

# A Distributed Problem Solving Environment (PSE) for Scientific Computing

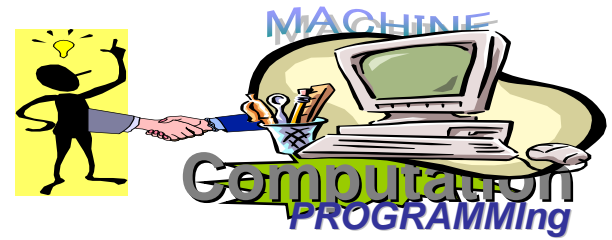
Shigeo Kawata<sup>1)</sup>, Hideaki Fuju<sup>1)</sup>, Hideaki  
Sugiura<sup>1)</sup>, Yuichi Saitoh<sup>1)</sup>, Yoshikazu Hayase<sup>2)</sup>,  
Takayuki Teramoto<sup>3)</sup>, Takashi Kikuchi<sup>1)</sup>,  
Hitohide Usami<sup>4)</sup>

- 1) Graduate School of Engineering, *Utsunomiya University*
- 2) Department of Electronic Control Engineering, Toyama  
National College of Maritime Technology
- 3) Tsuyama National College of Technology
- 4) National Institute of Informatics

# OUTLINE

1. PSE: Purposes & Background
2. D-NCAS  
for Programming-Free Computation  
& Job Execution Support  
on distributed computers
4. PSE Research Group & Future in PSE
5. Summary & Conclusions

# Background



HPC & Simulations in Science & Engineering

—>

/ Useful & Required at Companies, Labs & Univ.  
for Researches & Developments

/ The third method following Theoretical &  
Experimental approaches

However

—> requires time & hard work for simulations!

—> =PSE (Problem Solving Environment)

→ We want to focus on the Problem itself!

# Background

7 faults / 1000 lines in Computer code

Ref.: D.E. Post & L.G. Votta, Physics Today (2005) January, p.35

→Lack of Programming Power

→PSE Supports

the programming!

*Computing Power =*

*Computer Power + Algorithm Power + Programming Power*

*PSE* 

# Problem Solving Environment (PSE) Definition

## **Definition:**

Computational Environment for Problem Solving : Software & Hardware

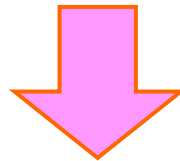
*A system that provides all the computational facilities necessary to solve a target class of problems. It uses the language of the target class and users need not have specialized knowledge of the underlying hardware or software.*

by **John R. Rice**  
IEEE Computational Science & Engineering  
Summer 1994.

- Not required special knowledge for Computer & Software
- Computer & Software Environment is a Tool for Solving & Thinking target Problems
- PSE is a Research area aiming to reduce time & efforts required to solve the target problems.

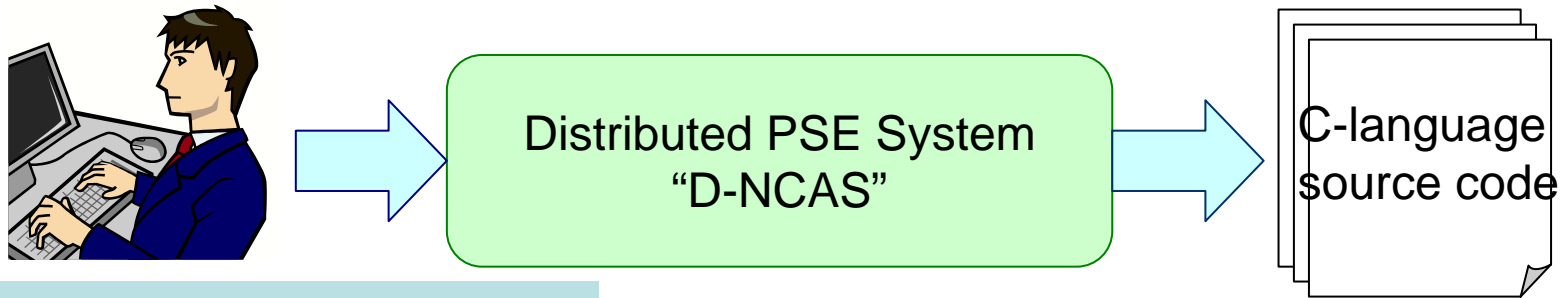
## Background

- Simulation in Science & Engineering
  - Requires time & Hard work for simulations
  - Computer & Software Environment is a Tool for Solving & Thinking target Problems
- **Module-based Distributed PSE System**
  - Easy for Module addition & update
  - Flexible system
  - Easy for maintenance



Distributed PSE System “D-NCAS”

# Distributed PSE System “D-NCAS”



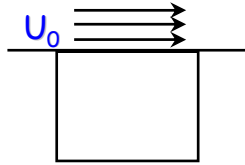
**Input:**PDEs, initial conditions, etc...

**Output:**simulation program

- Simulation Assisted System
  - partial differential equation (PDE) based PSE
  - D-NCAS inputs a problem information including PDEs, initial and boundary conditions and discretization scheme
  - Outputs a C-language source code for the problem
- Standard Technology
  - Protocol => SOAP
  - Data format => XML

# Language of the target class of problem

Physical model



Mathematical model

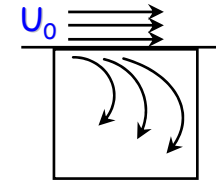
$$\frac{\partial \mathbf{v}}{\partial t} = -(\mathbf{v} \cdot \nabla) \mathbf{v} - \nabla P + \frac{1}{R_e} \Delta \mathbf{v}$$

Discretization model

$$\frac{u_{i,j}^{n+1} - u_{i,j}^n}{dt} = -u_{i,j}^n \frac{u_{i+1,j}^n - u_{i,j}^n}{dx} + \dots$$

Solver  
Generator

$$u_{i,j}^{n+1} = u_{i,j}^n + dt \left( -u_{i,j}^n \frac{u_{i+1,j}^n - u_{i,j}^n}{dx} + \dots \right)$$



Visualization  
& evaluation  
of results

Language of the Target class problem

Language of underlying software & hardware

Programming  
language

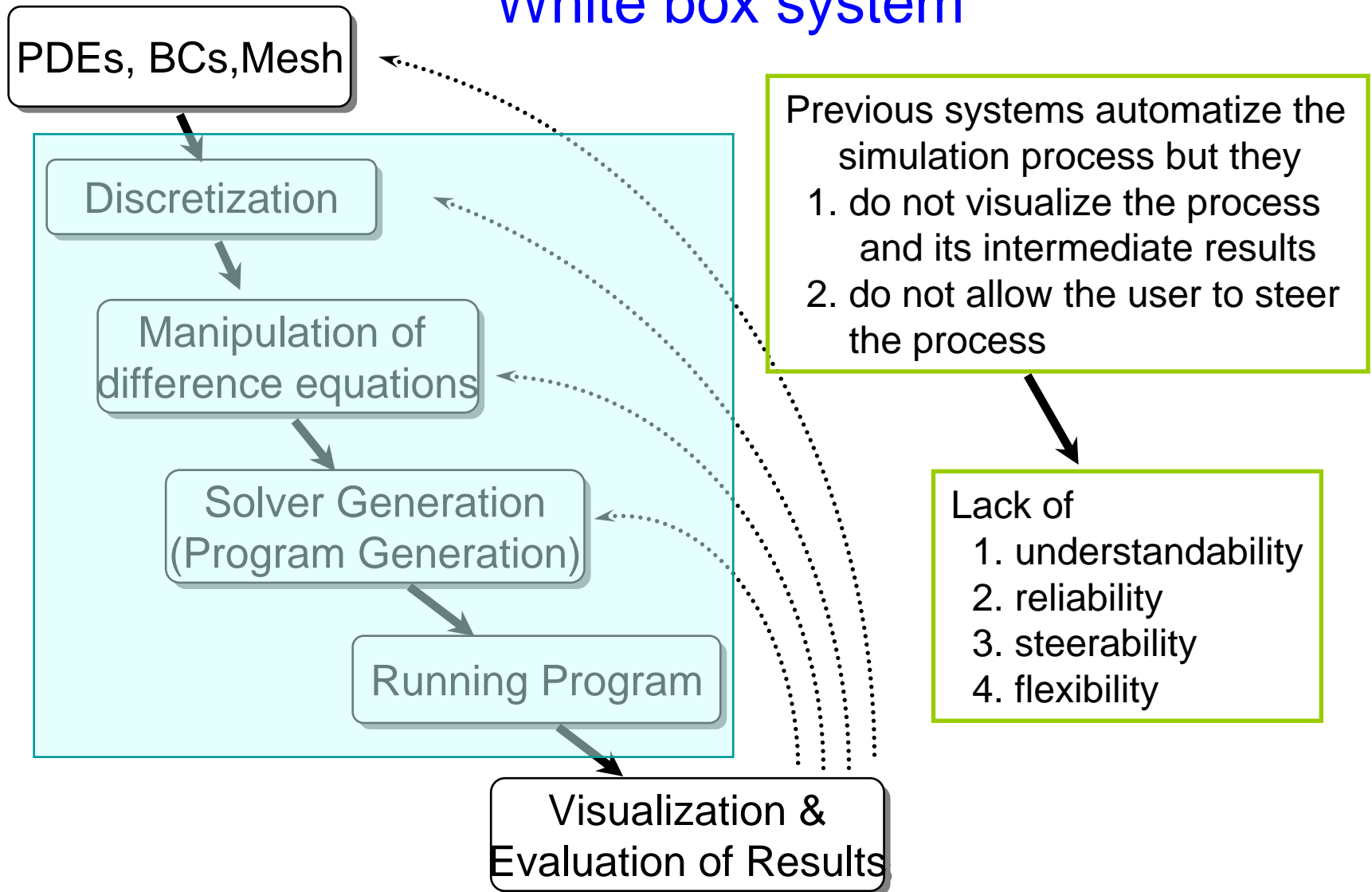
```
for (i=1;i<=IMAX;i++) {
  u[n+1][i][j]=u[n][i][j]+dt*(-u[n][i]
  [j]*(u[n][i+1][j]-u[n][i][j])/dx+...
```

Numerical  
results

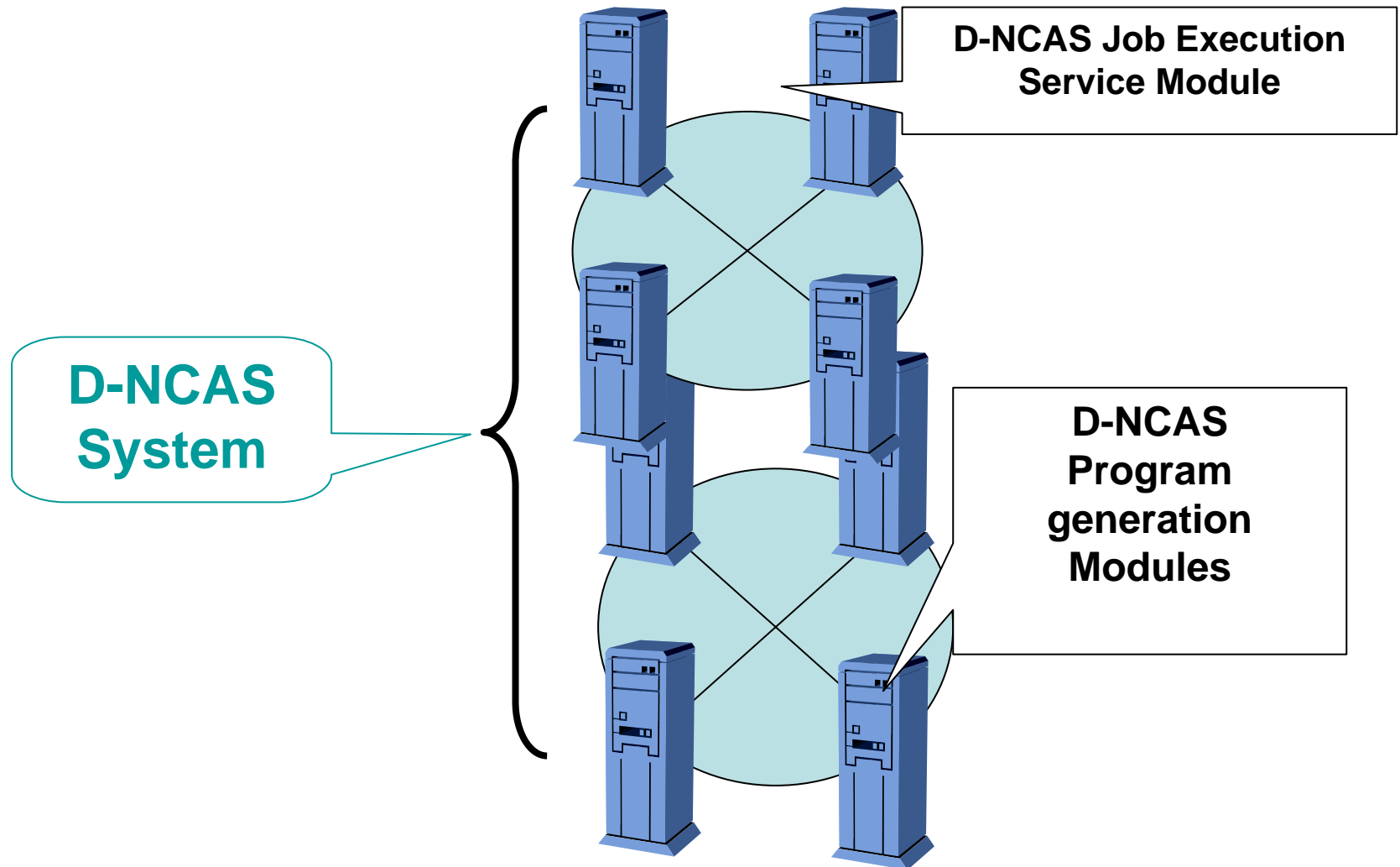
```
0.0e0 0.2e0 0.4e1 0.5e0 0.8e0
0.1e0 0.12e0 0.1e1 0.1e0 0.3e0
...
```



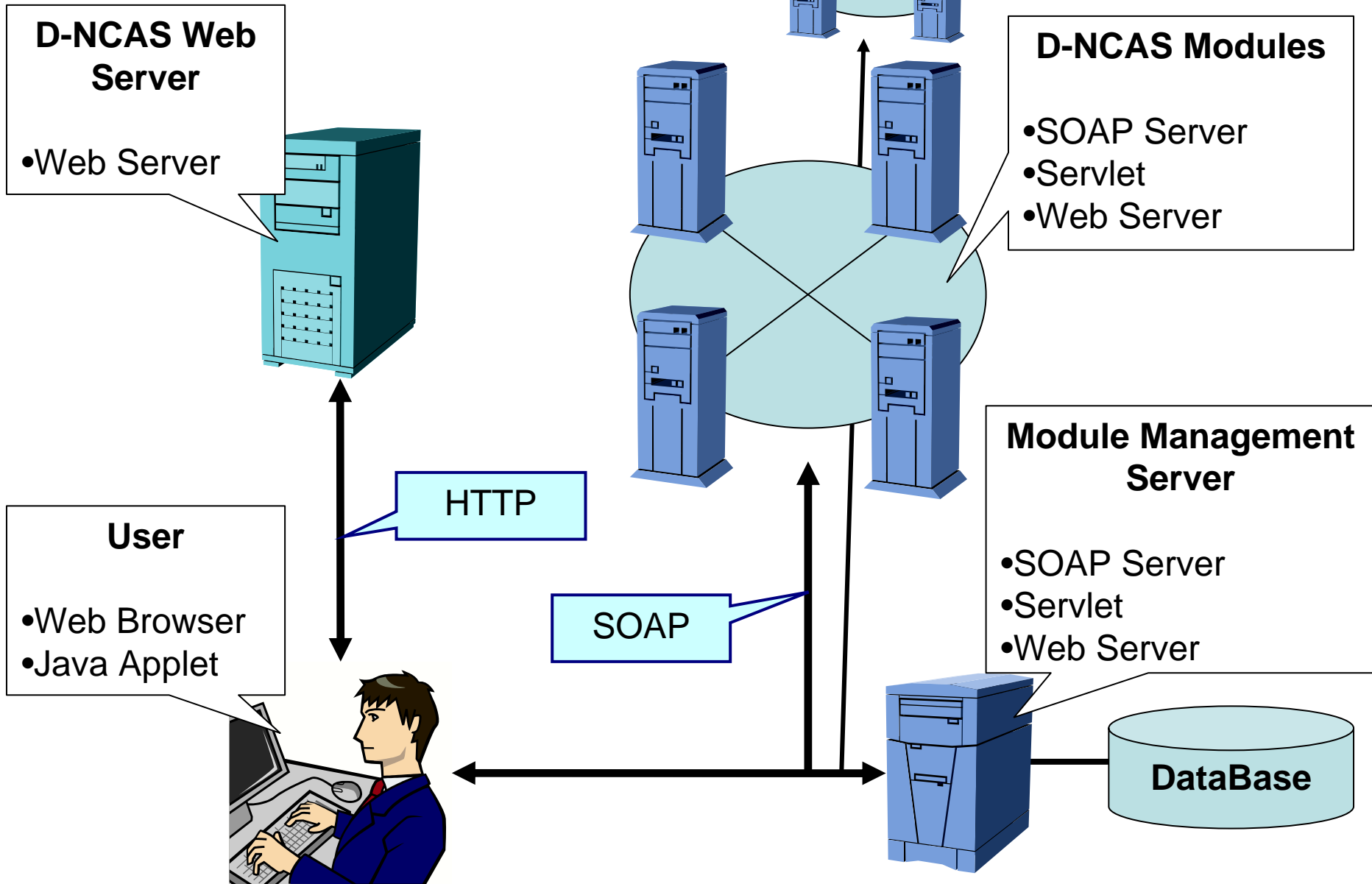
# White box system



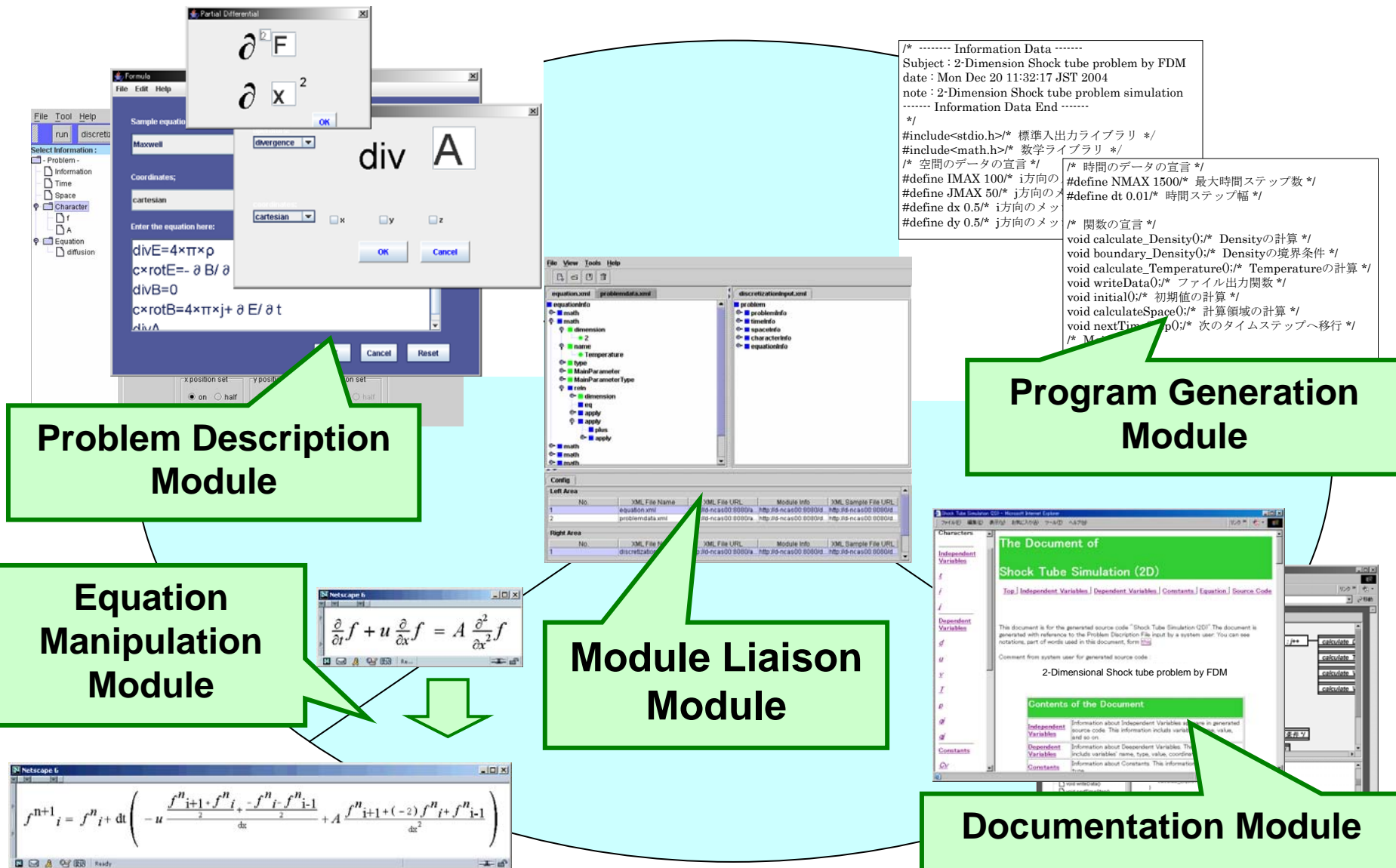
# System Structure



# System Structure



# D-NCAS Modules



# Module Management

D-NCAS

Process List

Contents

- Module List
- Process List
- User List

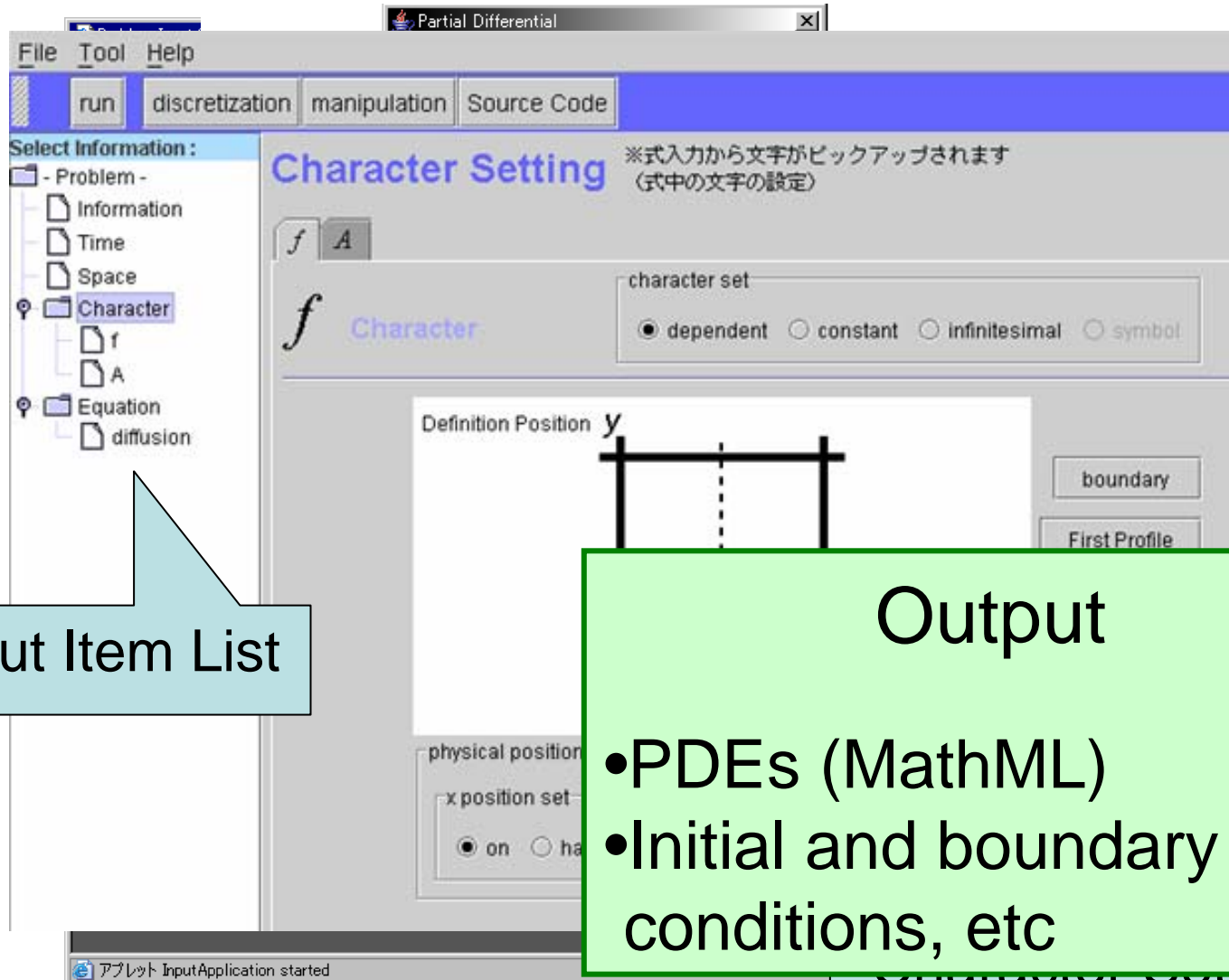
Powered by Kawata Lab.

No.	name	location	date	author
1	<a href="#">FDMprocess</a>	<a href="http://d-ncas00:8080/d-ncas/system/process/FDMProcess.xml">http://d-ncas00:8080/d-ncas/system/process/FDMProcess.xml</a>	2004-12-09-1721	inaba
2	<a href="#">FDMprocess2</a>	<a href="http://d-ncas00:8080/d-ncas/system/process/FDMProcess2.xml">http://d-ncas00:8080/d-ncas/system/process/FDMProcess2.xml</a>	2004-12-14-1445	inaba
3	<a href="#">FDMprocess3</a>	<a href="http://d-ncas00:8080/d-ncas/system/process/FDMProcess3.xml">http://d-ncas00:8080/d-ncas/system/process/FDMProcess3.xml</a>	2005-01-20-1723	inaba
4	<a href="#">FEMprocess</a>	<a href="http://d-ncas00:8080/d-ncas/system/process/FEMProcess.xml">http://d-ncas00:8080/d-ncas/system/process/FEMProcess.xml</a>	2004-12-14-1528	inaba

Link to Detail Information

URL

# Problem Description Module



Input  
Assistant

Input Item List

Output

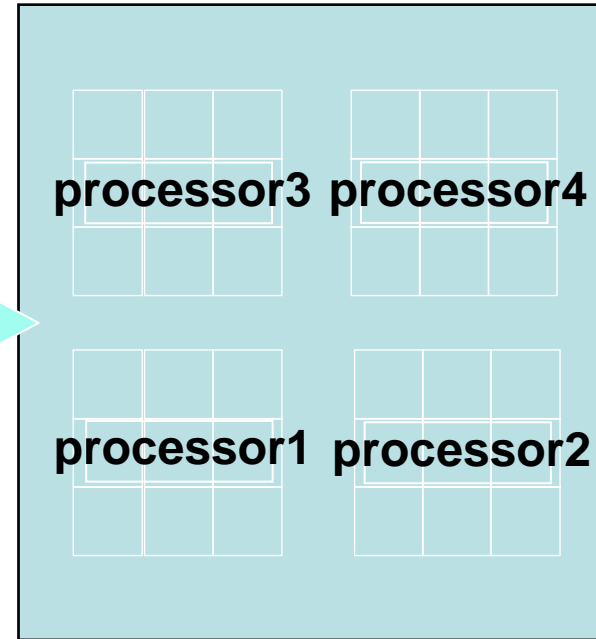
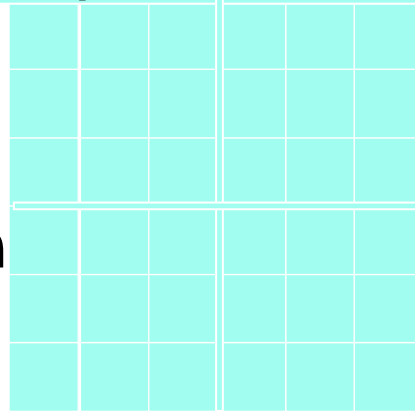
- PDEs (MathML)
- Initial and boundary conditions, etc

# P-NCAS module: Parallel Program generation

Computational domain

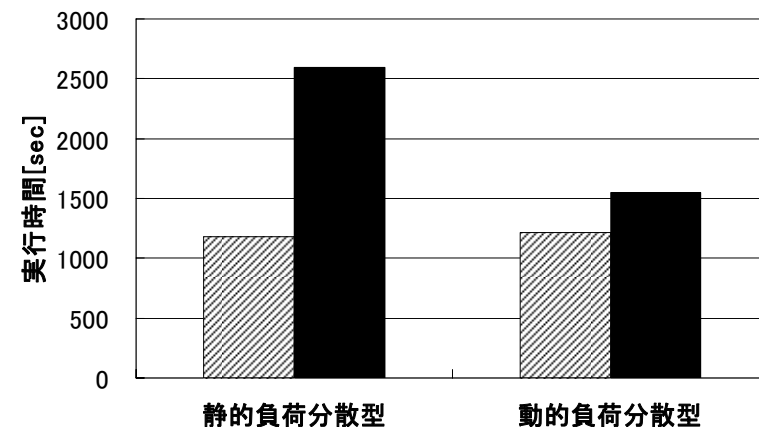
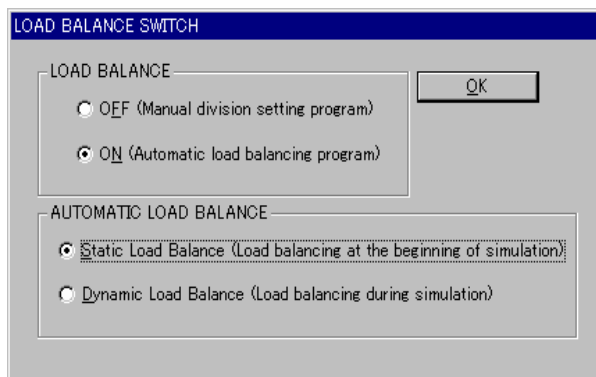
Domain decomposition

SPMD model



## Type of Parallel Program

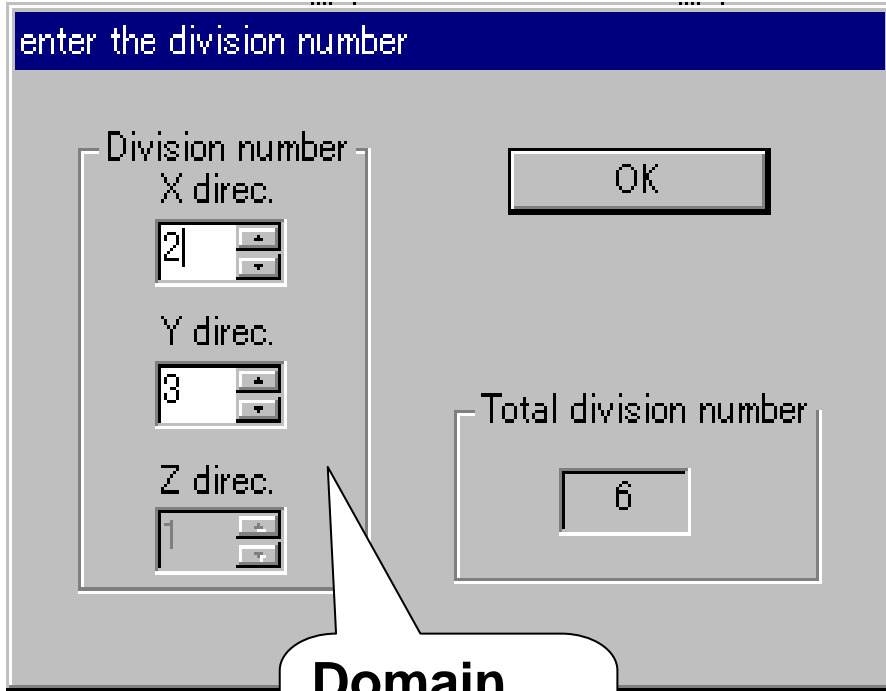
### Dynamic load balance



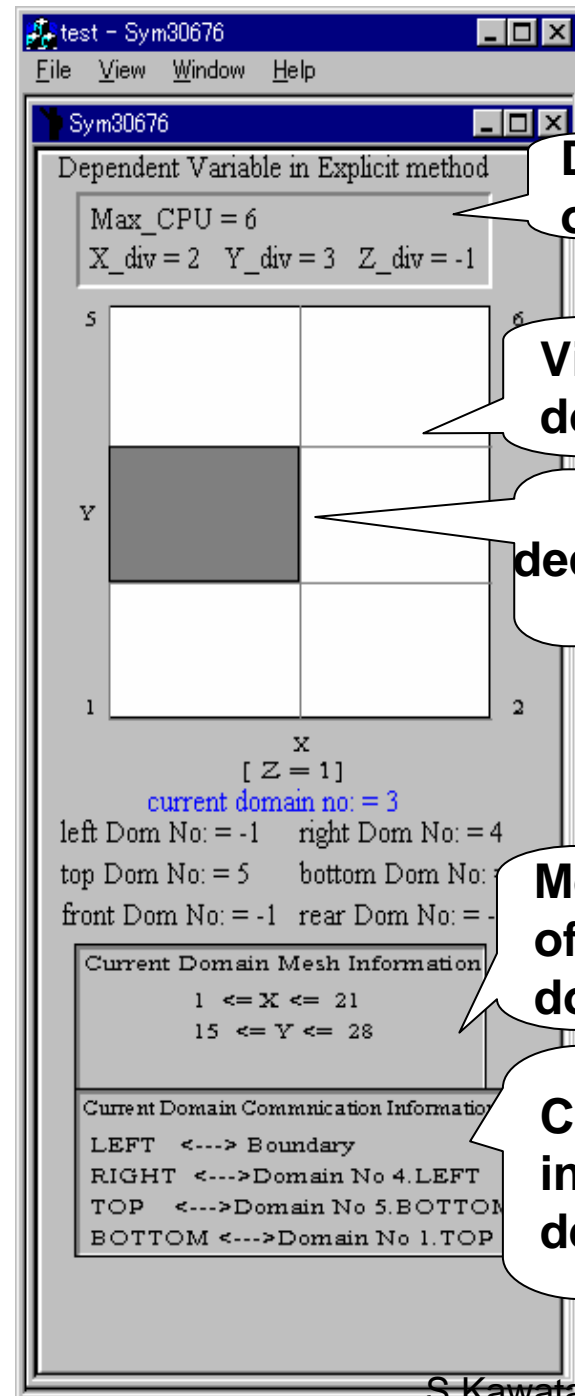
Back

# Parallel Program generation process

## Setting and Visualization the domain decomposition



**Domain decomposition control**



**Division number of each axis**

**Visualization of domain decomposition**

**Active decomposition domain**

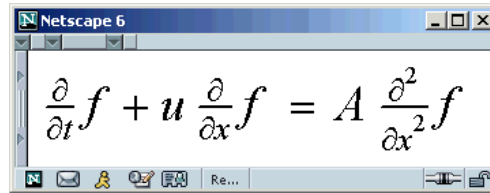
**Mesh information of Active decomposition domain**

**Communicating data information of Active decomposition domain**



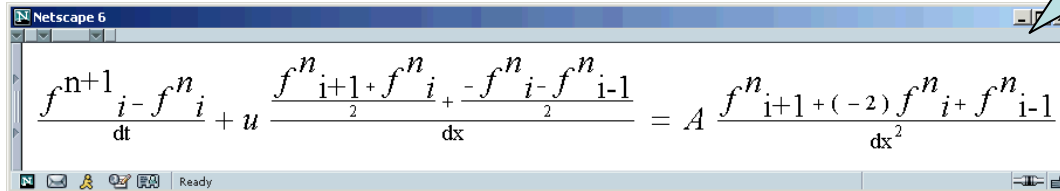
# Equation Manipulation Module

White Box  
System



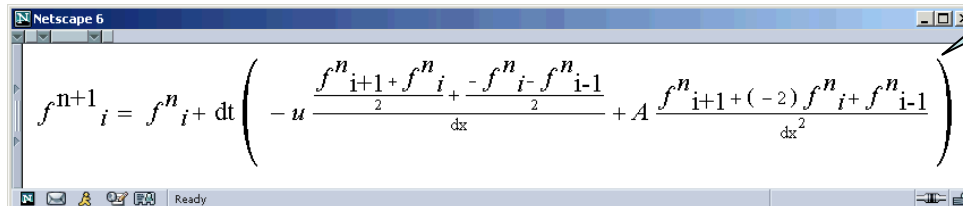
$$\frac{\partial f}{\partial t} f + u \frac{\partial f}{\partial x} f = A \frac{\partial^2 f}{\partial x^2} f$$

Discretization  
Equation



$$\frac{f^{n+1}_i - f^n_i}{dt} + u \frac{f^n_{i+1} + f^n_i - f^n_i - f^n_{i-1}}{2 dx} = A \frac{f^n_{i+1} + (-2)f^n_i + f^n_{i-1}}{dx^2}$$

Manipulation  
Equation



$$f^{n+1}_i = f^n_i + dt \left( -u \frac{f^n_{i+1} + f^n_i - f^n_i - f^n_{i-1}}{2 dx} + A \frac{f^n_{i+1} + (-2)f^n_i + f^n_{i-1}}{dx^2} \right)$$

# Program Generation Module

Problem Information

Comment

```
/* ----- Information Data -----
```

```
Subject : 一次元移流拡散方程式
```

```
date : 06/03
```

```
note : 有限差分法による一次元移流拡散方程式のシミュレーション解析
```

```
----- Information Data End -----
```

```
*/
```

```
#include<stdio.h> /* 標準入出力ライブラリ */
```

```
#include<math.h> /* 数学ライブラリ */
```

```
/* 空間のデータの宣言 */
```

```
#define IMAX 50 /* i方向のメッシュ数 */
```

```
#define JMAX 50 /* j方向のメッシュ数 */
```

```
#define dx 0.1 /* i方向のメッシュ幅 */
```

```
#define dy 0.1 /* j方向のメッシュ幅 */
```

```
/* 時間のデータの宣言 */
```

```
#define NMAX 1000 /* 最大時間ステップ数 */
```

```
#define dt 0.01 /* 時間ステップ幅 */
```

```
/* 関数の宣言 */
```

```
void calculate_Diffuse2AD(); /* Diffuse2ADの計算 */
```

```
void boundary_Diffuse2AD(); /* Diffuse2ADの境界条件 */
```

```
void writeData(); /* ファイル出力関数 */
```

```
void initial(); /* 初期値の計算 */
```

```
void calculateSpace(); /* 計算領域の計算 */
```

```
void nextTimeStep(); /* 次のタイムステップへ移行 */
```

```
/* Main関数 */
```

```
void main(){
```

```
    /* 定数値の計算 */
```

```
    /* 初期値の設定 */
```

```
    initial();
```

Indent

# Documentation Module

2次元ショックチューブ - Netscape

ファイル(E) 編集(E) 表示(V) ジャンプ(G) ブックマーク(B) ツール(T) ウィンドウ(W) ヘルプ(H)

file:///E:/MyDocuments/pse/1014\_docMaker/test07/index.html

概要 独立変数 従属変数 定数 数式

独立変数  
[t](#)  
[i](#)  
[i](#)

従属変数  
[d](#)  
[u](#)  
[v](#)  
[T](#)  
[p](#)  
[qi](#)  
[qi](#)

定数  
[Cv](#)  
[m](#)

従属変数情報

名前
d
u
v
T

数式

Density

Temperature

Pressure

SpecificHeat

VelocityU

VelocityV

LiquidViscosityU

LiquidViscosityV

Temperature

$$\frac{\partial}{\partial t} T + u \frac{\partial}{\partial x} T + v \frac{\partial}{\partial y} T = -\frac{p + q_i}{C_{vd}} \frac{\partial}{\partial x} u + \left( -\frac{p + q_i}{C_{vd}} \frac{\partial}{\partial y} v \right)$$

Density

$$\frac{\partial}{\partial t} \rho + u \frac{\partial}{\partial x} \rho + v \frac{\partial}{\partial y} \rho = -\frac{p + q_i}{C_{vd}} \frac{\partial}{\partial x} u + \left( -\frac{p + q_i}{C_{vd}} \frac{\partial}{\partial y} v \right)$$

離散化手法

項	指定された手法
左辺第1項	ForwardDifference
左辺第2項	CentralDifference
左辺第3項	CentralDifference
右辺第1項	CentralDifference
右辺第2項	CentralDifference

戻る

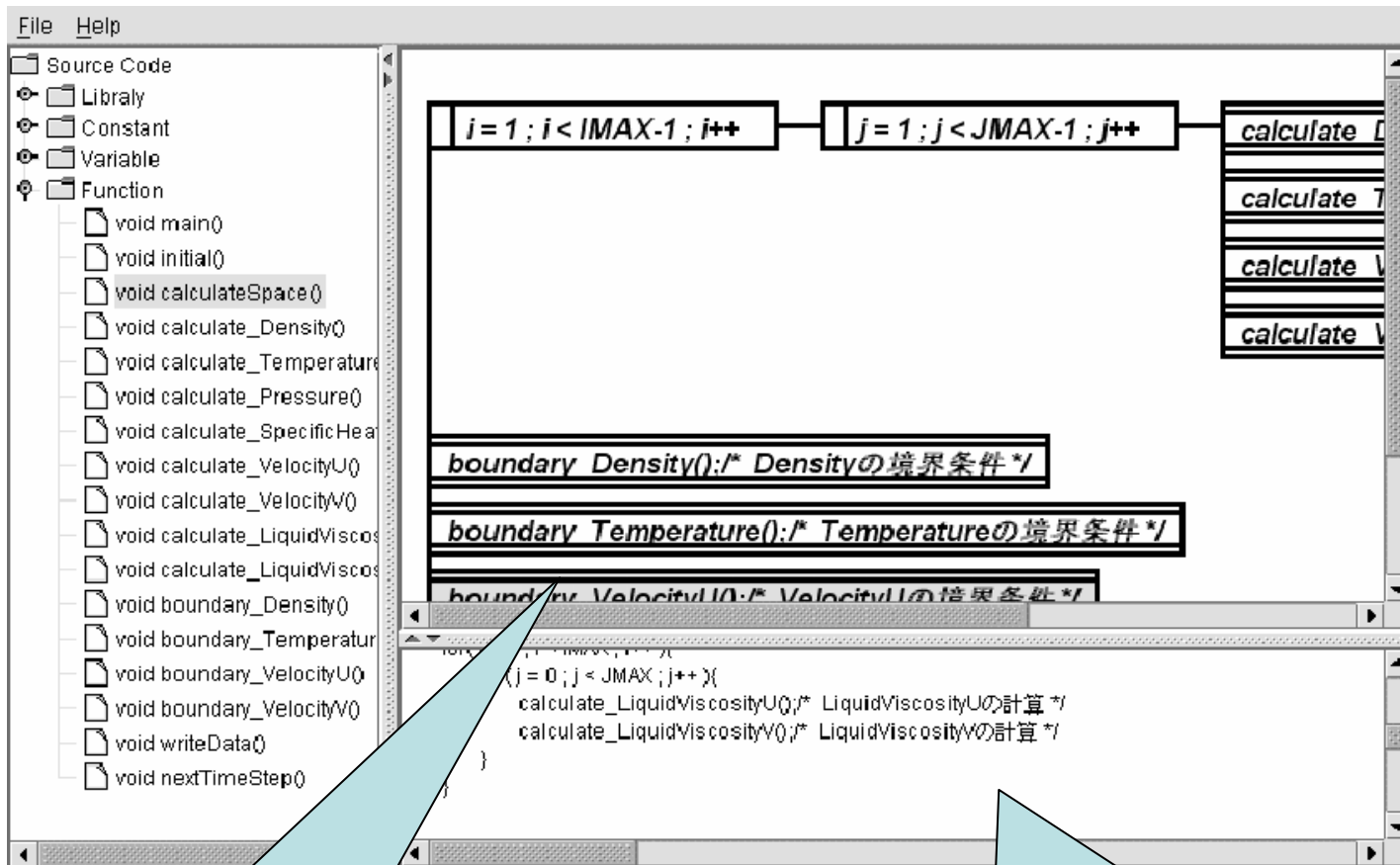
Detailed account of each variable

Information on discretization schemes that user specified

A list of all variables and that appear in the source code

Link to Caption

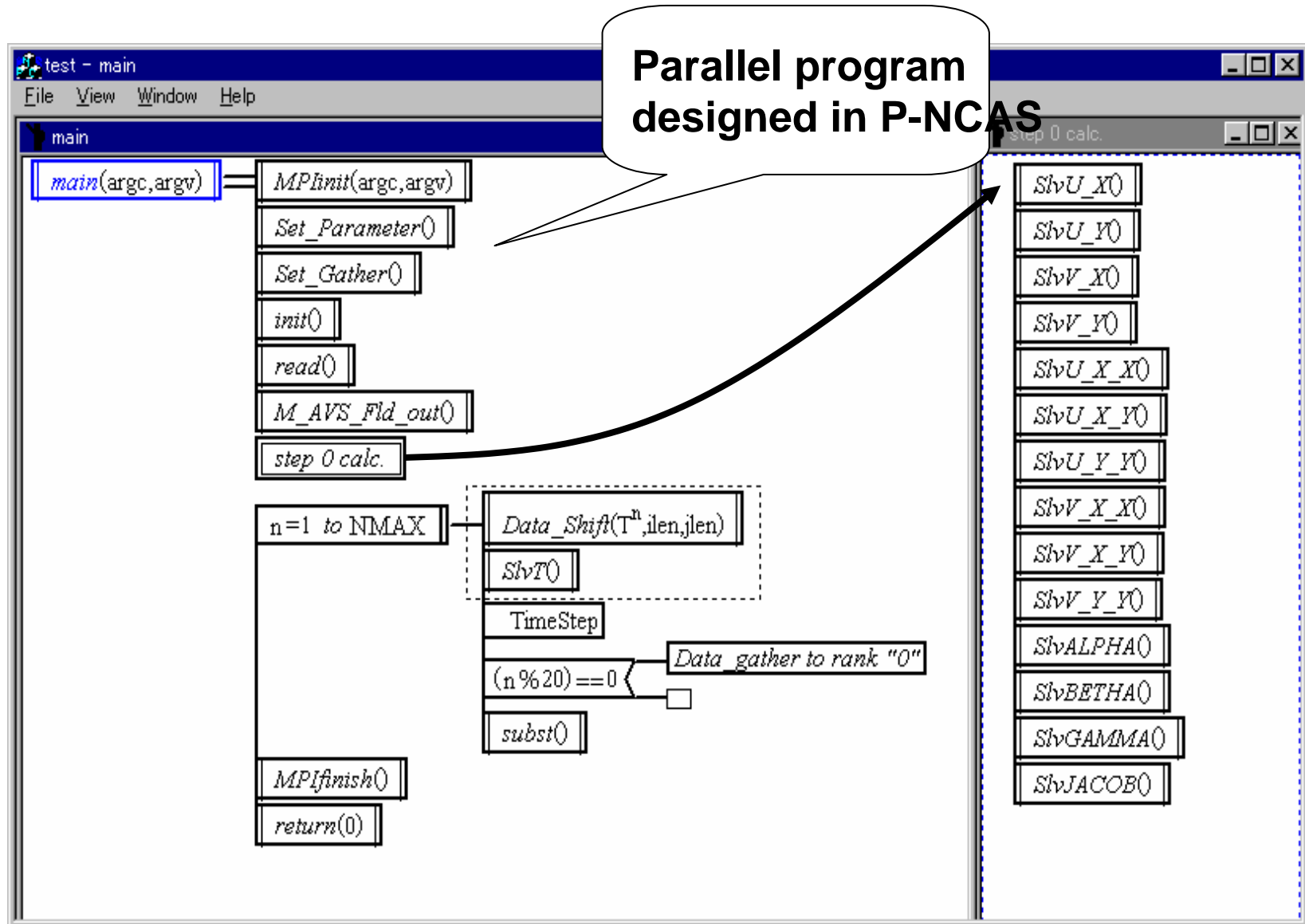
# Documentation Module



PAD : Program flow

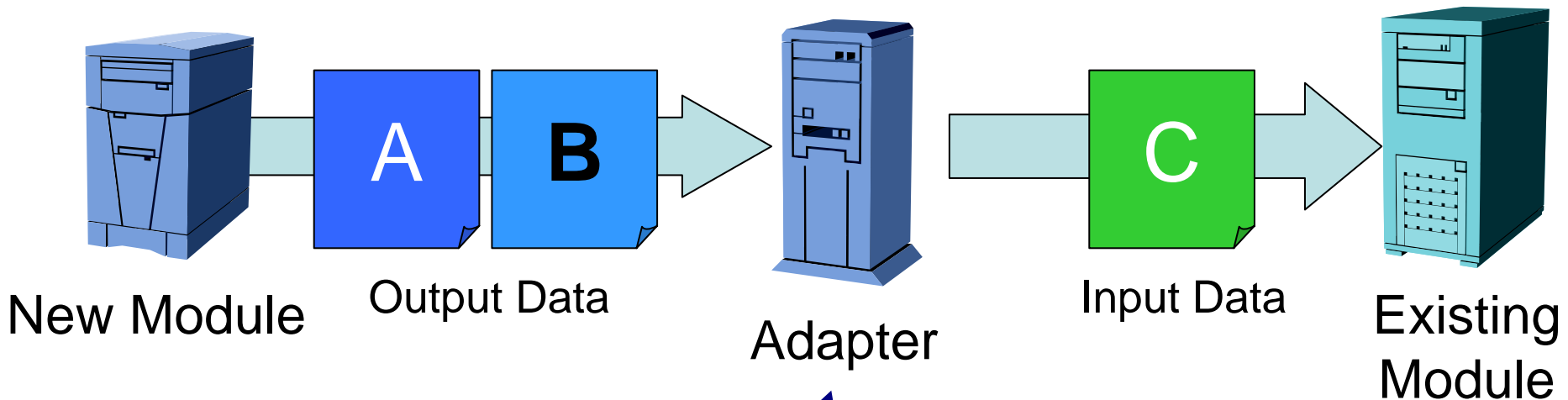
Generation Program

# Parallel Program generation process



# Module Liaison Module

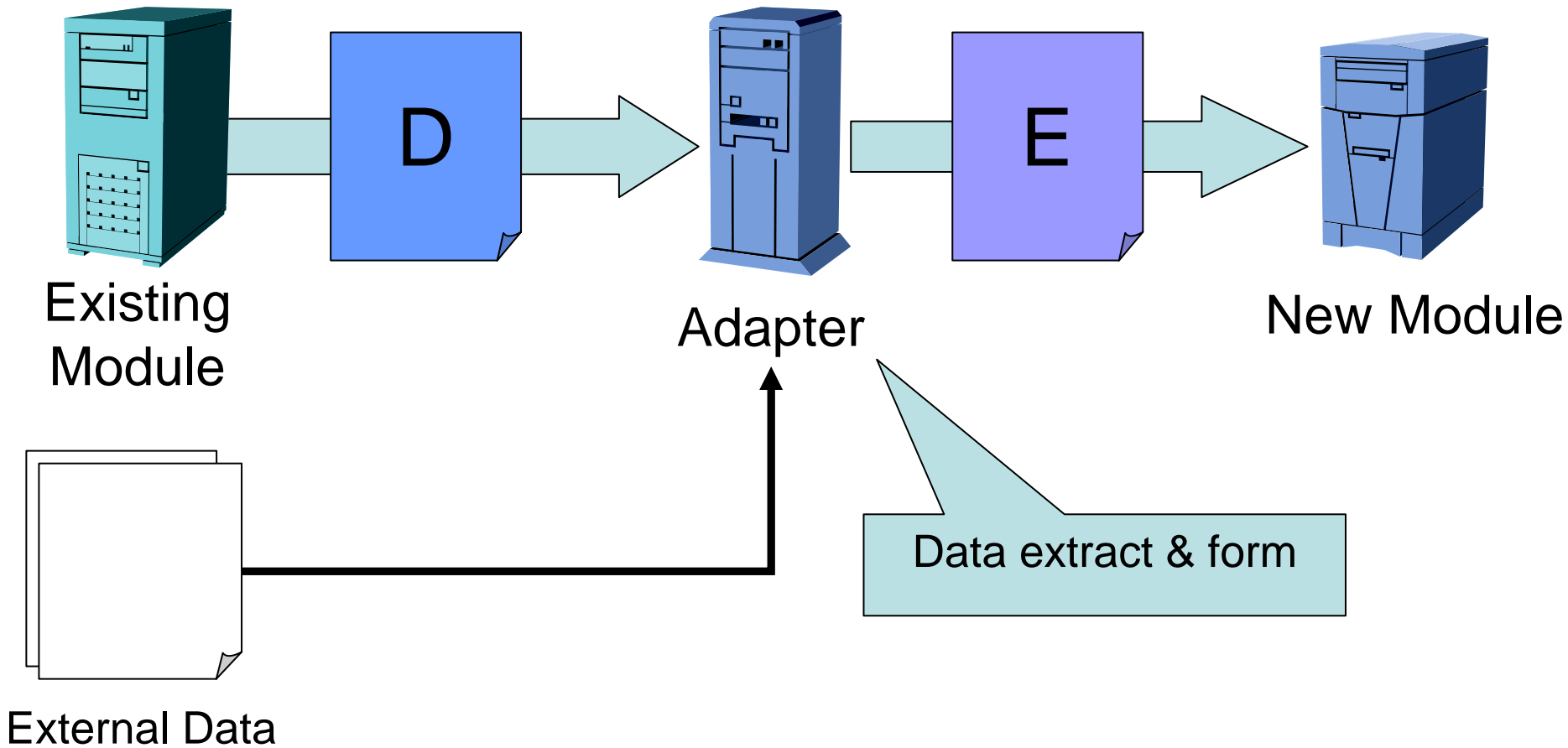
Output Data  $\neq$  Input Data



Generates an  
Adapter

**Conditions required:**  
Complete information for  
the input & output examples  
in XML

# Module Liaison Module



# Module Liaison Module

The screenshot displays the Module Liaison Module interface. The main window shows two XML data trees side-by-side. The left tree, titled 'equation.xml', contains a hierarchy of elements including 'equationInfo', 'math', 'dimension', 'name', 'type', 'MainParameter', 'MainParameterType', 'reln', and 'dimension'. The right tree, titled 'problemdata.xml', contains a hierarchy including 'problem', 'problemInfo', 'timeInfo', 'spaceInfo', 'characterInfo', and 'equationInfo'. A green arrow points from the 'equationInfo' element in the left tree to the 'equationInfo' element in the right tree, illustrating data association through drag and drop. Below the XML trees is a table with columns: No., XML File Name, XML File URL, Module Info, and XML Sample File URL. The table contains two rows of data. Below the table is a section labeled 'Right Area' with a similar table structure. A green box highlights the 'equationInfo' element in the right tree, and a green box highlights the 'Data Association: Drag & Drop' text.

**XML Data**

**Data Association: Drag & Drop**

**Input Sample Data**

**Output Sample Data**

No.	XML File Name	XML File URL	Module Info	XML Sample File URL
1	equation.xml	http://d-ncas00:8080/a...	http://d-ncas00:8080/d...	http://d-ncas00:8080/d...
2	problemdata.xml	http://d-ncas00:8080/a...	http://d-ncas00:8080/d...	http://d-ncas00:8080/d...

**Right Area**

No.	XML File Name	XML File URL	XML Sample File URL
1	discretizationInput.xml	http://d-ncas00:8080/a...	http://d-ncas00:8080/d...

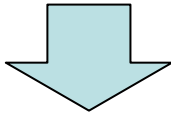
**Adapter generation information**



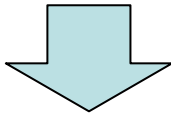
# Example

- 2D Shock Tube Problem

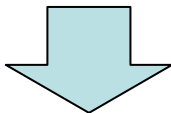
Problem Description (PDEs, Initial conditions,  
boundary conditions, ...)



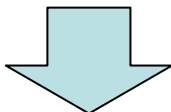
Equation Discretization (Discretization scheme : FDM)



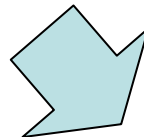
Equation Manipulation



Simulation Program Generation



Run & Visualization



Documentation

# Example

Generation Program

```

/* 関数の宣言 */
void calculate_Density();/* Densityの計算 */
void boundary_Density();/* Densityの境界条件 */
void calculate_Temperature();/* Temperatureの計算 */
void boundary_Temperature();/* Temperatureの境界条件 */
void calculate_Pressure();/* Pressureの計算 */
void calculate_SpecificHeat();/* SpecificHeatの計算 */
void calculate_VelocityU();/* VelocityUの計算 */
void boundary_VelocityU();/* VelocityUの境界条件 */
void calculate_VelocityV();/* VelocityVの計算 */
void boundary_VelocityV();/* VelocityVの境界条件 */
void calculate_LiquidViscosityU();/* LiquidViscosityUの計
*/
void calculate_LiquidViscosityV();/* LiquidViscosityVの計

```

```

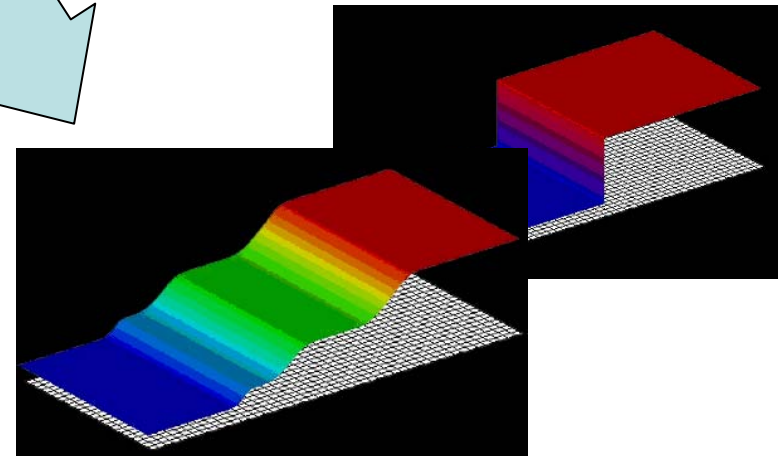
/* ----- Information Data -----
Subject : 2-D Shock Tube
date : Tue Feb 8 15:32:52 JST 2005
note : 有限差分法による二次元衝撃波問題
----- Information Data End -----
*/

#include<stdio.h>/* 標準入出力ライブラリ */
#include<math.h>/* 数学ライブラリ */

/* 空間のデータの宣言 */
#define IMAX 100/* i方向のメッシュ数 */
#define JMAX 50/* j方向のメッシュ数 */
#define dx 0.5/* i方向のメッシュ幅 */
#define dy 0.5/* j方向のメッシュ幅 */

```

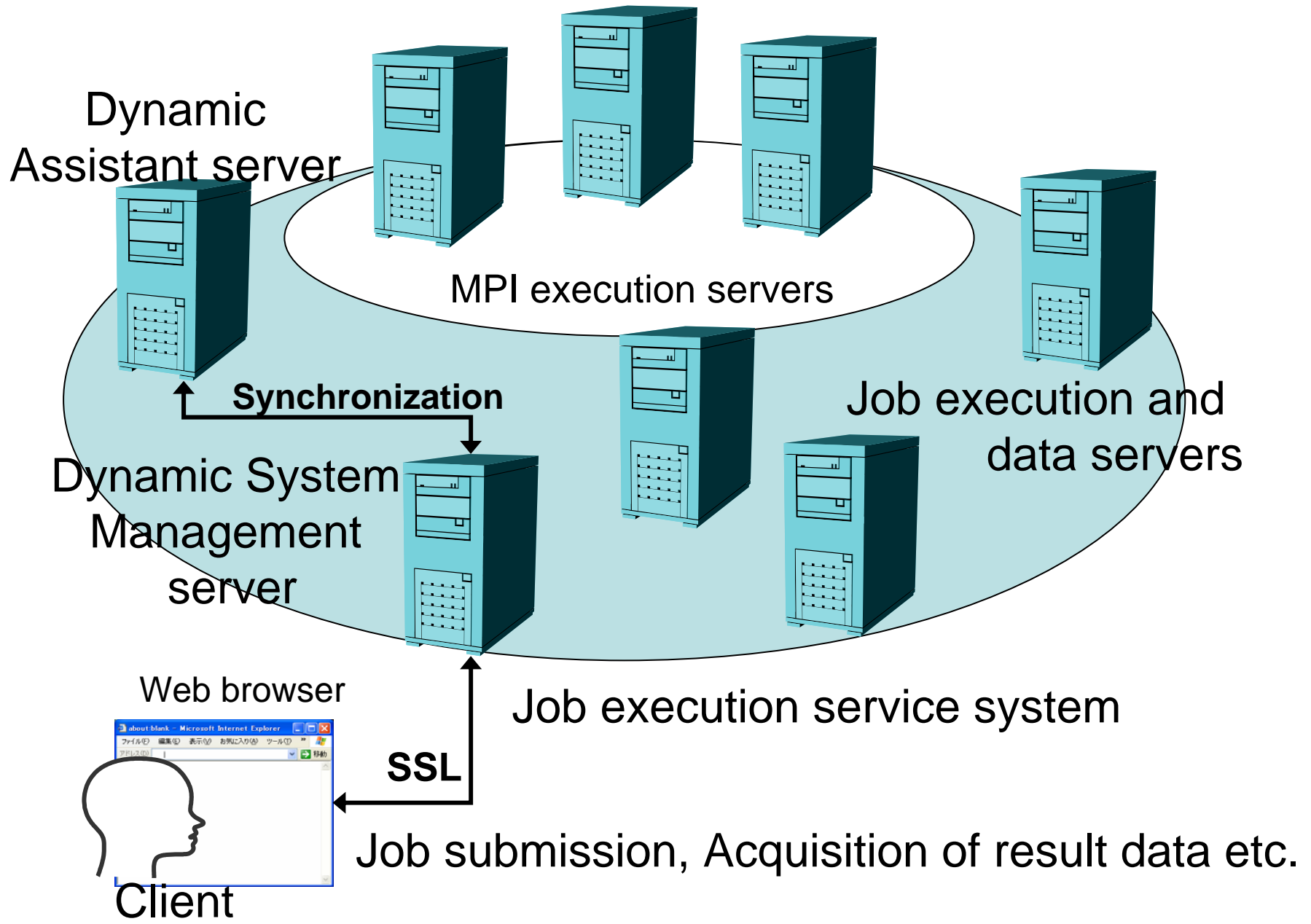
Run & Visualization



## Job Execution Service Module in our Distributed PSE of D-NCAS

- in Closed distributed computer environment
- Dynamic system management
- MPI job executable
- Robust data storage system - Result data duplication
- Service Construction support
  - Easy construction for users

# Job execution service system



## Dynamic system management

- Dynamic System Management server (DSM server) controls the job execution service system servers.
  - Is duplicated, and has backup server (Dynamic Assistant server: DA server)
  - When the DSM server meets some troubles, the DA server takes place of the DSM server.
- These two management servers move autonomously



The duplication of the management server provides a robust and fault tolerant system

# Dynamic system control

Management of dynamic servers

	Server name	Status
Dynamic management server	skk-s1	Starting
Dynamic assistance server	gridlab0	Starting

Dynamic management server stop (become static)

Dynamic management server restart

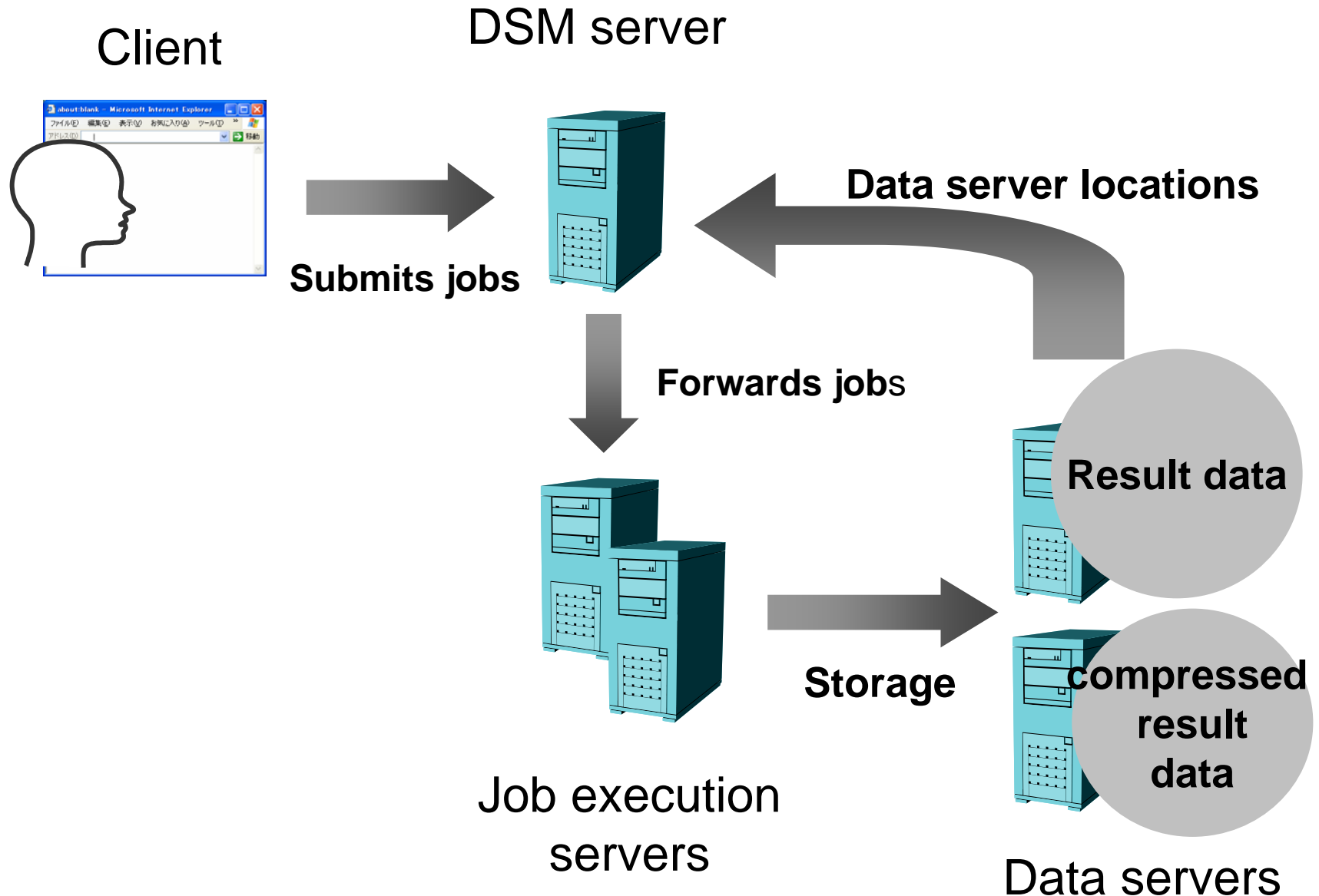
Dynamic assistance server stop (become static)

Dynamic assistance server restart

Server name  
Server status

The Administrator can control the dynamic system management.

# Result data duplication



# Job setting

Compile command  
& execution method

The screenshot shows a web browser window with the URL `https://skk-s1/~ias/cgi-bin/management/setting.cgi#`. The main form is titled "test.c" and includes the following fields:

- Make a new script for execution [make a new script](#)
- Comment:
- Compile command:
- Execution pass:
- Comment:

Below these fields is a "Select Servers" section with a table:

No.	check	Host name	CP
1	<input checked="" type="checkbox"/>	skk-s1	Pentium4--
2	<input checked="" type="checkbox"/>	itsc-9	Pentium4--
3	<input checked="" type="checkbox"/>	gridlab1	Celeron--
4	<input checked="" type="checkbox"/>	gridlab0	Celeron--
5	<input type="checkbox"/>	gridlab2	Not ava
6	<input type="checkbox"/>	waste01	Not ava
7	<input type="checkbox"/>	skk-c5	Not ava

Below the table is the "Setting of a save server" section:

- System dependence
- Specification of a save location
- Directory name:

At the bottom of the form is a "Setting and submit" button.

An open window titled "https://skk-s1 - source edit - Mozilla" shows a shell script for execution:

```

#!/bin/sh
mkdir dataOut
make
./a.out
  
```

Below the script is a "Save the content and close window" button.

Target servers for  
the job execution

Result data  
save setting

Shell script for  
the job execution



# MPI job setting

Mpi compile command  
& execution method

Master & slave  
server

Data output  
server

Compile command: mpicc  
Execution pass: mpirun -np 2 ./a.out  
Argument:

ID	Master	Slave	Data Output Server	Host name	CPU	memory	Use memory	Free memory	Load average	HDD free capacity
<b>gridlab</b>										
	<input checked="" type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	gridlab0	Celeron--2.02GHz	483.208MB	174.752MB	308.456MB	0.01	3.8GB
	<input checked="" type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	gridlab1	Celeron--2.02GHz	515.404MB	191.016MB	324.388MB	0.00	56GB
<b>skk</b>										
	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	itsc-9	Pentium4--2.387GHz	1034.04MB	751.316MB	282.724MB	0.00	179GB

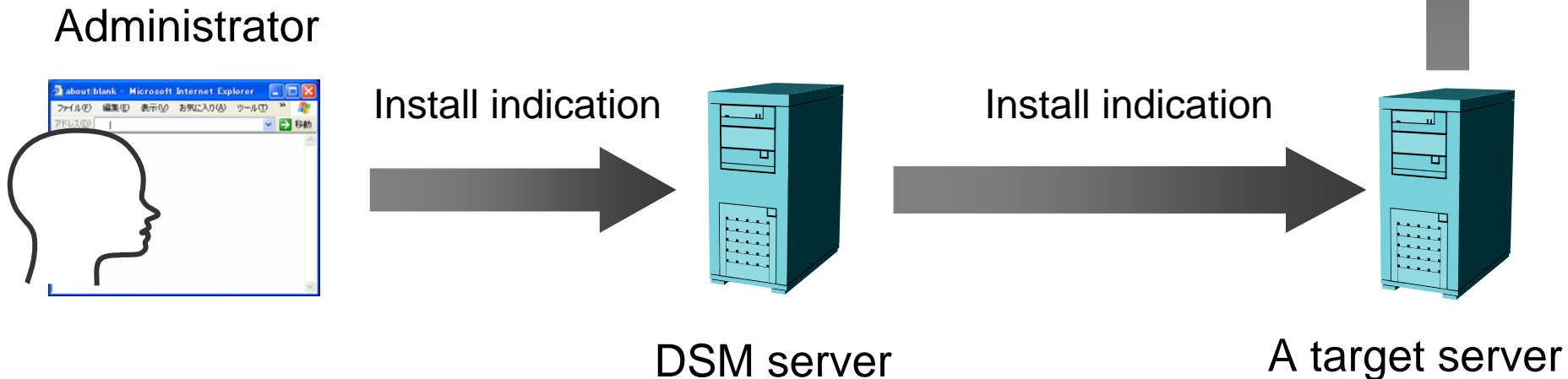
Setting of a save server  
 System dependence  
 Specification of a save location  
 Directory name:  :Absolute path

Setting and submit

Done

# RPM package installation support

- Supports RPM package installation
  - Administrator can install new compiler & RPM package
  - Administrator can use this function through the Web page of the system



# RPM package installation support

RPM Package Viewer

Server select:

RPM package name	Server	RPM package full name
gcc	gridlab0	gcc-4.0.0-8
gcc	gridlab1	gcc-3.4.2-6.fc3
gcc	skk-s1	gcc-3.4.2-6.fc3
gcc	waste01	gcc-3.4.3-22.fc3
j2sdk	skk-s1	j2sdk-1.4.2_08-fcs

Update•RPM Search

Update with YUM

System Authorization

Login:

Password:

Server:

RPM package name:

RPM package Search

Server:

RPM package name:

Done

RPM package that installed with the job execution service system tool

Input information for new RPM package installation

Installed package search in the system

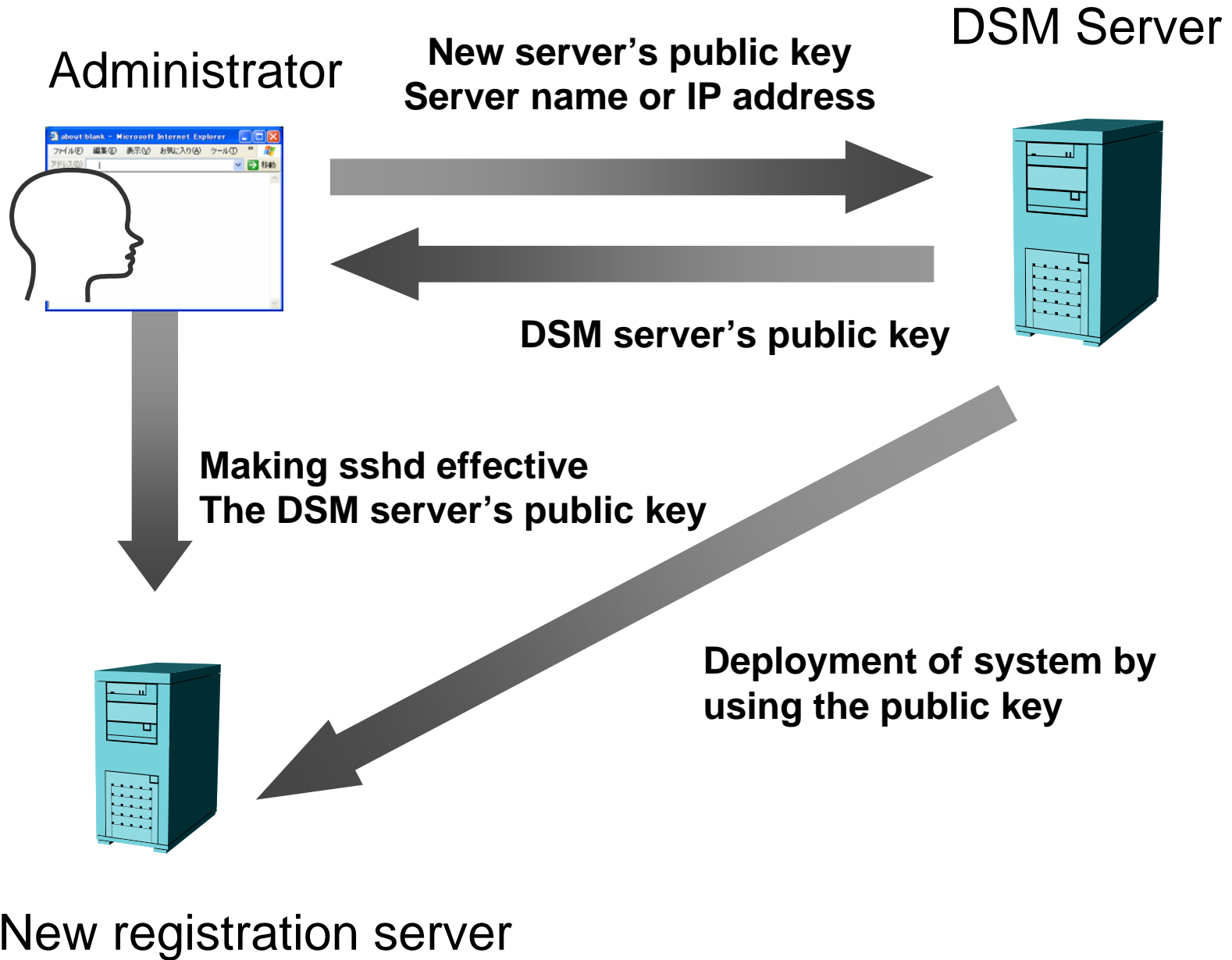
# System construction support

- Construction
  - Replaces a setting file of a Web server
  - Deploys composition files for job execution service system
- Registration
  - Notifies all servers the addition of the new servers



This function makes system construction and new servers addition

# System construction support



# Summary & Conclusions

- Development of Distributed Problem Solving Environment “D-NCAS”
  - Distributed System
    - Easy for Module & Functional addition
    - A Flexible System Construction
  - White Box System
- Standard Technology
  - SOAP, XML

# PSE: Newly Emerging Research Area

## PSE Japan Research group:

- <http://www.tsuyama-ct.ac.jp/teramoto/pse/>  
contact: [kwt@cc.utsunomiya-u.ac.jp](mailto:kwt@cc.utsunomiya-u.ac.jp)

- 1st PSE Workshop 2005 (Dec. 2005)

May have the next PSE Workshop?

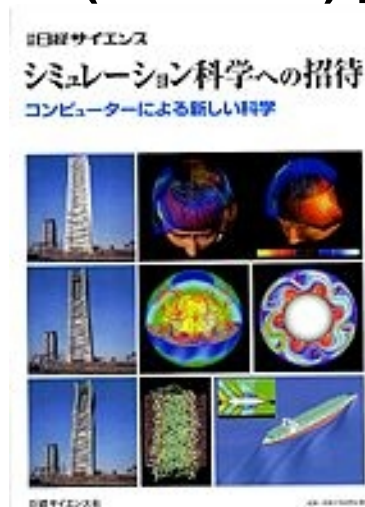
- 8th PSE Workshop in Japan (Sept., 2005)

- 「PSE Book」s

/edited by Houstis, et al.

/PSE Book (in Japanese) Nikkei science

/PSE Book 2005 (Baifukan) [Basics], [Application]



培風館(2005)3月18日出版

# FUTURE Works in PSE

- Programming Free Environment
- Intelligent PSE, including Knowledge Discovery
- Collaboration support PSE
- Network-linked Distributed PSE
- Large Coupled Simulation support & Large data Handling
  
- PSE for PSE may be required
  - PSE for Documentation
  - PSE for Modeling
  - Advanced PSE for EDUCATION Purpose
  
- Domain specific PSEs --- Economics, Bio/Medical purposes, Manufacturing management, Human resource management, Welfare support, Education, Energy Problem, ... more



# Summary & Conclusions

**Intelligent + Network-linked  
Programming-free Environment  
→ Excellent & Productive Tools !**

**PSEs => Infrastructure in IT**

**PSEs may bring us Dreams!**