economy, Trade and Industry, National Institute of Advanced Industrial Science
and Technology and Electrotechnical Laboratory (- 3/31)

Independent governmental Cooperation, National Institute of Advanced Industrial Science
and Technology, Information Processing Research Department (4/1-)

Contribution of the National Institute of Advanced Industrial Science and Technology to the APC Project



What new parallelizing compiler technologies and parallelizing tuning technologies will APC require for the parallelizing computers of the future?



The National Institute of Advanced
Industrial Science and Technology knows.

What will the standard platform look like several years from now?

Chip multiprocessors

+

Speculative execution support mechanism

+

Dynamic information collection mechanism

Cf. IBM Power4, Compaq/DEC Piranha,
Stanford Hydra, NEC Merlot, Intel?

Parallelizing computers will incorporate a large number of computers on silicon chips capable of integrating as many as a million transistors.

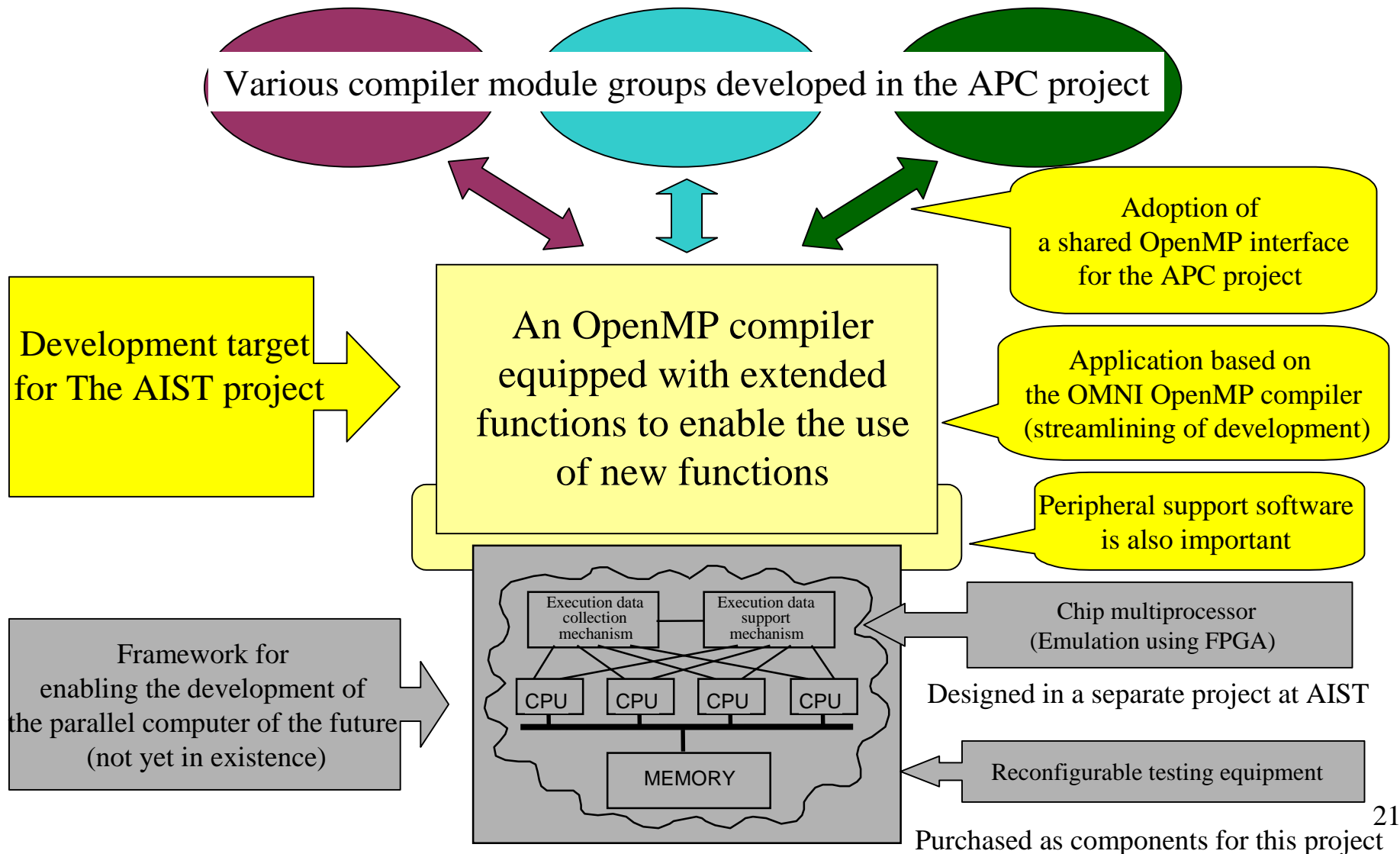
This mechanism maximizes parallel processing based on speculative forecasts of operations that are expected not to cause problems if executed simultaneously.

Information on the execution characteristics of programs that is only acquired when the program is run is fed back to the compiler, boosting the results of compiling.

Objective of research:

The purpose of this research is to establish parallelizing compiler technology and parallelizing tuner technology, clarify the performance and scope of application of each, and feed back these results into the overall APC project.

R&D in chip-multiprocessor parallelizing compilers equipped with dynamic information collection mechanisms and speculative execution support mechanisms



Progress of Research (1)

■ Details and results of research in FY 2000

- Examination of a program model that assumes information collection and speculative execution
 - Development of a model in which various execution data are held as arc characteristics in a control-data-dependent graph
 - Examination of an overview of the extended functions of OpenMP
 - Exposition of all required speculative execution functions and information collection functions
- Koike, “Examination of chip multiprocessors equipped with dynamic information collection mechanisms and speculative execution support mechanisms,” Report of the Architecture Research Society of the Information Processing Society, 2001-ARC-141, pp. 47-52, January 2001
- Preparing the compiler development environment
 - Development of a suite of development support software (externally sourced)
 - Preliminary design of support software modules

Progress of Research (2)

■ Plans for FY 2001

- Examination of specifications for extended functions of an OpenMP compiler
 - Examination and determination of specific extended language specifications
- Start of mounting of OpenMP compilers with extended functions
 - Work on the OMNI OpenMP compiler, with extended functions based on the examination described above
 - Development of support software modules
- Information collection test based on the dynamic information collection mechanism
 - Feedback to the project of basic data from a new perspective

■ Final targets

- Launch of an OpenMP compiler with extended functions on a chip multiprocessor
- Evaluation of program execution performance using compilers equipped with the above extended functions and feedback of those results to the project