NArray and scientific computing with Ruby

田中昌宏
Masahiro Tanaka

自己紹介 Self Introduction

- NArrayの開発者
  - NArray developer
- 昨年度からつくば市在住
  - I live in Tsukuba since Apr 2009

内容 Contents

- NArrayを使ったサイエンス
  - Science using NArray
- NArrayの概要
  - NArray overview
  - 詳しい話はしません。No details.
- 並列分散ワークフロー実行システム
  - Parallel Distributed Workflow System

NArrayを使ったサイエンス
Science using NArray

地球物理学
Geophysics

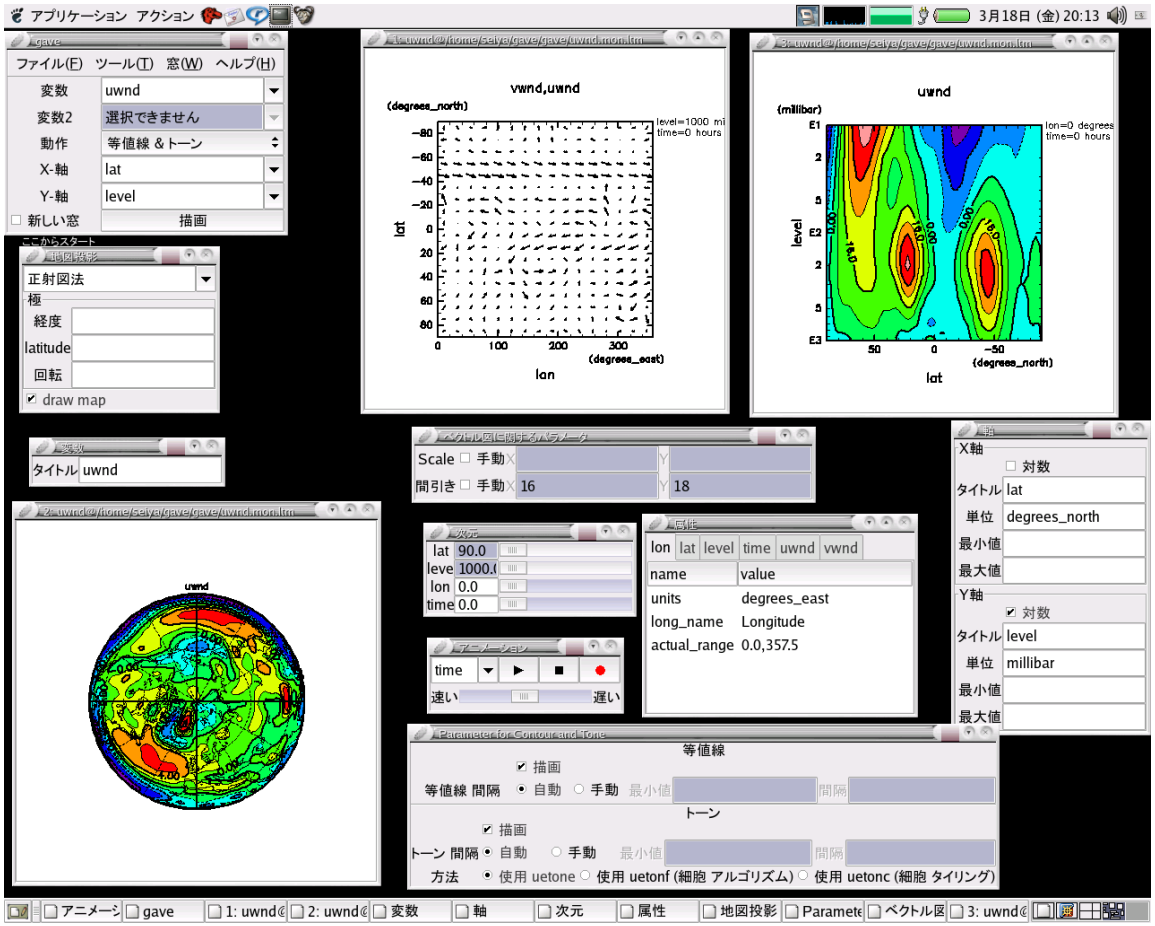
地球流体電脳倶楽部
Geophysical Fluid Dynamics Dennou-Club

- http://www.gfd-dennou.org/

- GPhys
  - 多次元物理量表現ライブラリ
  - multi-dimensional physical quantity library
- RubyDCL
  - グラフ描画ライブラリ
  - Plotting Graph
- RubyNetCDF
  - 物理データフォーマットNetCDFのIOライブラリ
  - Physical data format library

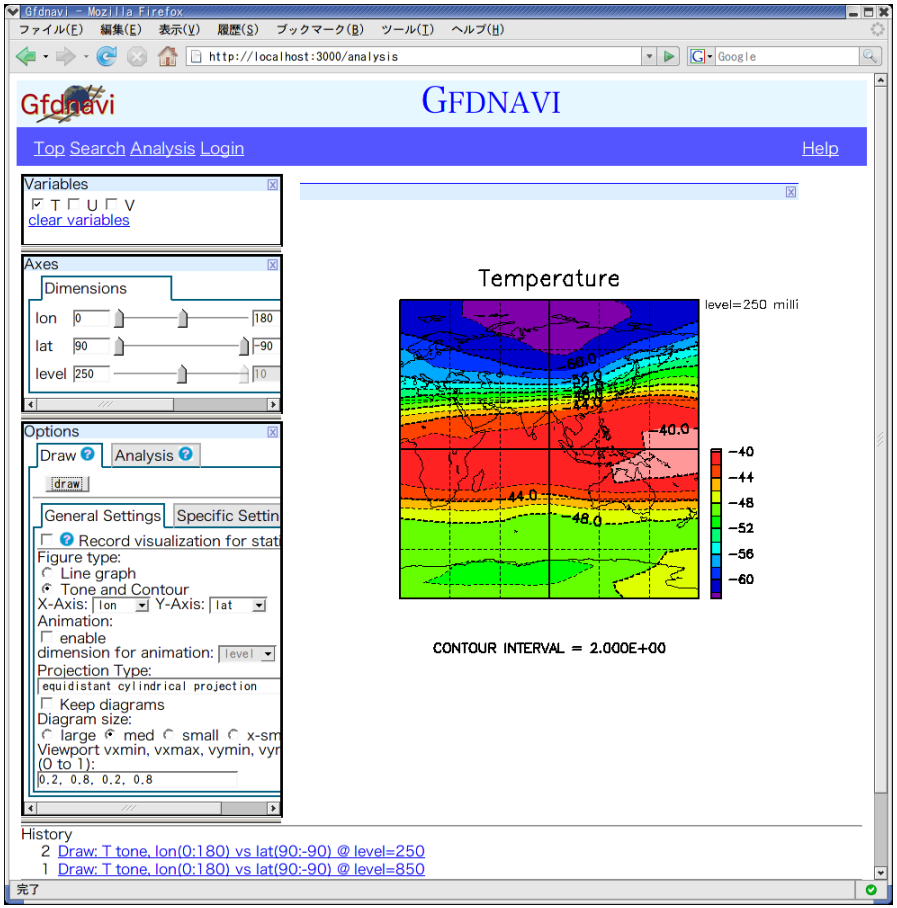
gave

- Gtk ベース格子点可視化・解析ツール
- Gtk-base visualize & analysis tool for lattice data



Gfdnavi

- Railsベースの地球流体データベース・解析・可視化ツール
- Rails-base visualize & analysis tool for Geophysical database



物質科学
Material Science

Virtual Sample Library

- 物質科学における、多次元試料データの管理を効率的に行う
- For managing multi-dimensional sample data in material science.

INSTITUTE OF PHYSICS PUBLISHING MEASUREMENT SCIENCE AND TECHNOLOGY
Meas. Sci. Technol. **16** (2005) 285–291 doi:10.1088/0957-0233/16/1/037

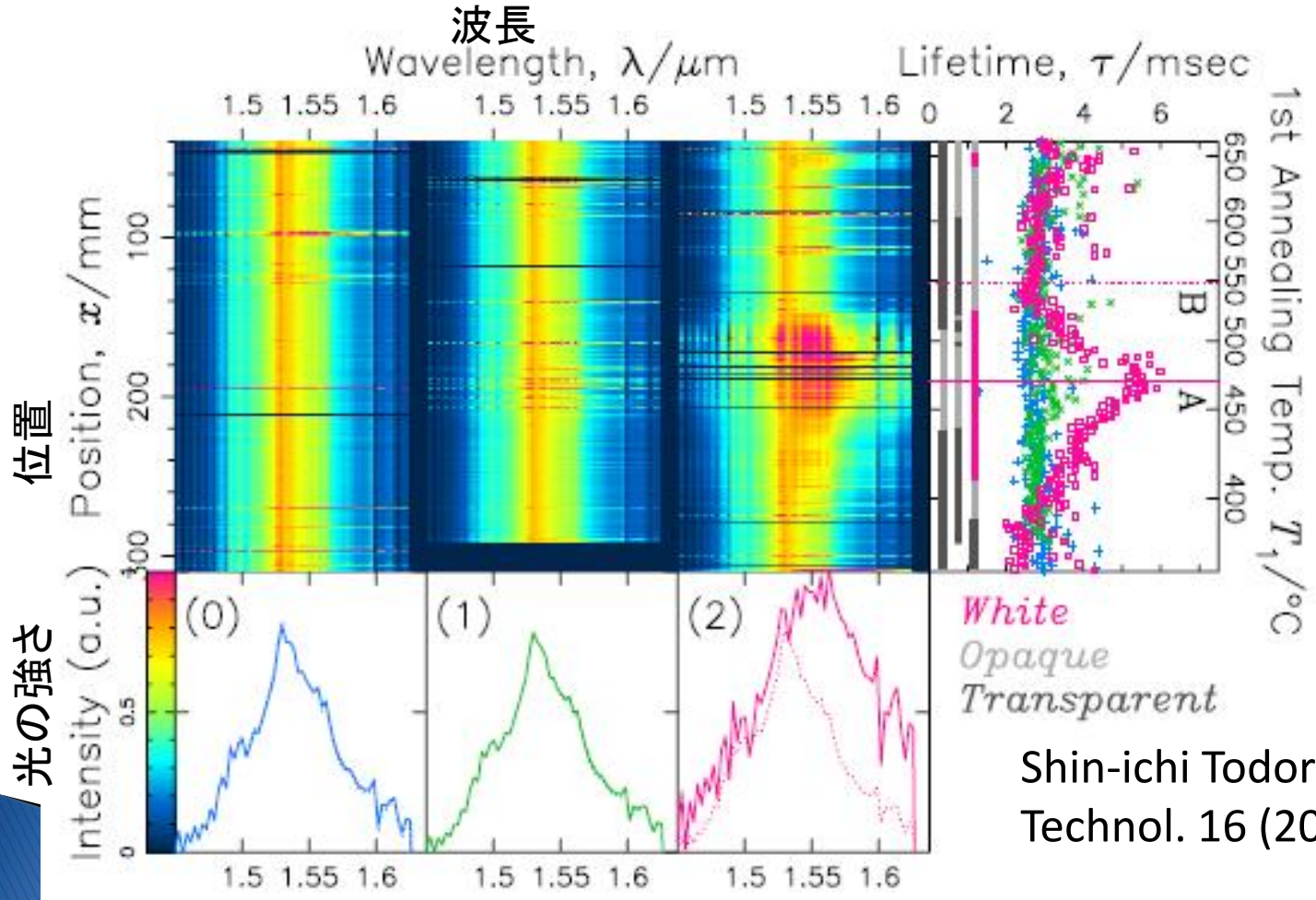
Object-oriented virtual sample library: a container of multi-dimensional data for acquisition, visualization and sharing

Shin-ichi Todoroki

Advanced Materials Laboratory, National Institute for Materials Science, Namiki 1-1, Tsukuba, Ibaraki 305-0044, Japan

E-mail: todoroki.shin-ichi@nims.go.jp

エルビウムの蛍光スペクトル
Fluorescence spectra of Er^{3+}



Shin-ichi Todoroki, Meas. Sci. Technol. 16 (2005) 285–291

分子生物学
Molecular Biology

Ruby-Helix

Ruby-Helix - Mozilla Firefox

ファイル(F) 編集(E) 表示(V) 履歴(S) ブックマーク(B) ツール(T) ヘルプ(H)

Ruby−Helix

Masahide Kikkawa

ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Journal of Structural Biology 157 (2007) 95–105

Journal of Structural Biology

www.elsevier.com/locate/yjsbi

Ruby-Helix: An implementation of helical image processing based on object-oriented scripting language

Zoltan Metlagel [a], Yayoi S. Kikkawa [b], Masahide Kikkawa [a,*]

[a] *Department of Cell Biology, University of Texas, Southwestern Medical Center, 5323 Harry Hines Boulevard, Dallas, TX 75390-9039, USA*
[b] *Department of Otolaryngology-Head and Neck Surgery, University of Texas, Southwestern Medical Center, 5323 Harry Hines Boulevard, Dallas, TX 75390-9039, USA*

Received 4 April 2006; received in revised form 20 June 2006; accepted 5 July 2006
Available online 15 August 2006

Abstract

Helical image analysis in combination with electron microscopy has been used to study three-dimensional structures of various biological filaments or tubes, such as microtubules, actin filaments, and bacterial flagella. A number of packages have been developed to carry out helical image analysis. Some biological specimens, however, have a symmetry break (seam) in their three-dimensional structure, even though their subunits are mostly arranged in a helical manner. We refer to these objects as "asymmetric helices". All the existing packages are designed for helically symmetric specimens, and do not allow analysis of asymmetric helical objects, such as microtubules with seams. Here, we describe Ruby-Helix, a new set of programs for the analysis of "helical" objects with or without a seam. Ruby-Helix is built on top of the Ruby programming language and is the first implementation of asymmetric helical reconstruction for practical image analysis. It also allows easier and semi-automated analysis, performing iterative unbending and accurate determination of the repeat length. As a result, Ruby-Helix enables us to analyze motor–microtubule complexes with higher throughput to higher resolution.

Keywords: Cryo-electron microscopy; Helical image analysis; Asymmetric helical reconstruction; Ruby-Helix

Ruby-Helix

- 電子顕微鏡写真から、生体微小管などの三次元構造を構築するためのプログラム
- Program set for Helical image analysis in combination with electron microscopy used to study three-dimensional structures of microtubules etc.

The EMBO Journal (2004) 23, 2459–2467 | © 2004 European Molecular Biology Organization | All Rights Reserved 0261-4189/04
www.embojournal.org

THE EMBO JOURNAL

Dynein and kinesin share an overlapping microtubule-binding site

Naoko Mizuno[1,2], Shiori Toba[2], Masaki Edamatsu[2], Junko Watai-Nishii[2], Nobutaka Hirokawa[3], Yoko Y Toyoshima[2] and Masahide Kikkawa[1,*]

電子顕微鏡写真
Electron Microscope image

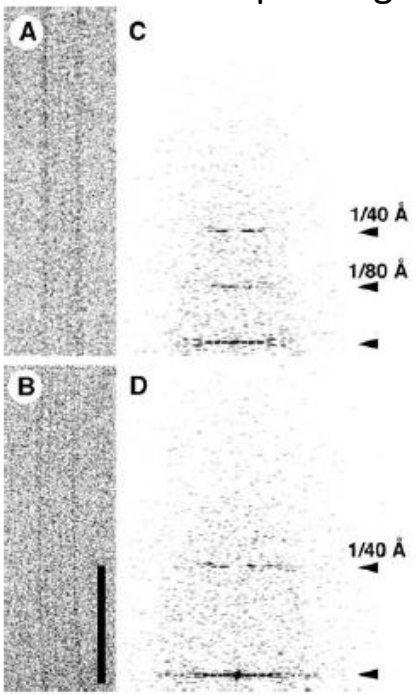
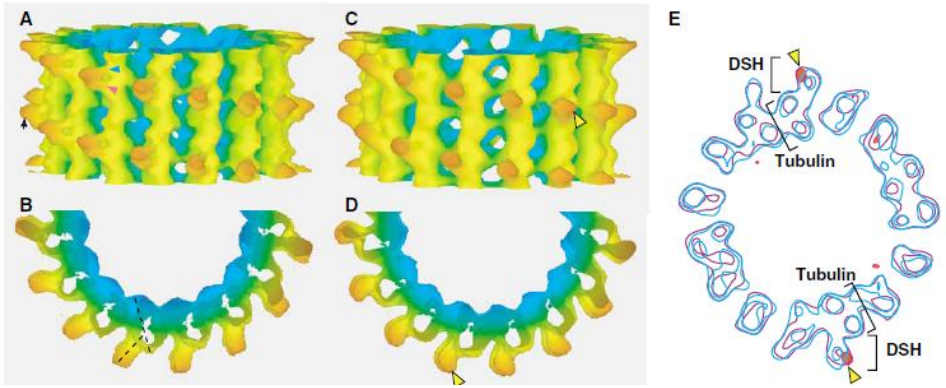


Figure 3 Cryo-EM images of a DSH–microtubule complex. Cryo-EM images (bar = 800 Å) showing (**A**) a DSH–microtubule complex and (**B**) an undecorated microtubule. (**C**, **D**) Computed diffraction patterns of the images.

画像から構築した3次元構造
3-D structure constructed from the EM image

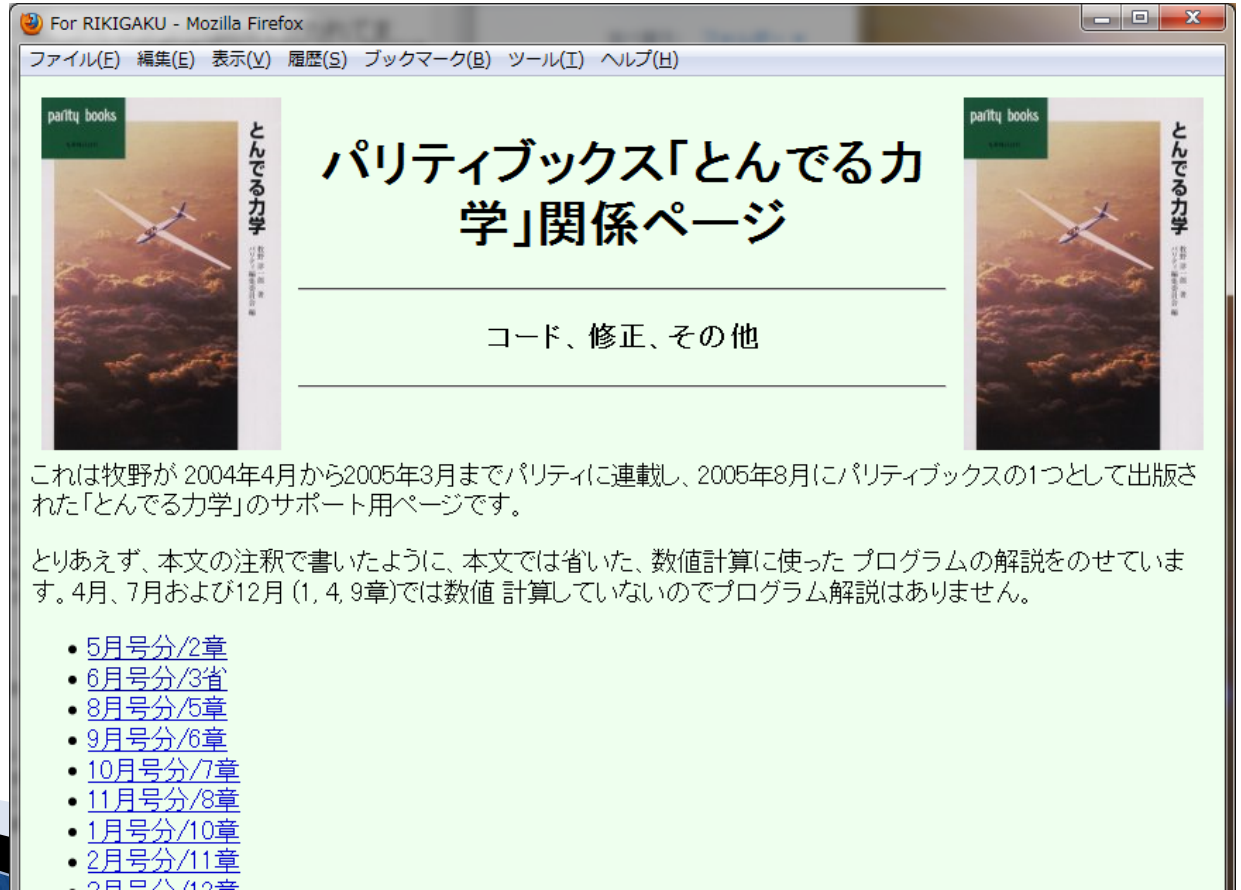


manually adjusted using O (Jones *et al*, 1991). Most of the data analyses were carried out using the newly developed Ruby-Helix scripting system, which we will describe in detail elsewhere. The

力学
Dynamics

とんでる力学
Aerodynamics

- http://www.artcompsci.org/~makino/rikigaku/
- A book in dynamics for general readers



Program and explanation

File: .may.cp - Mozilla Firefox

ファイル(F) 編集(E) 表示(V) 履歴(S) ブックマーク(B) ツール(T) ヘルプ(H)

1.「とんでる力学」5 月分

1.1. プログラムと解説

赤木: というわけで、ここではなんか、本文でしなかった、微分方程式を解くプログラムはどんなのかって話をするんだけど。

学生S: えーと、そうですね。図3 のグラフをどうやって書いたかです ね。とりあえずプログラム出してみましょうか。

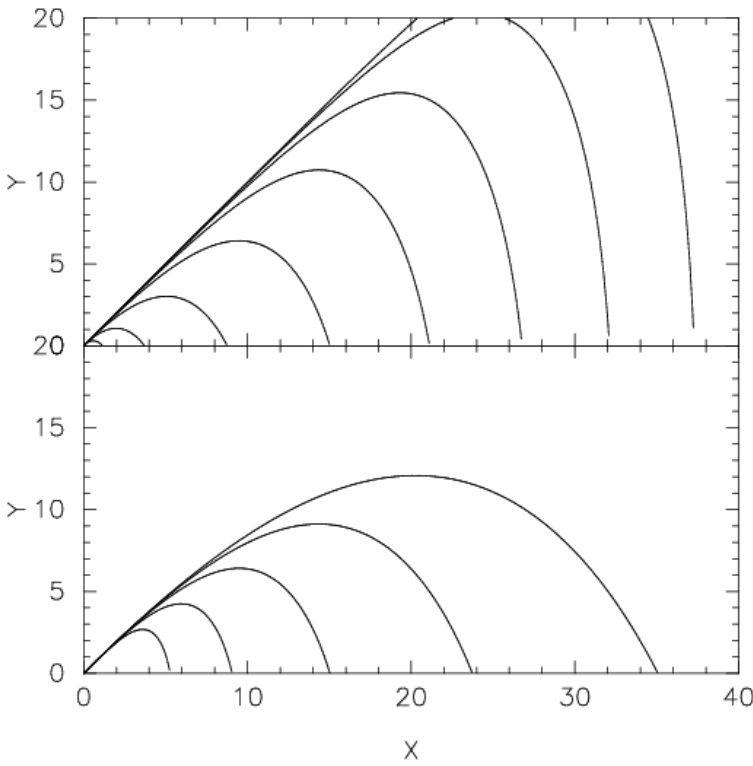
---

may-fig3.rb

```
require 'narray'
include Math
$g=-9.8

def dx(x,c)
  d = NArray.float(4)
  d[0]=x[2]
  d[1]=x[3]
  vabs = sqrt(x[2]*x[2]+x[3]*x[3])
  d[2]= -c*x[2]*vabs
  d[3]= -c*x[3]*vabs + $g
  d
end

def rk2(x,c,dt)
  d = dx(x,c)*0.5*dt
  x += dx(x+d,c)*dt
  x
end

def plotsolution(vx,vy,c,dt)
  x = NArray.float(4)
  x[0..3]=[0,0,vx,vy]
  prev=x[0..1]
  reloc x[0] x[1]
```

飛跡の計算結果

Result flight trajectory



NArrayを使っていただいて、ありがとうございます
Thank you for using NArray

なぜRubyでデータ解析をしたいか？
Why do we want to analyze data using Ruby

データ処理を簡単にしたい
We want to make data processing easy

- 処理を簡単に書きたい
  - write processing codes shortly
- 結果をすぐ確認したい
  - confirm processing result immediately
- インタラクティブな操作もしたい
  - manipulate interactively

科学計算に必要だと思う機能
Requirement from scientific computing

- プログラムが書ける – **Ruby**
  - Programmable
- インタラクティブな実行 – **irb**
  - Interactive execution
- データ配列操作 – **NArray**
  - Data Array Manipulation
- 数値計算ライブラリ
  - Library for Numerical Algorithms
- グラフ表示機能
  - Plotting Graphs

Ruby/NArray

- 多次元数値配列
  - Numerical N-dimensional Array
- http://narray.rubyforge.org
- http://github.com/masa16/narray

リリース
Release

- 0.3.0 – 1999
- 0.5.0 – 2000
- 0.5.9p7 – 2009（current version）
- 0.7.0 – 20??（next version, under development）

NArrayのモデル

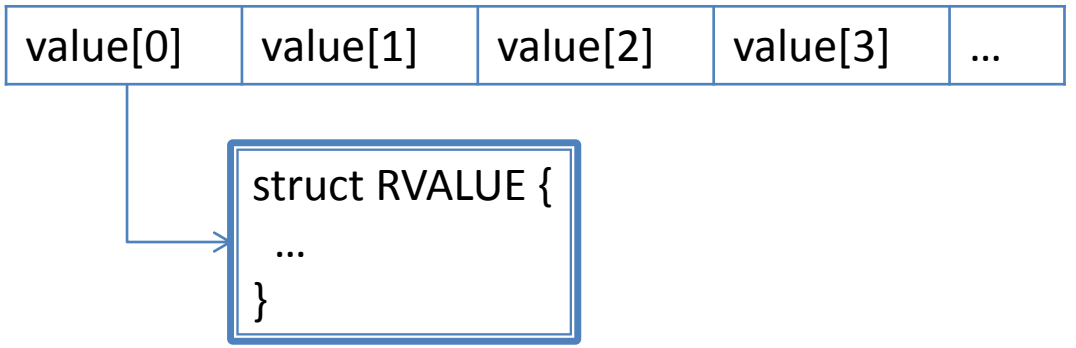
- IDL (Interactive Data Language)
- Yorick
- Python Numeric

NArray 構成要素 (基本部分)
NArray constituent (basic part)

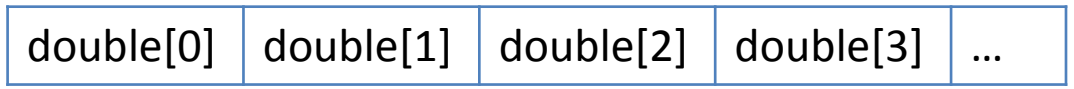
- rank
  - 次元数 the number of dimension
- shape
  - 配列の「形」 array shape
- type
  - 要素の型 element type
- memory block
  - データを格納するメモリーブロック

メモリーブロック
Memory block

- Ruby Array :
  - VALUE *ptr; →



- NArray
  - void *ptr; →



  - or

要素の型
Element types

- 整数
  - BYTE (8bit)
  - SINT (16bit)
  - INT (32bit)
- 浮動小数点数
  - SFLOAT (32bit)
  - FLOAT
  - = DFLOAT (64bit)
- 複素数
  - SCOMPLEX (64bit)
  - COMPLEX
  - = DCOMPLEX (128bit)
- Rubyオブジェクト
  - ROBJECT

NArray types

- 現版では、ビルトインのみ
  - Current version has Built-in types only
- 次版 Next version
  - 後から付け足すことが可能
    - Append-able types

Shape

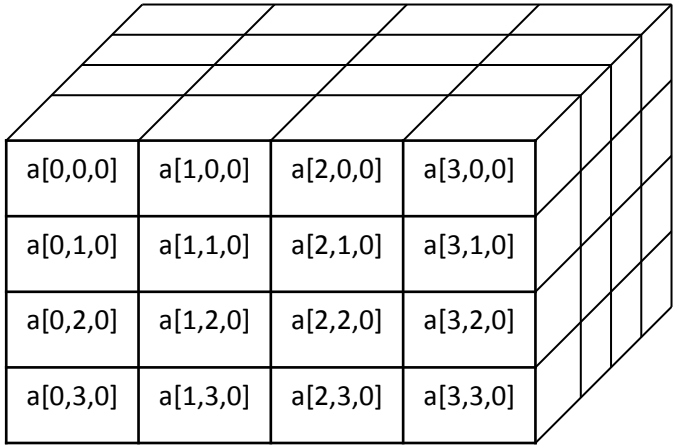
- shape = [4]
  - 1次元配列
  - one-dimensional array

a[0]	a[1]	a[2]	a[3]

- shape = [4,4]
  - 2次元配列
  - two-dimensional array

a[0,0]	a[1,0]	a[2,0]	a[3,0]
a[0,1]	a[1,1]	a[2,1]	a[3,1]
a[0,2]	a[1,2]	a[2,2]	a[3,2]
a[0,3]	a[1,3]	a[2,3]	a[3,3]

- shape = [4,4,4]
  - 3次元配列
  - three-dimensional array

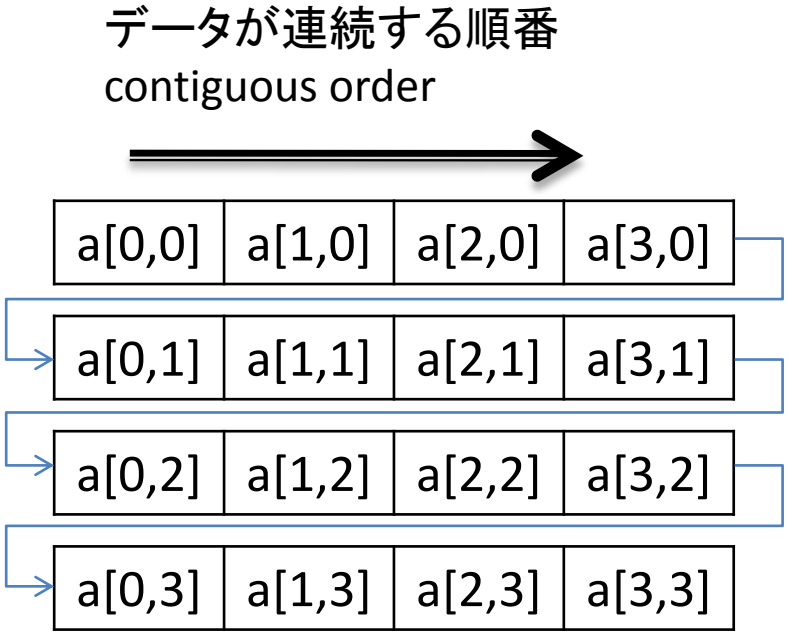


行列でいう次元とは異なる
Different from Matrix dimension

多次元データの順序
Order of Multi-dimensional data

- `a[i,j]`
- 左の次元(`i`)がはやく回る
  - Left dimension(`i`) is contiguous
  - Fortran-order
  - Row-major order

```
If a.shape == [m,n] then
  a[i,j] == a[i+j*m]
```



- `a[j][i]` と逆 reverse order
- 次版では変更予定
  - This will be changed for the next version.

NArrayの生成
Creating NArray

```
NArray.float(3,2)
=> NArray.float(3,2):
[ [ 0.0, 0.0, 0.0 ],
  [ 0.0, 0.0, 0.0 ] ]
```

```
NArray.float(3,2).indgen!
=> NArray.float(3,2):
[ [ 0.0, 1.0, 2.0 ],
  [ 3.0, 4.0, 5.0 ] ]
```

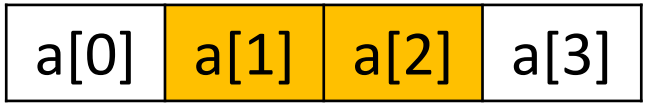
```
NArray[[1,2,3],[4,5,6]]
=> NArray.int(3,2):
[ [ 1, 2, 3 ],
  [ 4, 5, 6 ] ]
```

```
NArray[1..10]
=> NArray.int(10):
[ 1, 2, 3, 4, 5, 6, 7, 8,
  9, 10 ]
```

配列のスライス
Slice of NArray

- `a[1]` ----------
  - 要素参照
  - reference by element

- `a[1..2]` ----------
  - 範囲参照
  - reference by range



- `a[[1,3]]` ----------
  - インデックス配列参照
  - reference by index array

演算
Operations

- 演算は要素ごと
  - Operation is element-wise

a * b =

a[0]*b[0]	a[1]*b[1]	a[2]*b[2]	a[3]*b[3]

- 演算

  Operations
  - 四則演算
    - arithmetic operations
    - +, -, *, /, %, **
  - 数学関数演算
    - math operations
    - NMath module

NArray speed

- Float Array with 10^6 elements
- Ruby 1.9.2 :

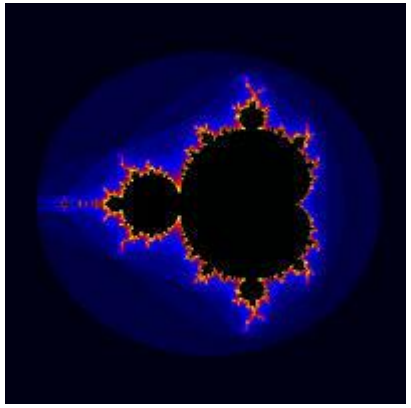
```
(0...n).map{|i| a[i]*b[i]}
```

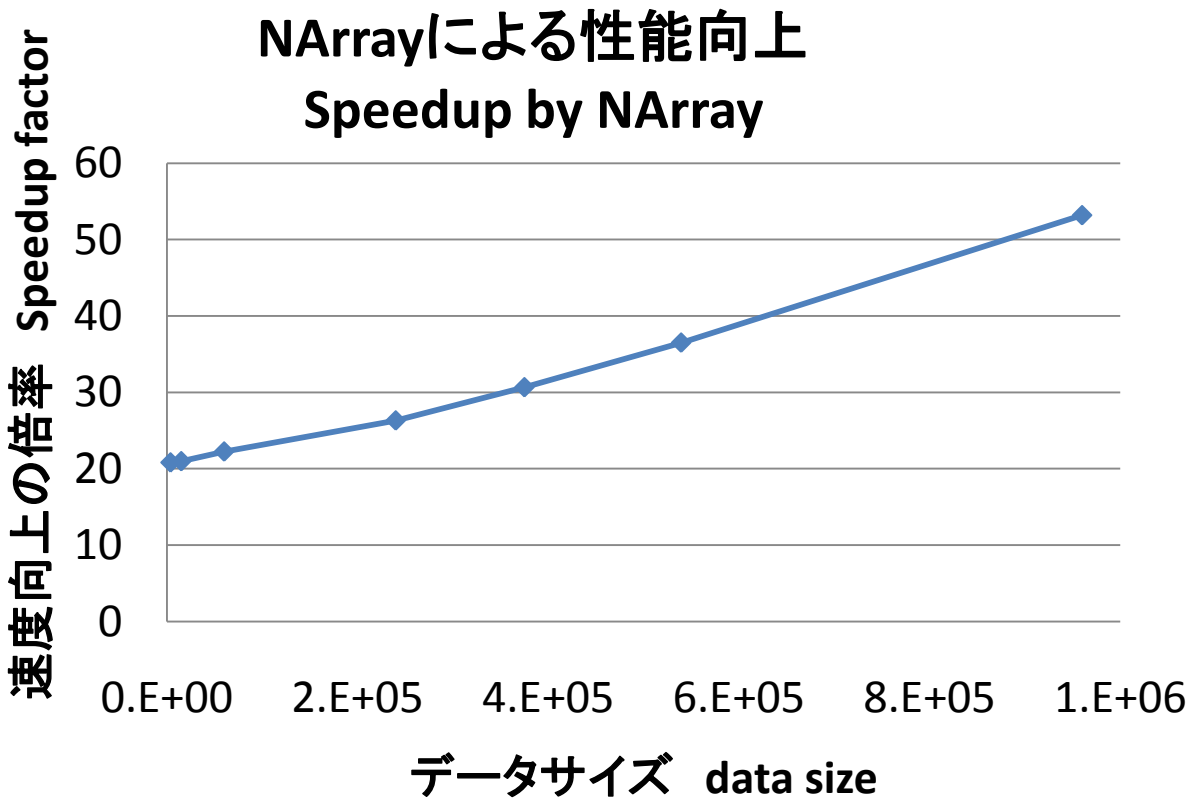
  - elapsed time : 180 ms
- NArray with Ruby 1.9.2 :

```
a*b
```

  - elapsed time : 6.4 ms
- NArray is ~ 28 times faster　　28倍速い
- 8 times less characters　8倍短く書ける

マンデルブロの計算速度
Mandelbrot calculation speed



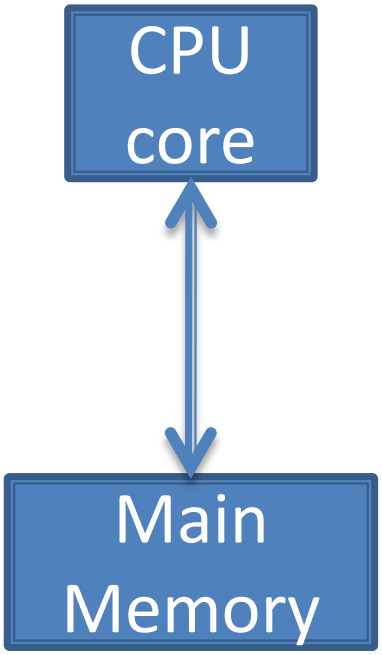


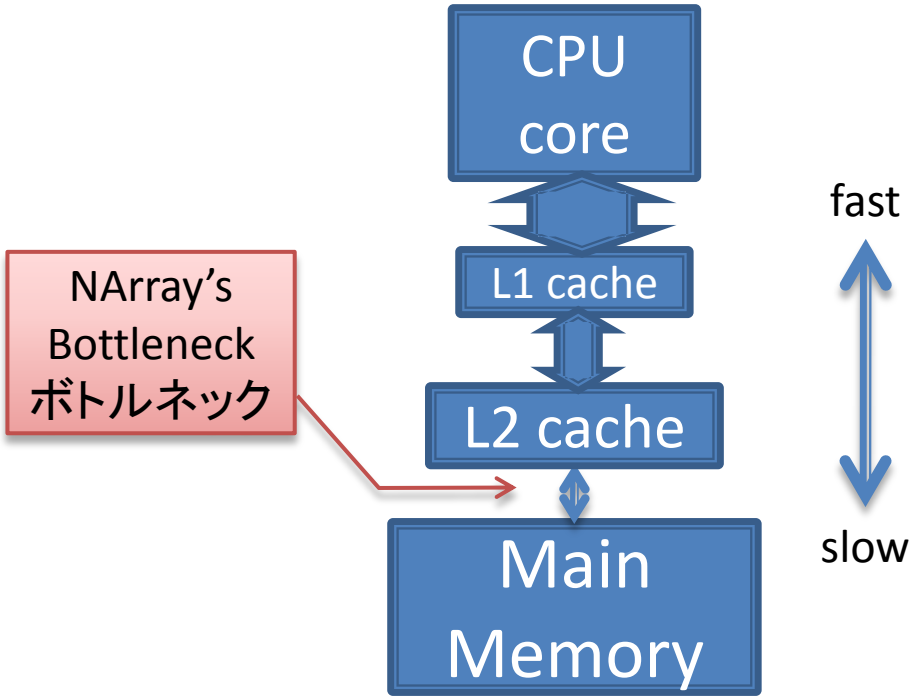
コードはNArrayのサイトにあります

NArrayは速くない
NArray is not fast

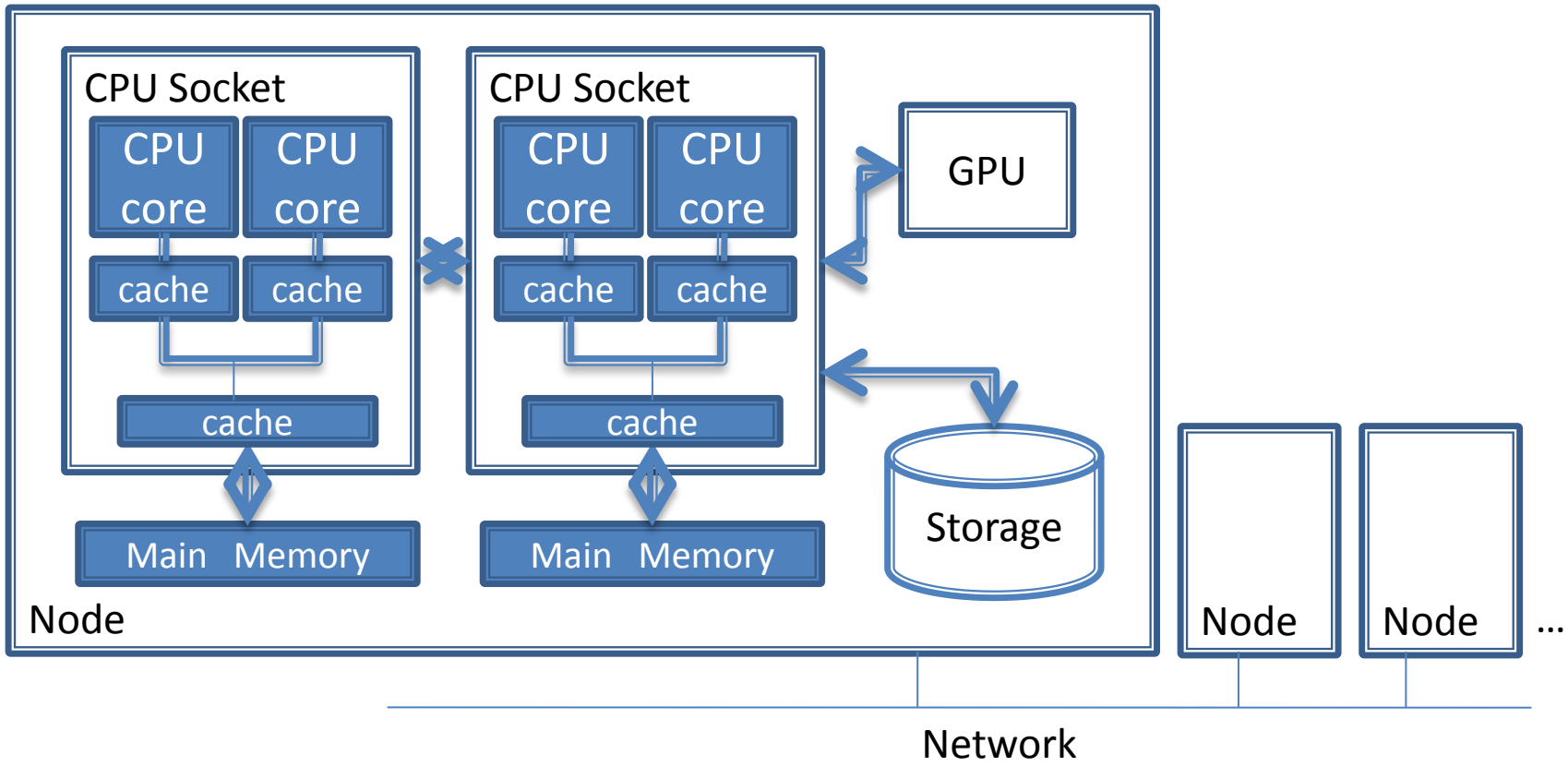
- NArrayのメモリアクセス
- Memory access for NArray

- 実際のPCアーキテクチャ
- Actual PC architechture





さらに複雑になる計算機
More complicated architecture



適材適所
Right tool for the right place

- Ruby and NArray
  - コードを書きやすい　Easy to write code
  - 複雑さの隠蔽　Hide complexity
    - アーキテクチャの違い　different architecture
    - データの分散　data distribution

科学計算に必要だと思う機能
Requirement from scientific computing

- プログラムが書ける – Ruby
  - Programmable
- インタラクティブな実行 – irb
  - Interactive execution
- データ配列操作 – NArray
  - Data Array Manipulation
- 数値計算ライブラリ – **Ruby/GSL** , etc.
  - Library for Numerical Algorithms
- グラフ表示機能 – **?**
  - Plotting Graphs

話さなかったこと
I have not talked about

- NArrayの各機能
  - NArray functions
- NArrayの使い方
  - NArray usage
- 次版NArray
  - Next version of NArray
- 他の類似数値配列との比較
  - Comparison with other numerical arrays, e.g. Numpy
- Python が科学分野で主流になりつつある件
  - Python is becoming major in Science field.

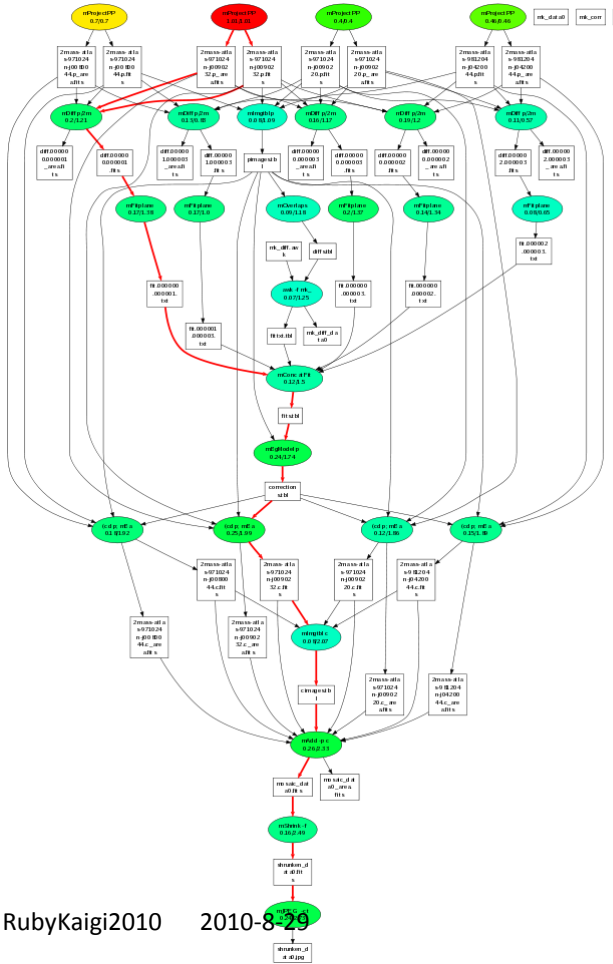
今の研究の紹介
My current research

田中昌宏
Masahiro Tanaka
筑波大学
University of Tsukuba

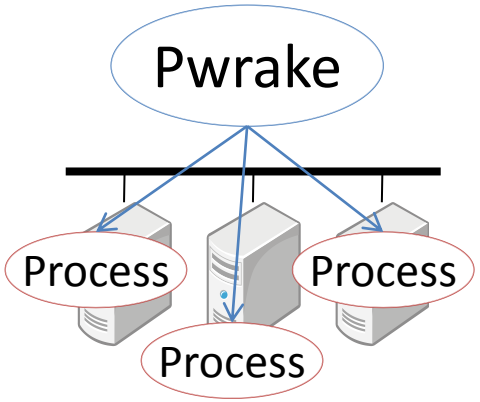
分散システムの利用
Distributed Systems

- 科学データ処理
  - Scientific data processing
- 複雑なワークフローである
  - is a complicated workflow.
- ワークフロー記述言語が必要
  - It needs definition language
- Makefileによる記述が提案
  - Use of Makefile has been proposed
  - e.g. GXP make

ワークフローのグラフ
Workflow graph



Pwrake – 並列分散Rake
Parallel distributed Rake



- **Rake**によるワークフロー記述を提案
- We propose defining workflows **Rake**
  - Makeより記述力が高い
  - More descriptive power than Make
- Rakeの並列分散拡張として、**Pwrake**（プレイク）を開発
- We are developing **Pwrake**, a Parallel workflow extension for Rake
  - http://github.com/masa16/pwrake

まとめ Summary

- NArrayを使った科学計算の例について紹介
  - Examples of Scientific computing using NArray
- NArrayとその特徴を紹介
  - NArray and its features
- 並列分散ワークフロー実行システムPwrake
  - Parallel Distributed Workflow System Pwrake

ご静聴ありがとうございました
Thank you !

- ご質問
- Questions?