

東京大学大学院 情報理工学系研究科  
平成23年度 夏学期  
最終回

# ソフトウェアの発展的構成法について

東京大学大学院  
情報理工学系研究科 創造情報学専攻

稻葉雅幸

Department of Creative Informatics

<http://www.jsk.t.u-tokyo.ac.jp/>

# 行動実現研究 東大 JSK

2



**H6 & H7 Humanoid**  
1999 S. Kagami,  
K.Nishiwaki



**Musculoskeletal 2000-**  
**I. Mizuuchi Y. Nakanishi**



**Assistive Humanoid**  
2002- K. Okada



OSS, RTM, ROS  
OpenRave  
2010R. Diankov  
All members



**COSMOS:**  
CognitiveSensor Motor  
Operation Studies  
1981 T. Ogasawara  
T. Matsui, H. Mizoguchi  
M.Inaba, H. Inoue



**Remote-Brained Robotics**  
1993- M. Inaba



**Interactive Systems**  
2004 T. Inamura

**IRT Home Assitance**  
2006-  
K. Yamazaki  
R.Hanai

IRT → PR2

HRP2 Integration

Musculoskeletal Humanoid

Sensor Suit

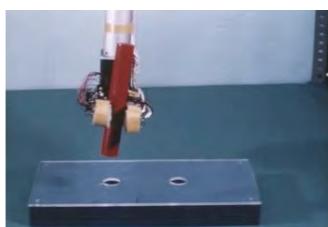
Sensor Flesh

Remote-Brained Robotics: about 60 robots

HARP: Humanoid Autonomous Robot Project H1-H7

Vision-Based Robotics: Manipulation, Interaction, Navigation

COSMOS: Lisp-based Robot System Integration Environment



**Bilateral**  
1969 H. Inoue

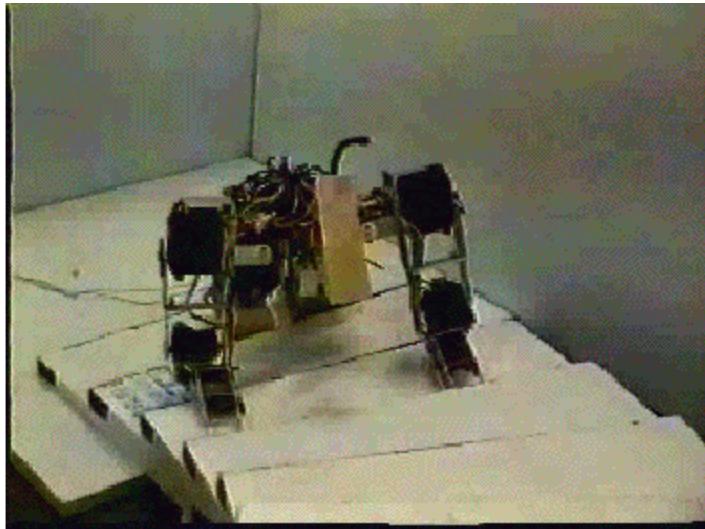
1980

1990

2000

2010  
2

# 金広文男 学位論文 1998 「人間型全身行動ロボットシステムの発展的構成法」



小型で移動する 1993



寝返り立ち上がり 1995



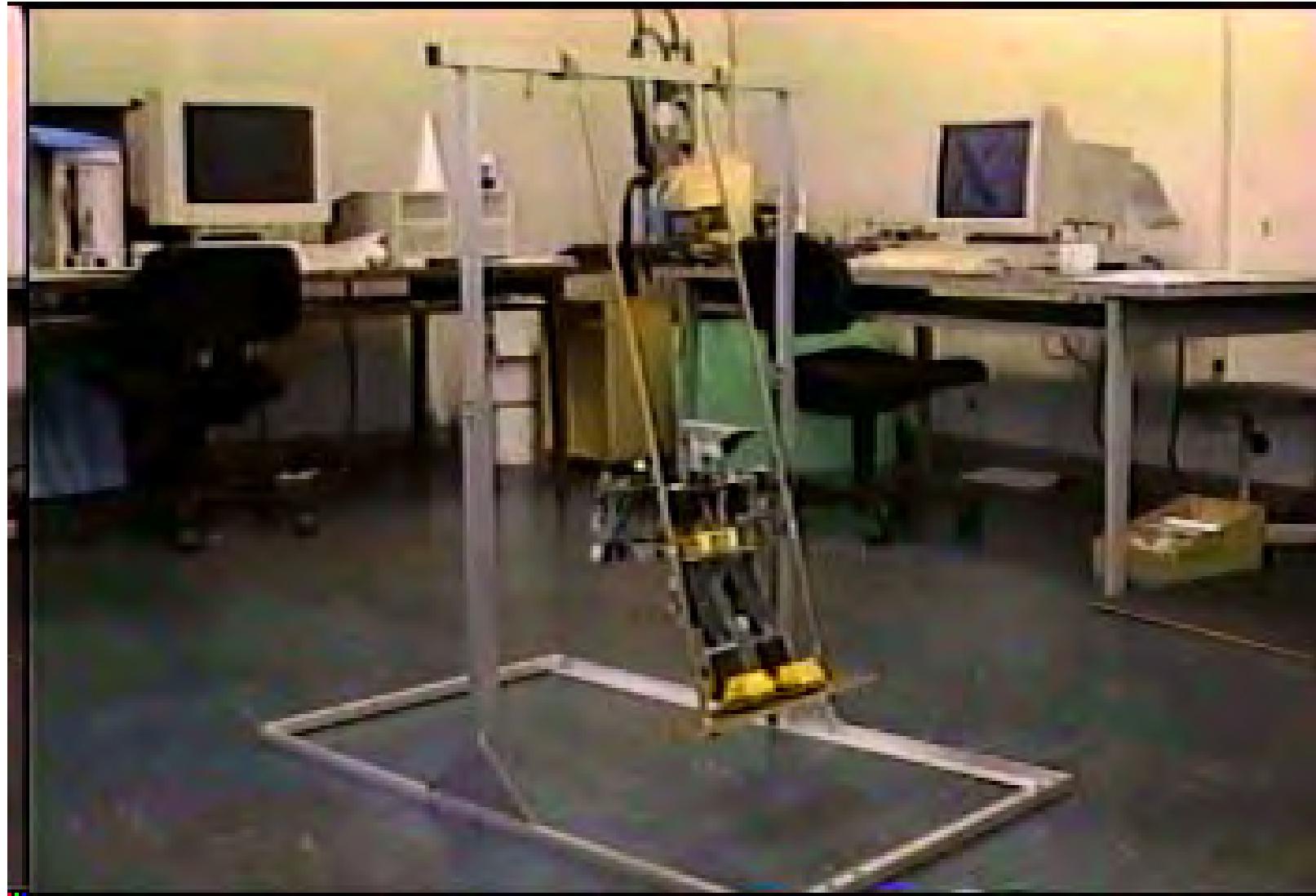
立ち上がり  
HRP2-P 2000  
  
金広 1993-1999,  
ETL MITI 2000-



## Remote-Brained Robot Environment 93

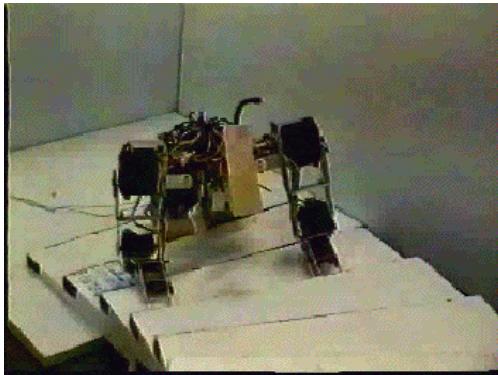
旧2号館3階301号実験室 今日の講義室332号室の真上あたり

# 人の姿形をしたロボットはどうやって作るべきか 1990—

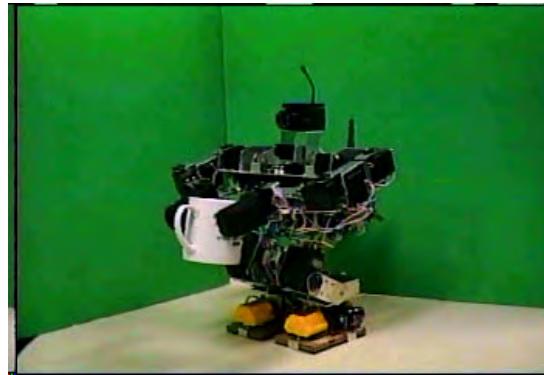


大きな計算機を使え、全身をもつこと

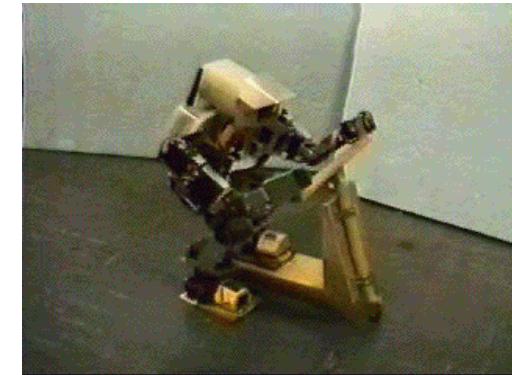
# ヒューマノイド全身行動ロボットでの展開



視覚認識 ナックル歩行



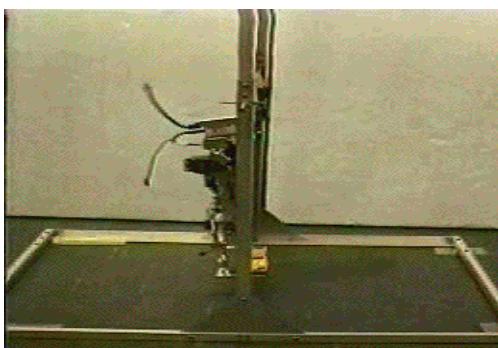
可変負荷対応平衡 物体操作



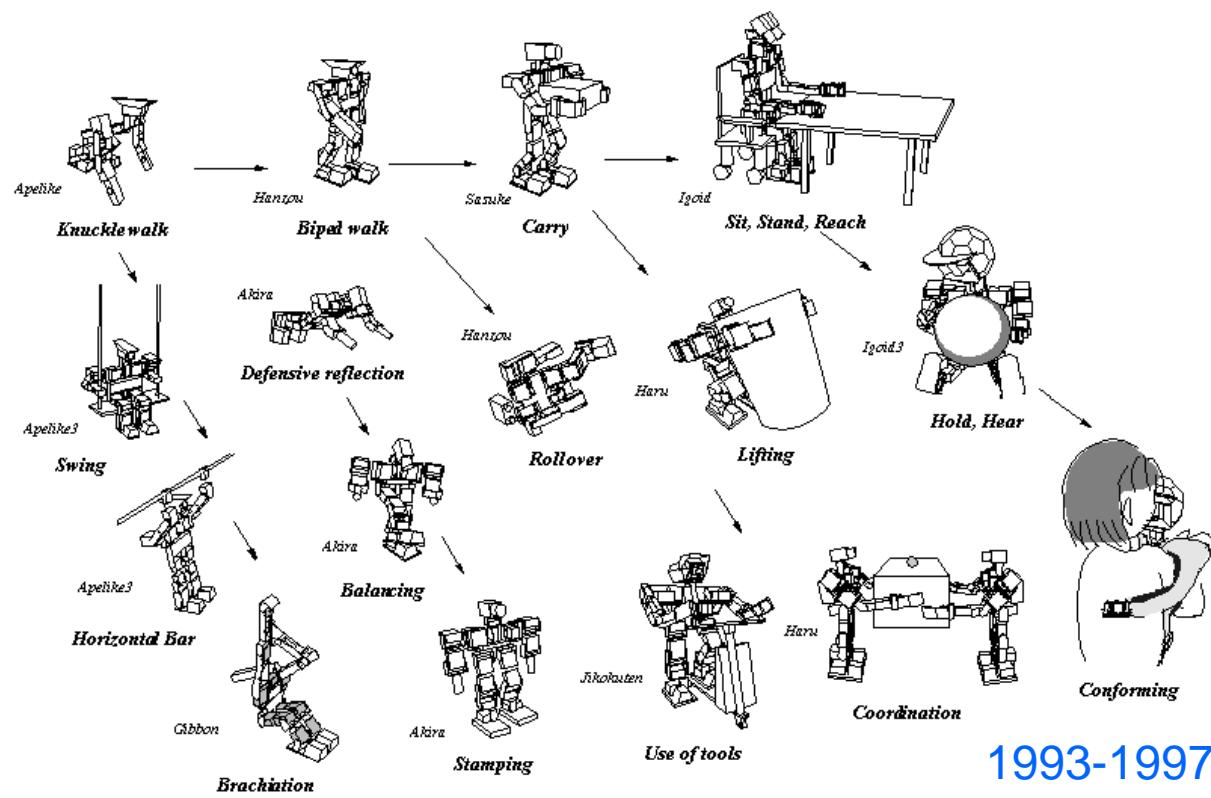
全身行動 道具使い



倒れても起き上がる

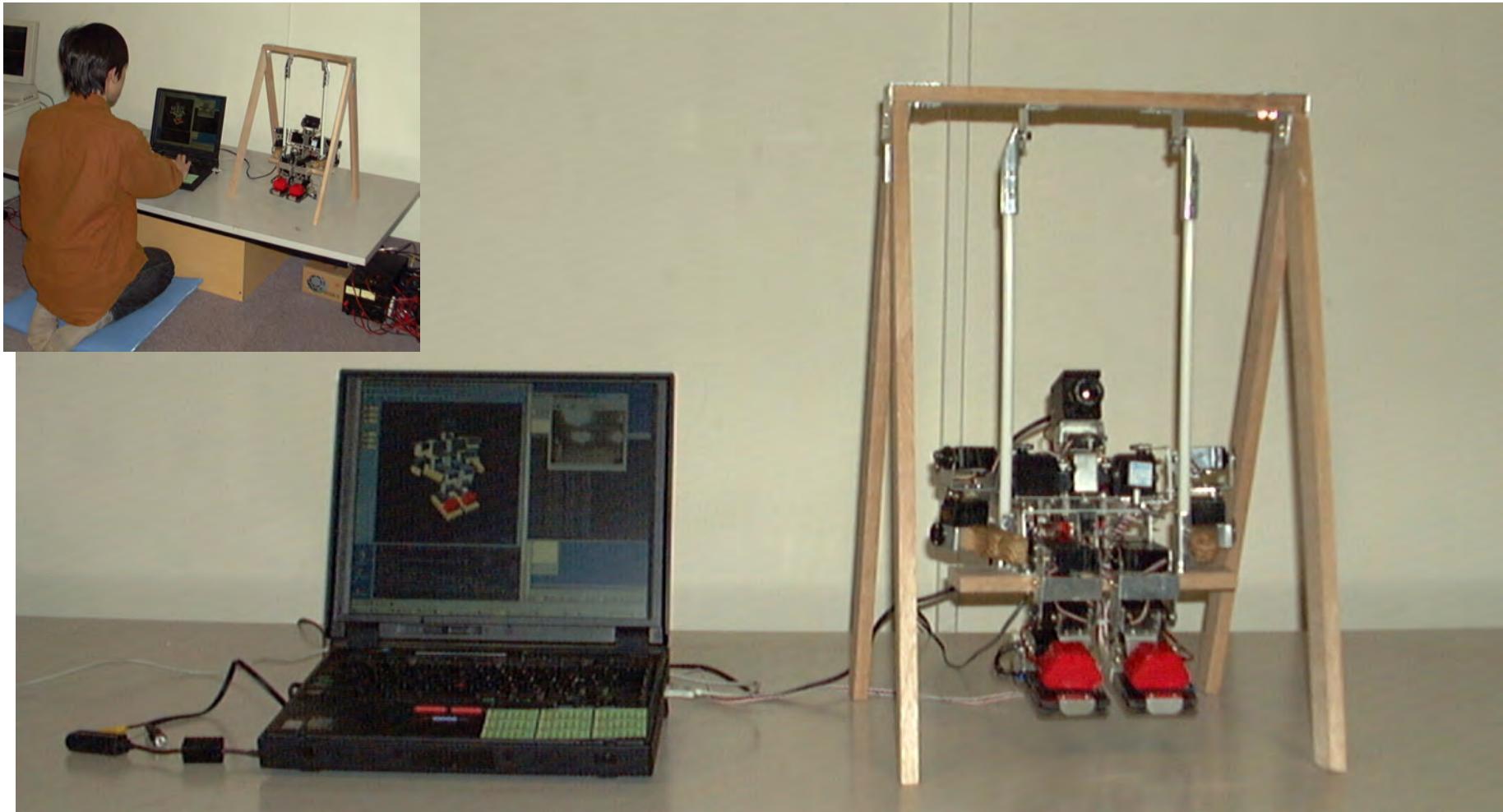


全身揺動 行動学習



1993-1997

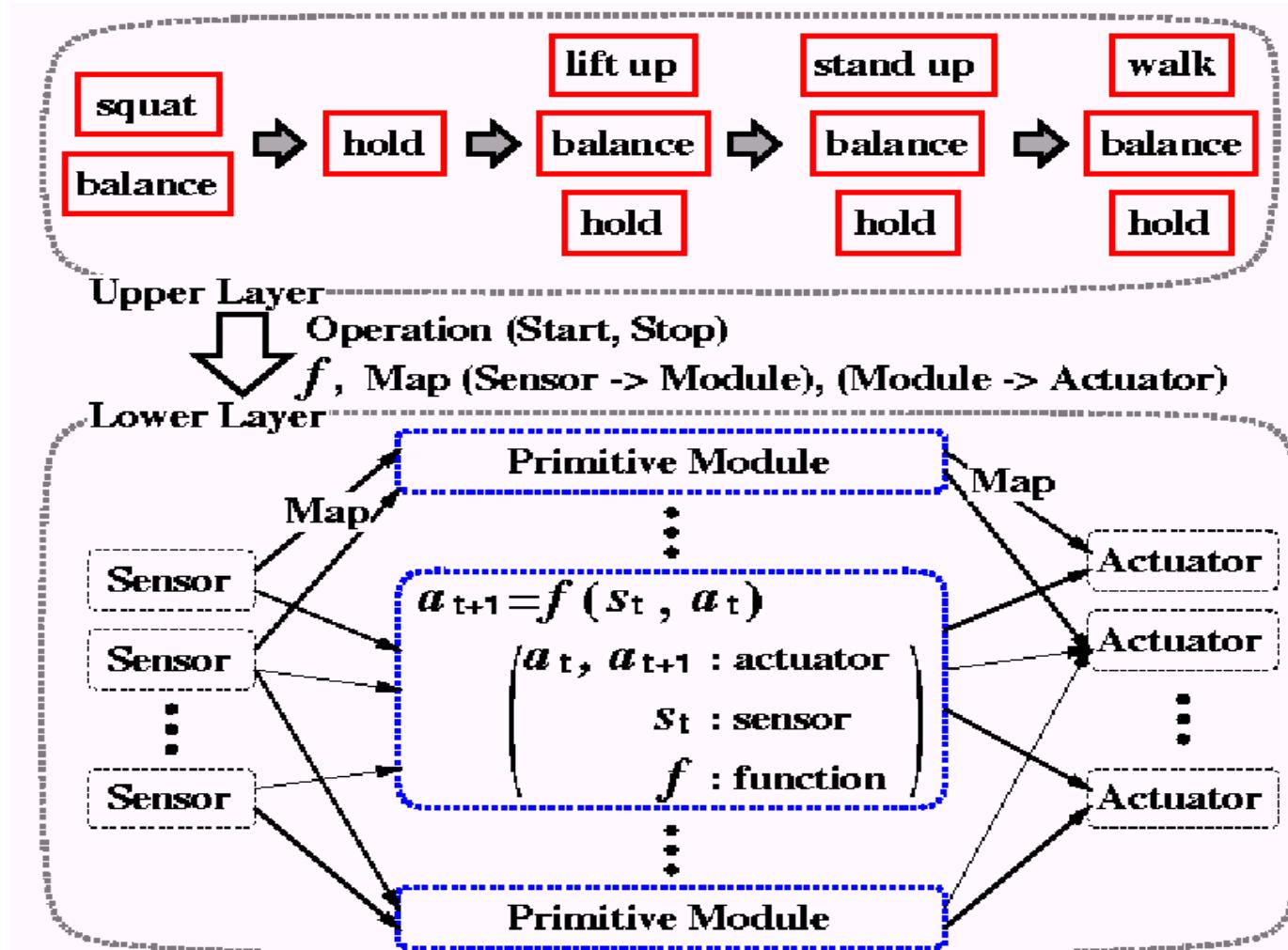
# 卓上リモートブレインヒューマノイド



自分で作る卓上ヒューマノイドロボット 1999 無線LANがまだ無かったころ

視覚カメラ、多自由度身体、幾何モデル、認識・行動ループ、開発マザー、繰返し行動環境

# Two Layered Brain System



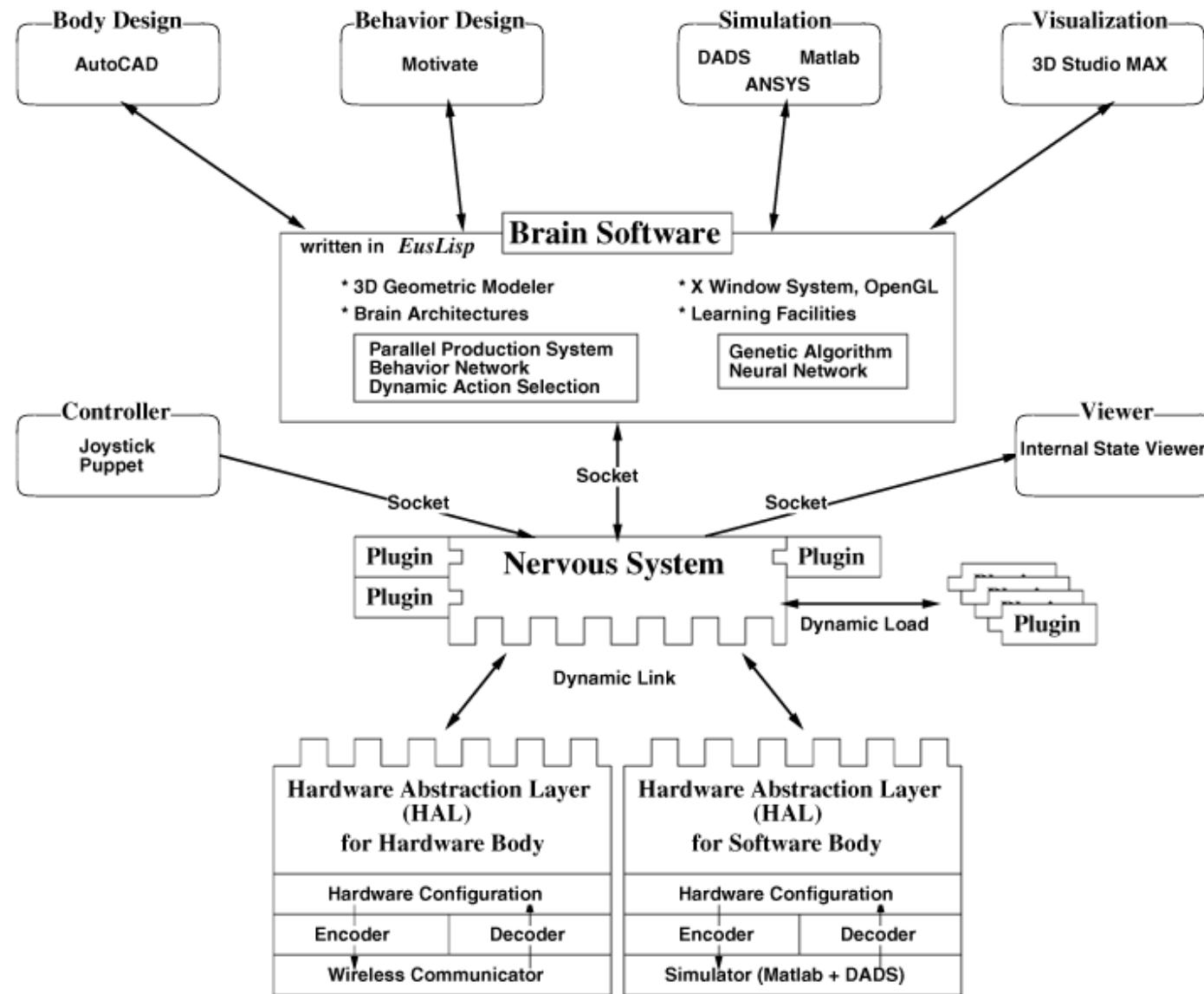
Goal Sequence  
Decision  
Layer

- State Analyzer
- Recovery

Real-time  
Autonomous  
Layer

- Goal-Driven
- Sensor-Driven

# Mother and Brain in Remote-Brained Robots 1998



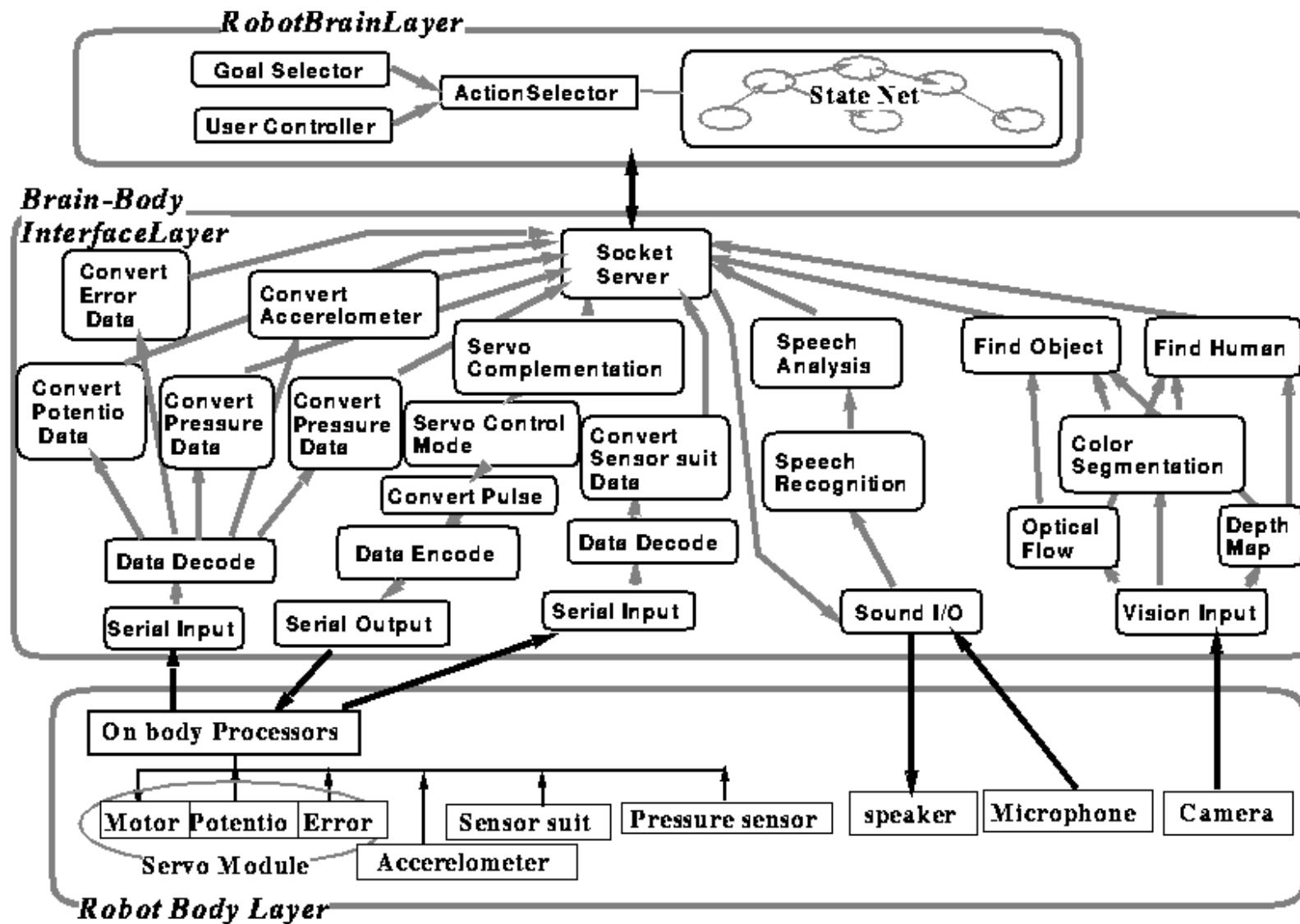
**Mother &  
Brain  
Development  
Environment**

**Sensor-Motor  
Integration  
Layer of Brain**

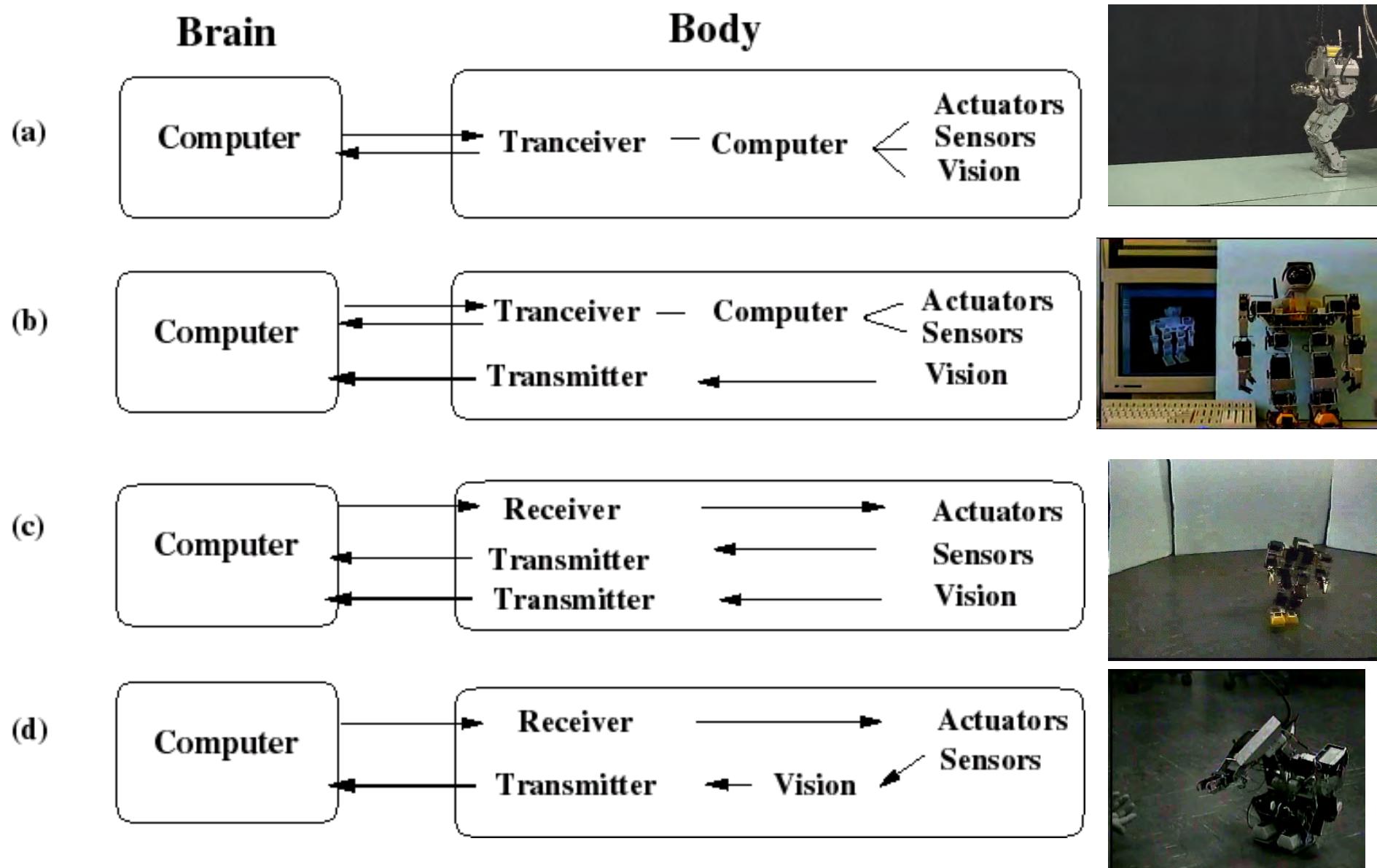
**Real-Time  
Autonomous  
Layer of Brain**

**Hardware  
Abstraction  
Layer of Body**

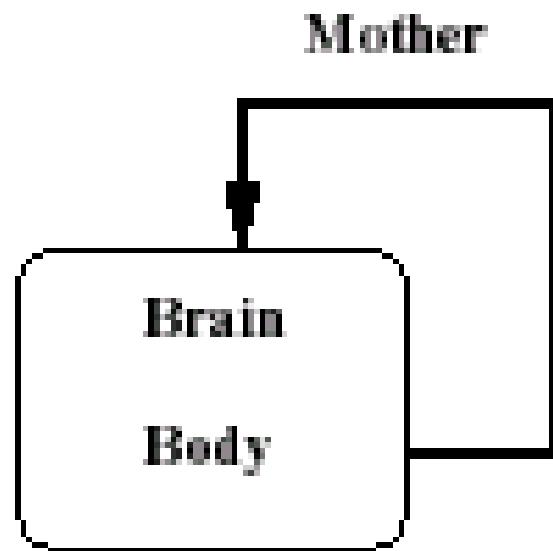
# CORBA Extension 1999



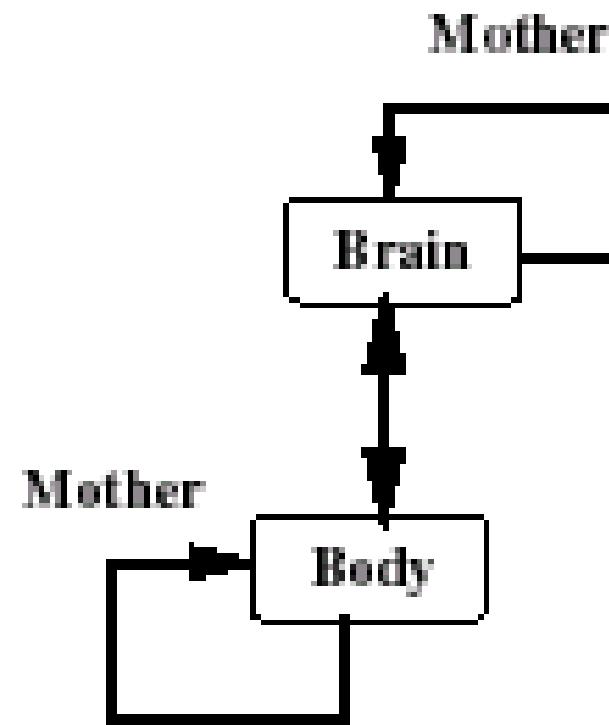
# ボディの作り方の変遷



# ロボットの開発 ボディとブレイン、マザー環境



(A)



(B)

開発者の存在  
を考える

開発者の操作  
 자체を研究

発展的構成法  
の研究

# リモートブレインロボット

## ブレイン・ボディ・インターフェース・マザー環境

*Mother Environment*

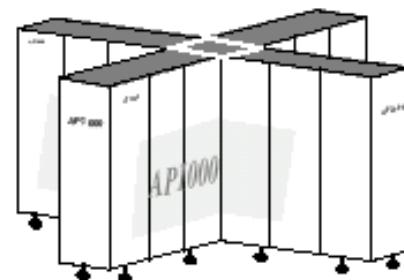
*Robot Brain*

Parallel Computer  
Sensor Receiver  
Motor Transmitter

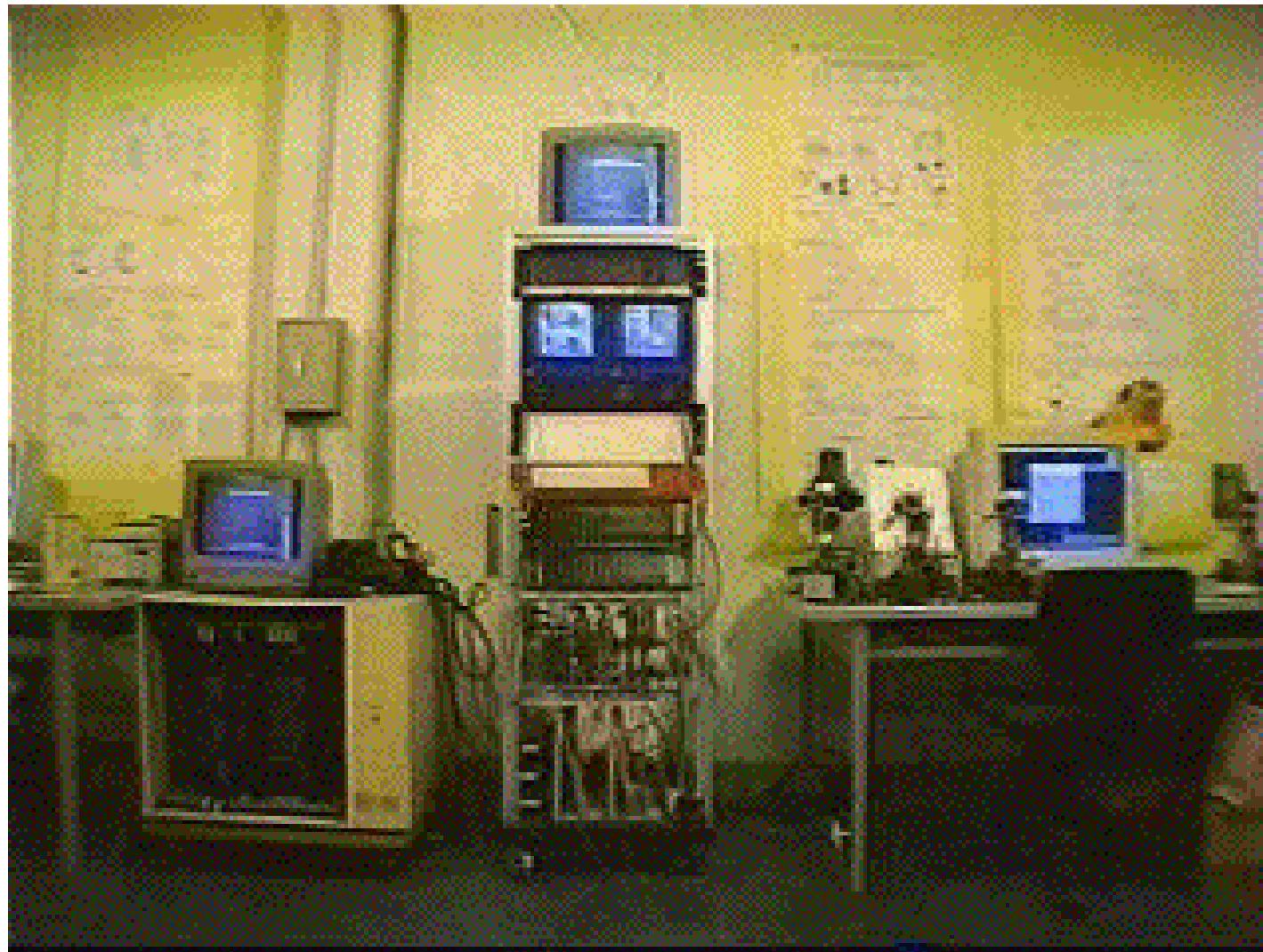
*Real Environment*

*Robot Body*

Sensor  
Effector  
Power Source



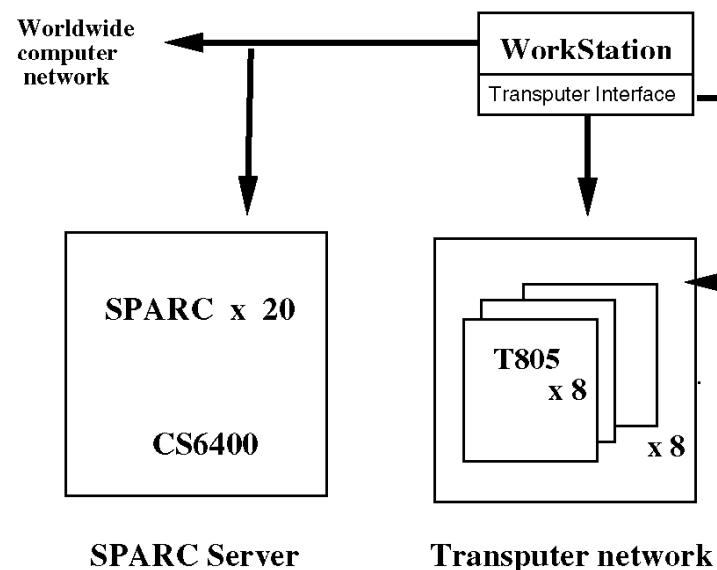
# リモートブレインロボット (1992)



脳を持ち歩かないロボット  
頭脳と身体を分けて無線で結ぶ

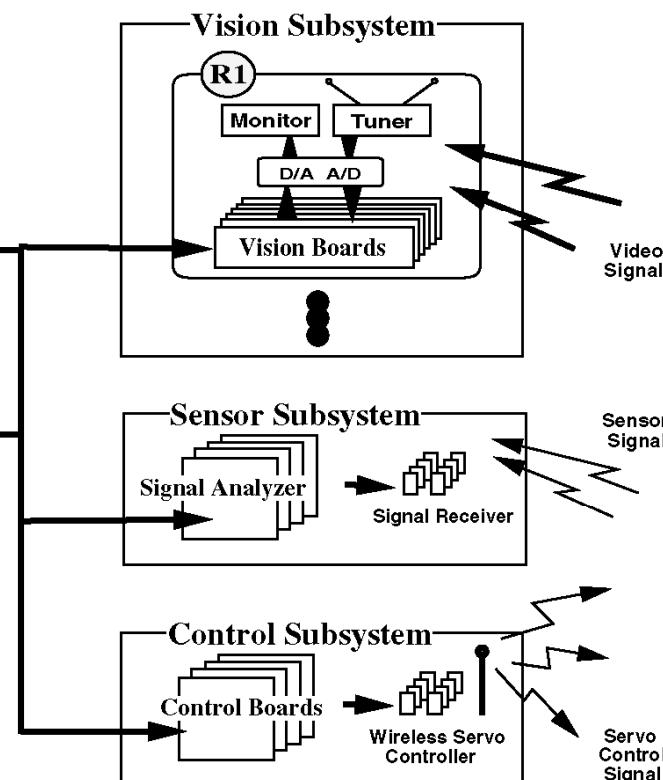
## Brain & Mother Environment

(Inoue & Inaba Lab.  
Dept. of Mechano-Informatics  
University of Tokyo)



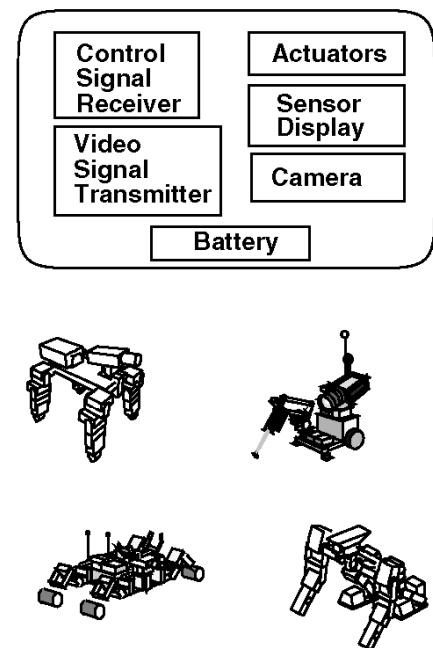
ワークステーション  
euslisp  
トランスペュータ分散並列  
処理コンピュータ  
Parallel C/Occam

## Brain-Body Interface



局所相関演算  
トラッキングビジョン  
ラジコン波形生成

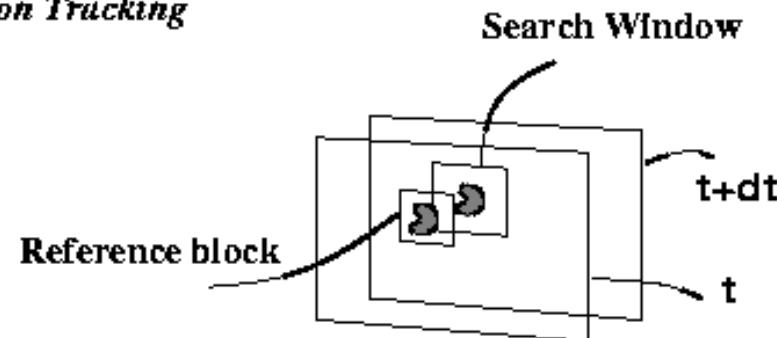
## Robot Bodies



ラジコン模型  
ロボット身体  
レシーバ  
128モータ

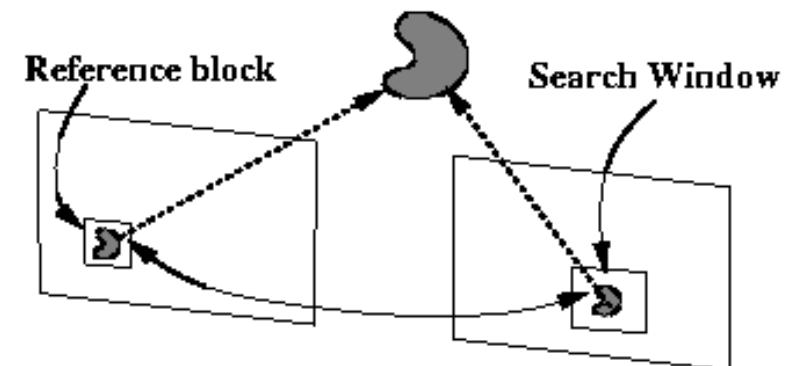
# 局所相関演算によるロボットビジョン

*Motion Tracking*



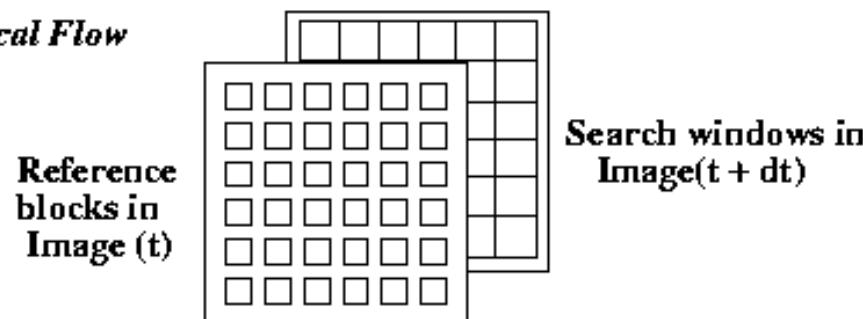
追跡

*Stereo Matching*



ステレオ

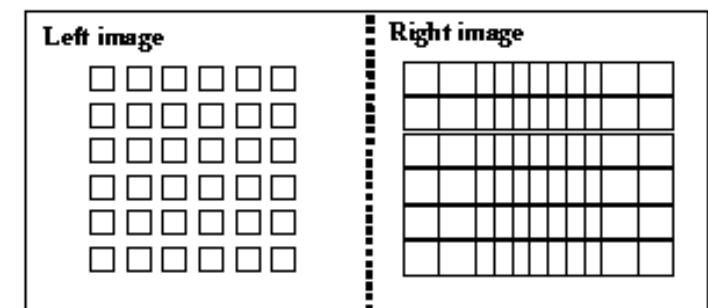
*Optical Flow*



Flow vector array

フロー

*Depth Map*



Reference blocks

Overlapped  
Search windows

Disparity array

デプス

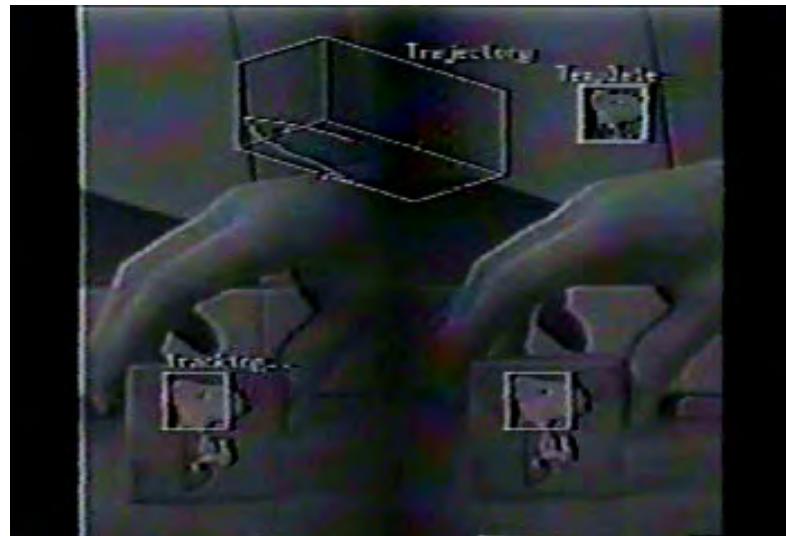
# 局所相關演算による視覚処理



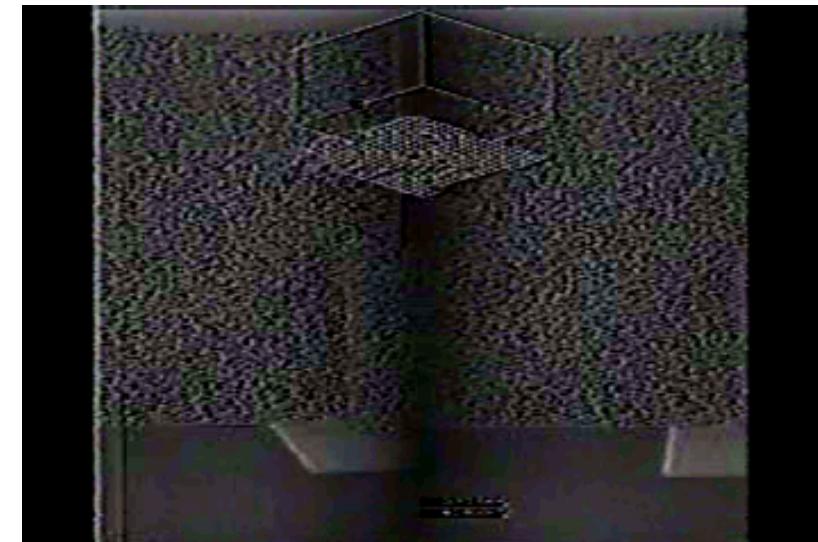
单眼 画像追跡



单眼 オプティカルフロー生成



両眼立体視 三次元追跡



両眼立体視 デプスマップ生成

# RTコンポーネント対応 組込み画像処理モジュール

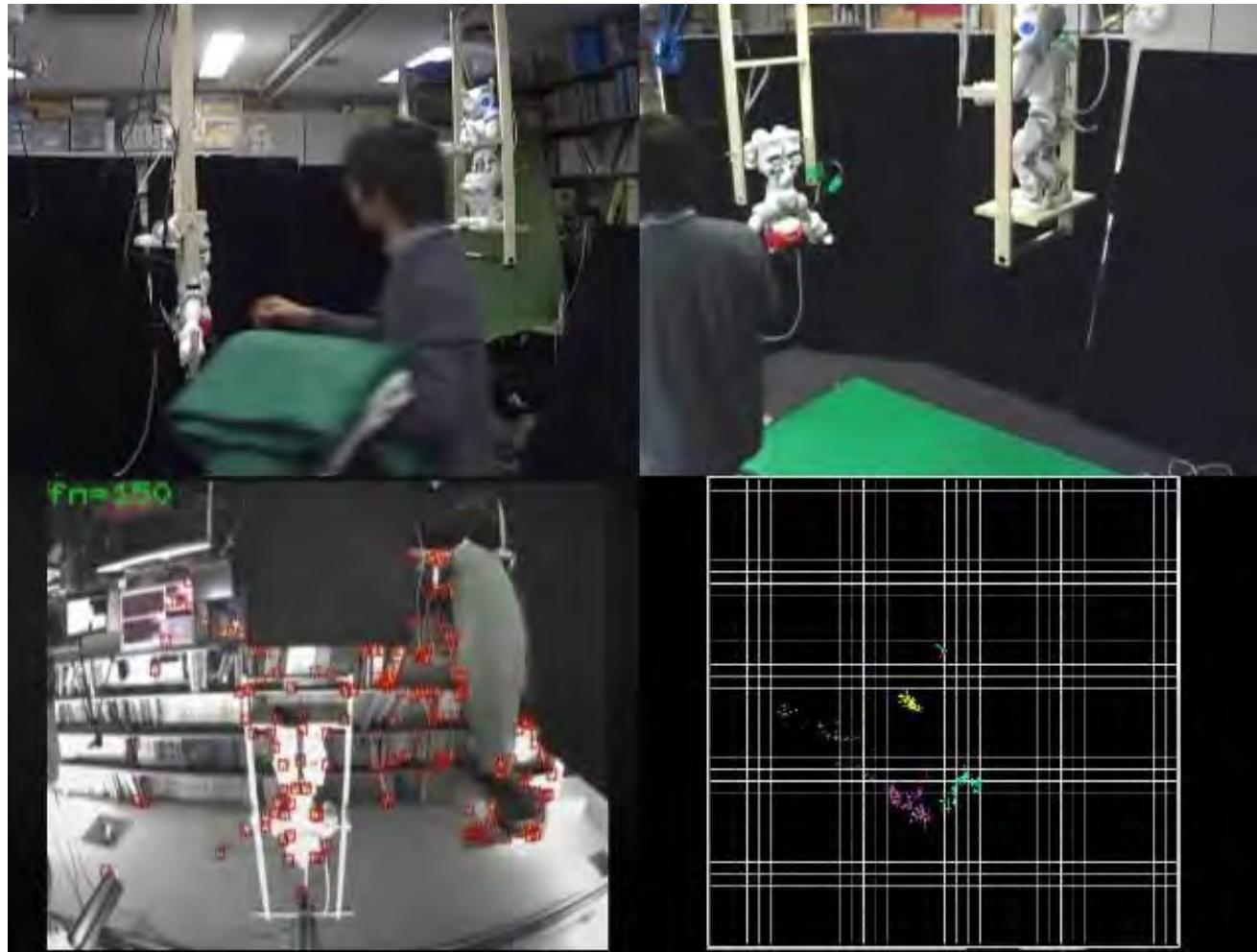
- 組込みOS搭載ハードウェア画像処理モジュール



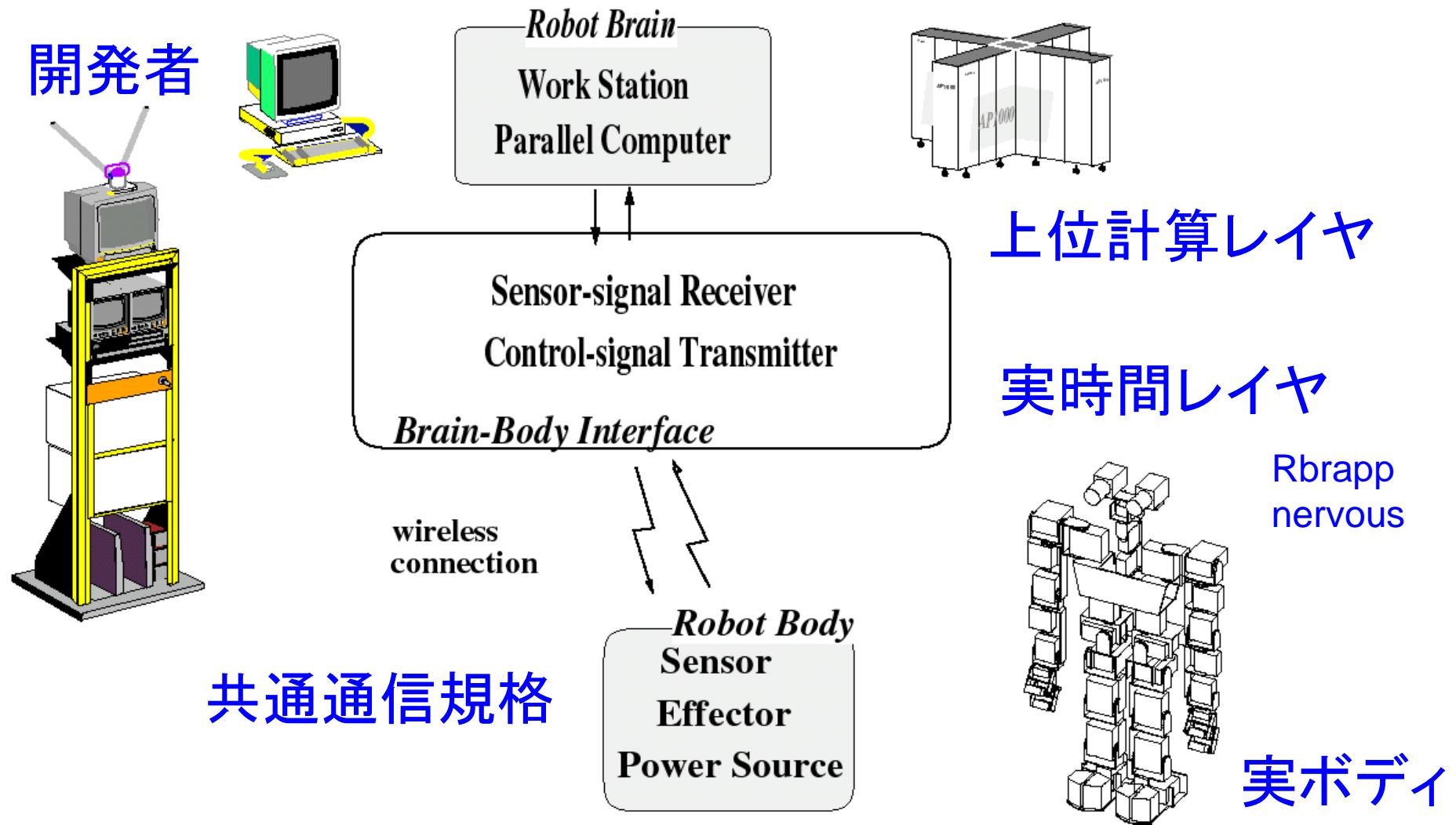
次世代知能化プロジェクトの成果として富士通研究所株式会社が開発、2009年より富士通九州ネットワークテクノロジーズ株式会社から販売

- PowerPC440EPx(666MHz) + ImageProcessingLSI(200MHz)
- OS : 組込Linux
- Ether IEEE 1394カメラ or Analog NTSCカメラ入力
  - C, C++でアプリが書ける
  - イーサネット経由でRTコンポーネントが走る
- 画像処理モジュール用RTMは株式会社セックより提供予定

# 3次元フロー応用:自己運動と移動物体の分離認識による空中ブランコ



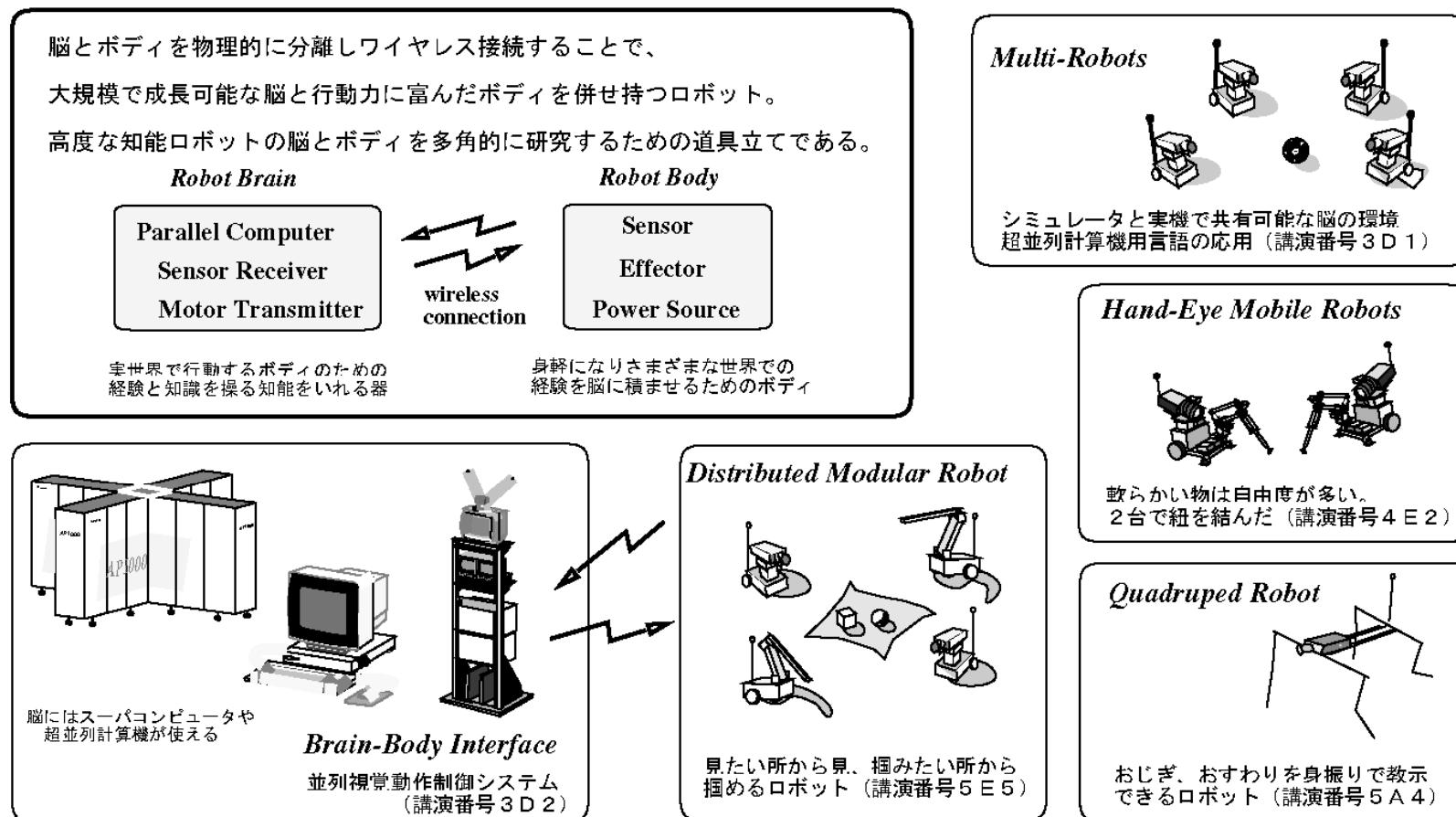
# ブレインとボディをつなぐ実時間レイヤ



多種類のボディに対応可能なプラグインの自動生成

# 車輪型、多脚型、ヒューマノイド型などの身体でも

## 脳を持ち歩かないロボット(Remote-Brained Robot) とは



## Remote-Brained Robotics の今後の展開

### 形態学的研究

*Apelike Robot,  
Humanoid Robot,  
Flying Robot, ...*

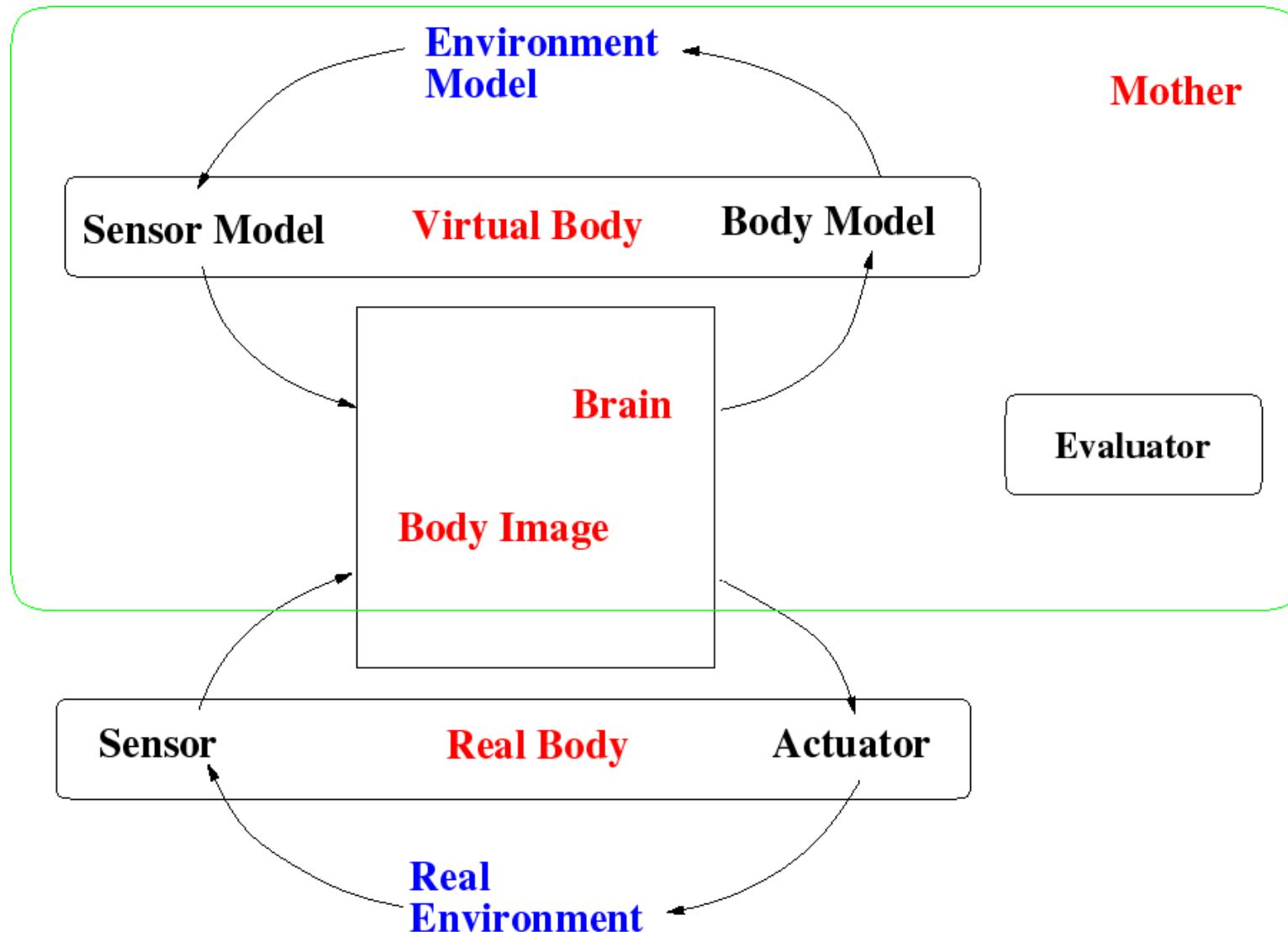
### 応用指向的研究

*Office Robot,  
Farm Robot,  
Amusement Robot, ...*

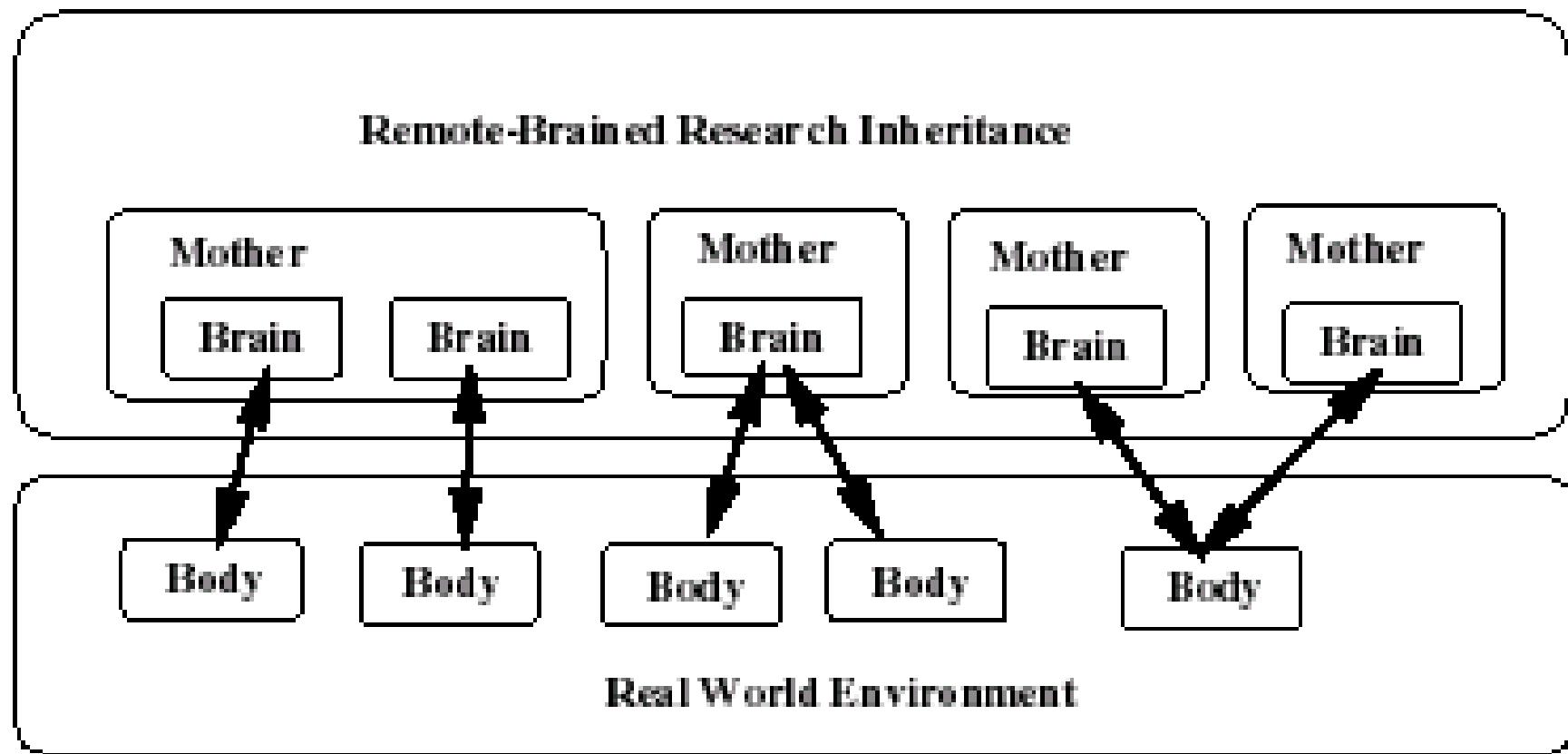
### 人工脳実現方式研究

*Memory Based Approach,  
Localist Connectionist Approach,  
PDP Approach, ...*

# 実ボディが感覚動作統合行動のループの中にある



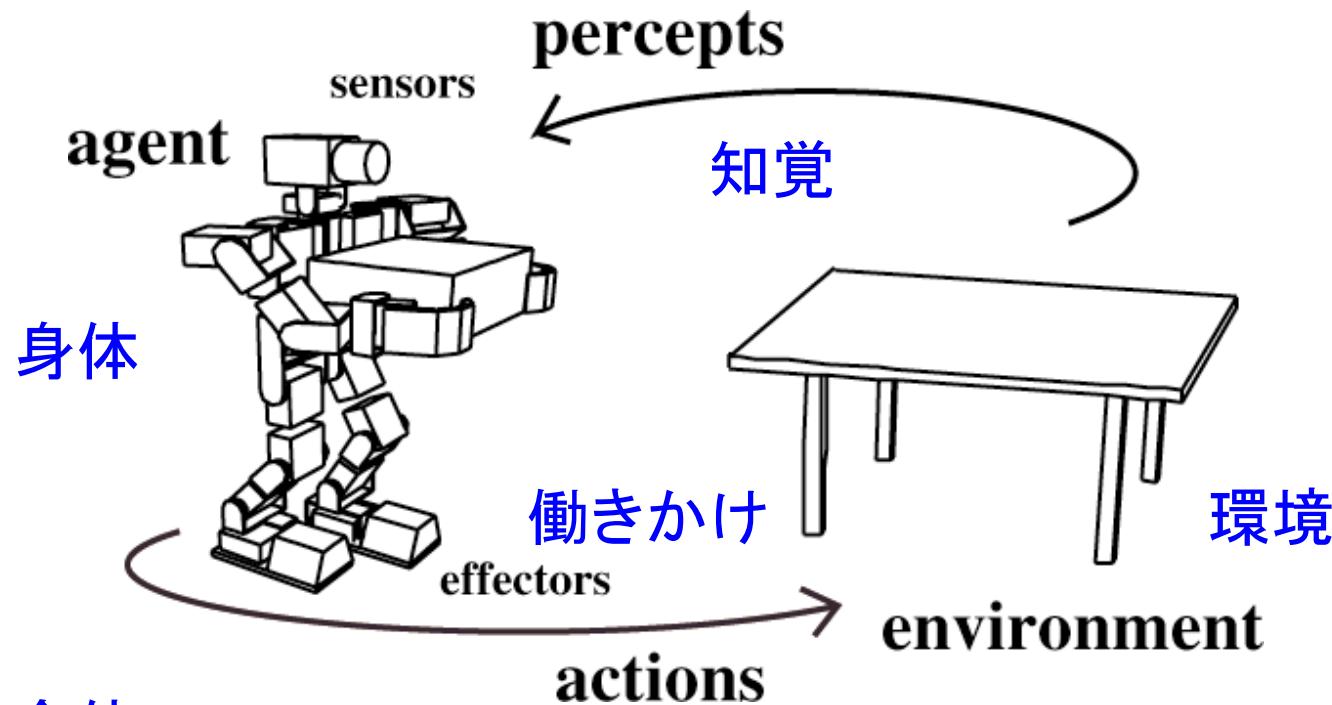
# リモートブレイン環境 ブレイン開発環境の共有



(C)

# 実世界 エージェントシステム

環境に応じて行動を実行できるロボット



## システム全体

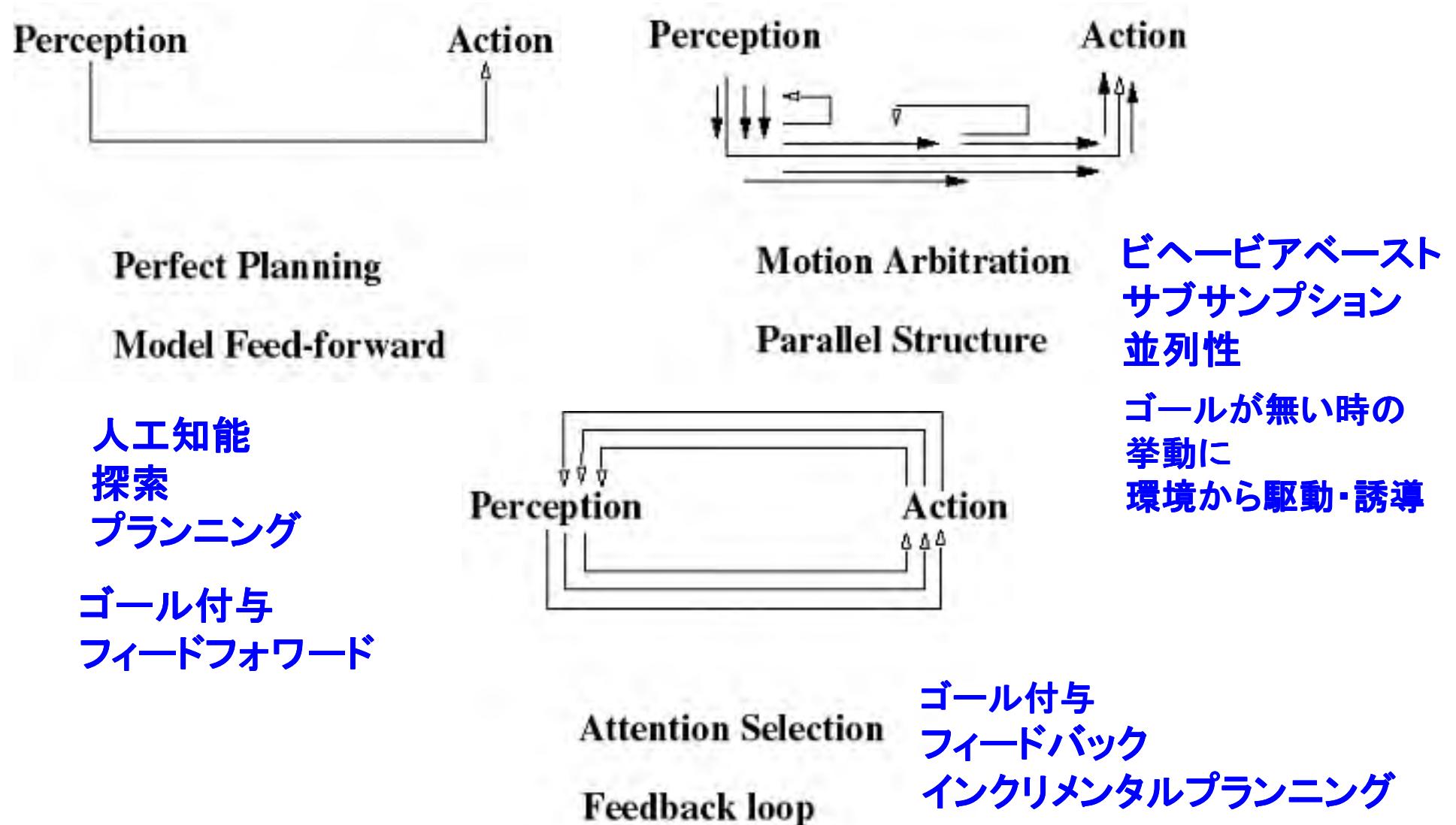
- 1) 各種身体ハードウェア
- 2) ソフトウェアの統合システム
- 3) 繙承発展可能なアーキテクチャ
- 4) 開発支援・共有継承・運用環境

## 課題

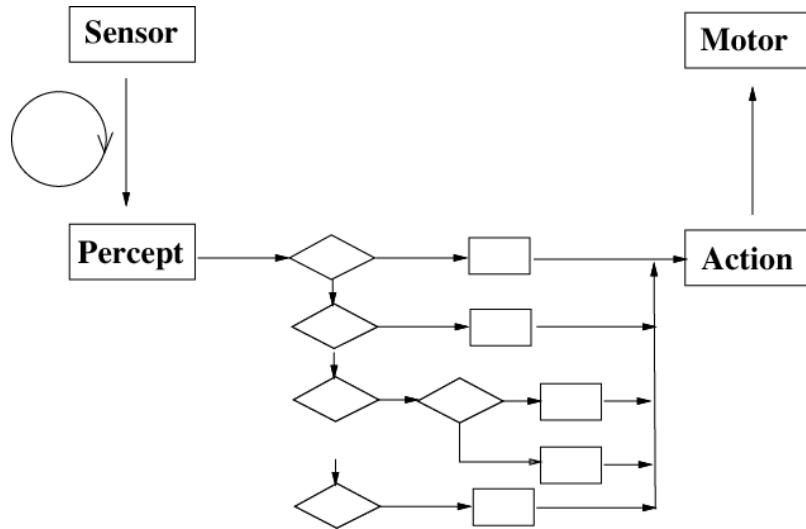
- 1) 行動目的の多様性
- 2) 異なる身体での再利用性
- 3) 異なる環境での実現性

# Robotics: Connection from Perception to Action

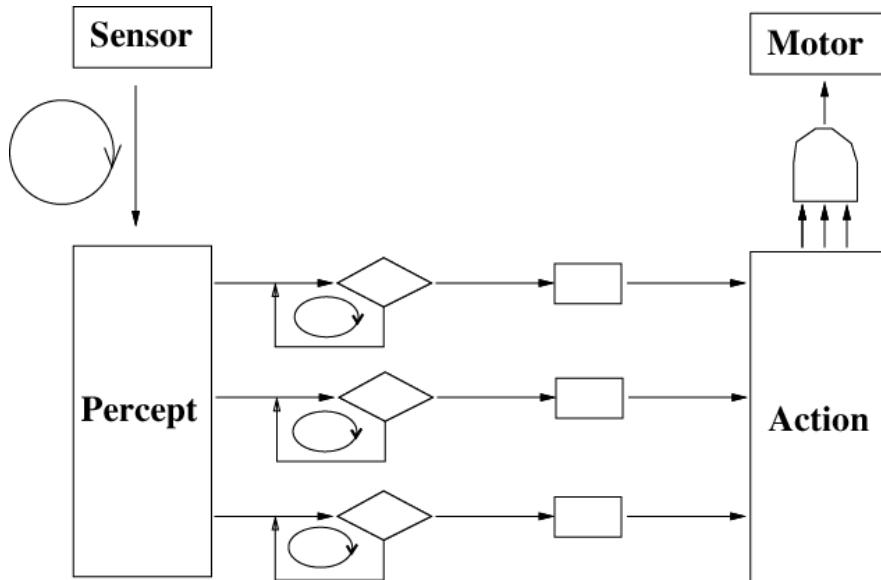
## 感覚から行動の接続形態と推移



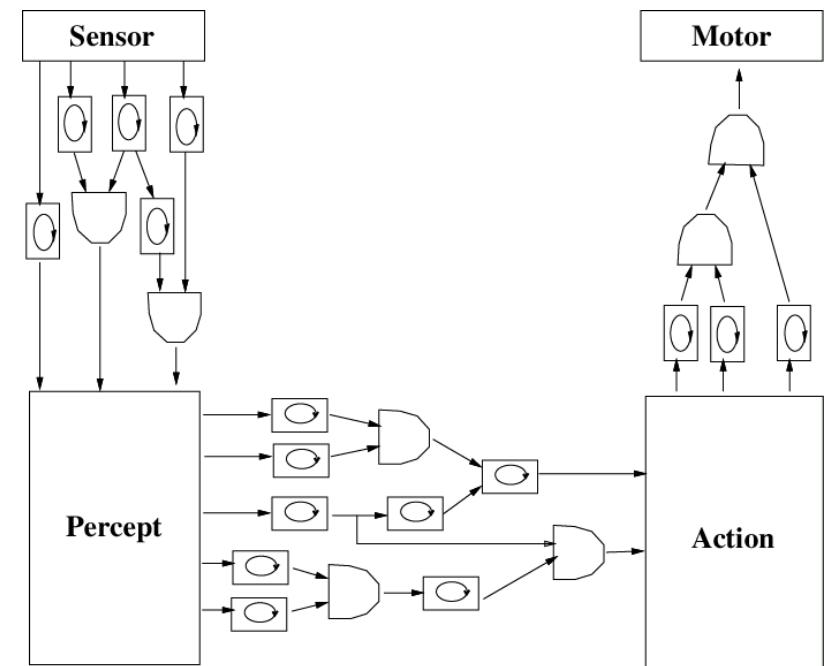
# 感覺系と動作系の ルール型接続と 構成要素構造



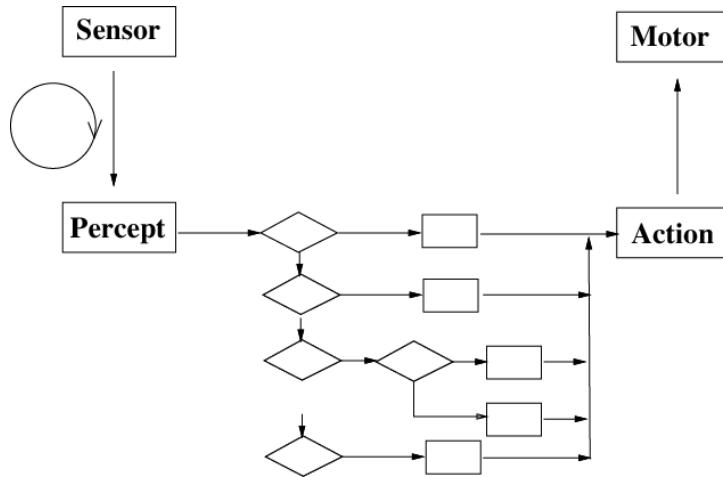
Single Process



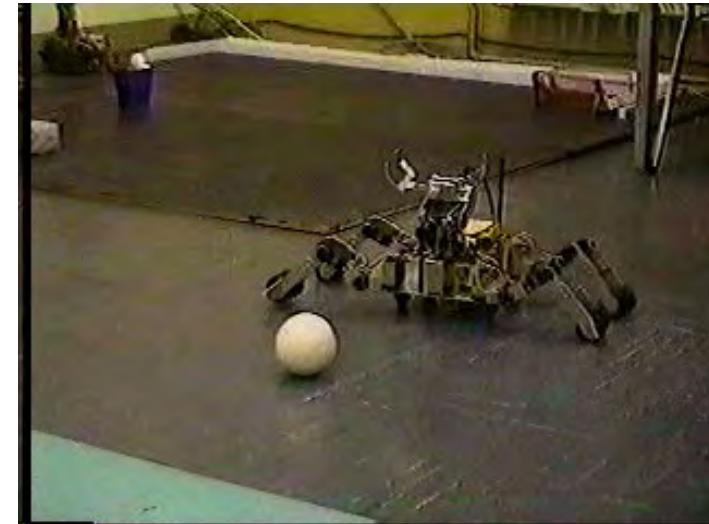
Parallel Processes



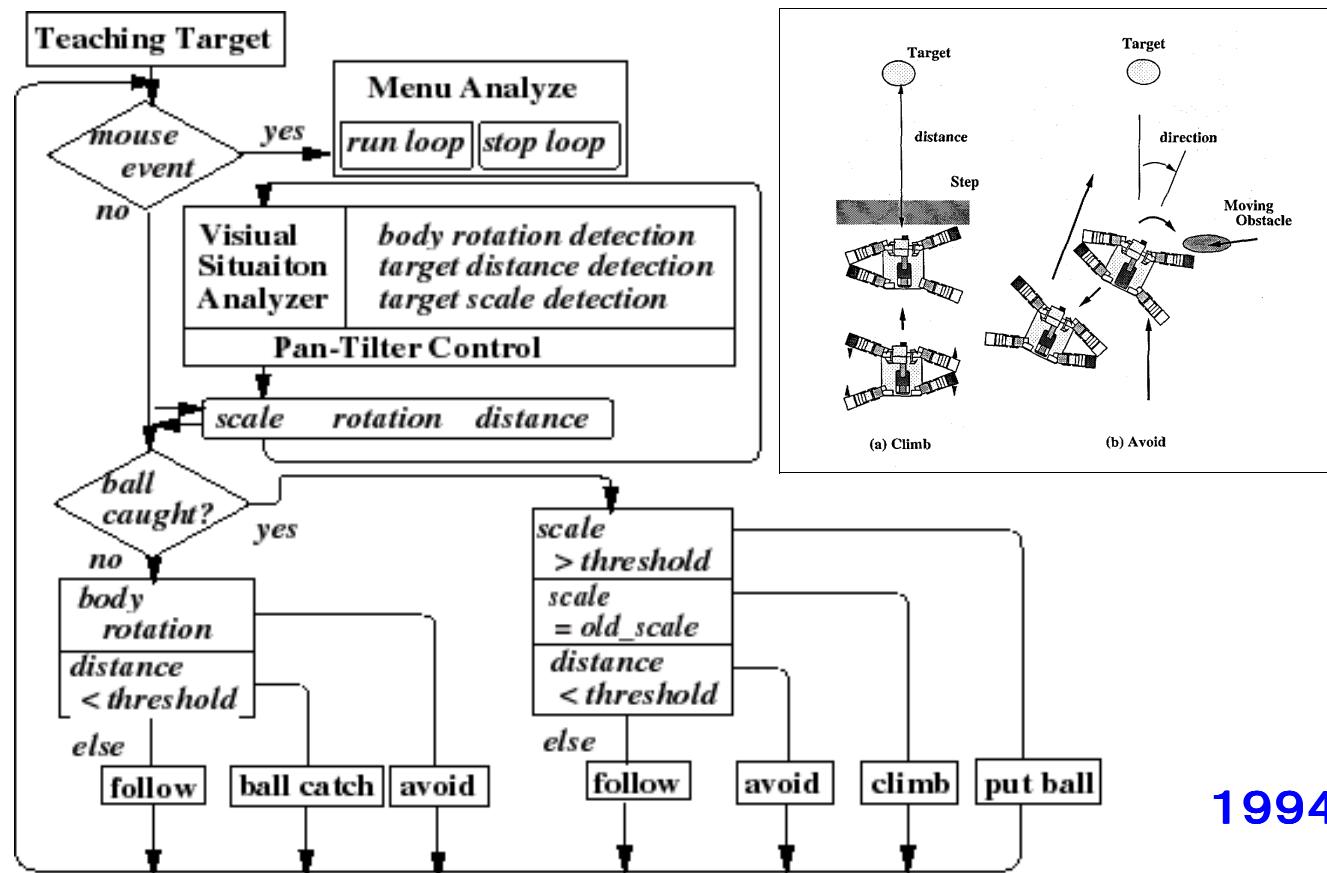
Networked Processes



单一プロセス型

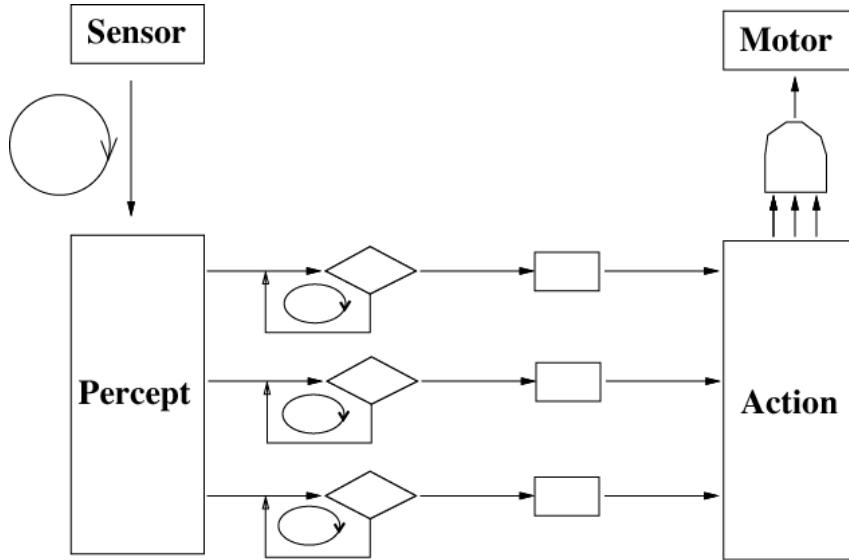


- If/then simple control
- Simple Programmed Behaviors
- Parallel Sensor Monitoring Process

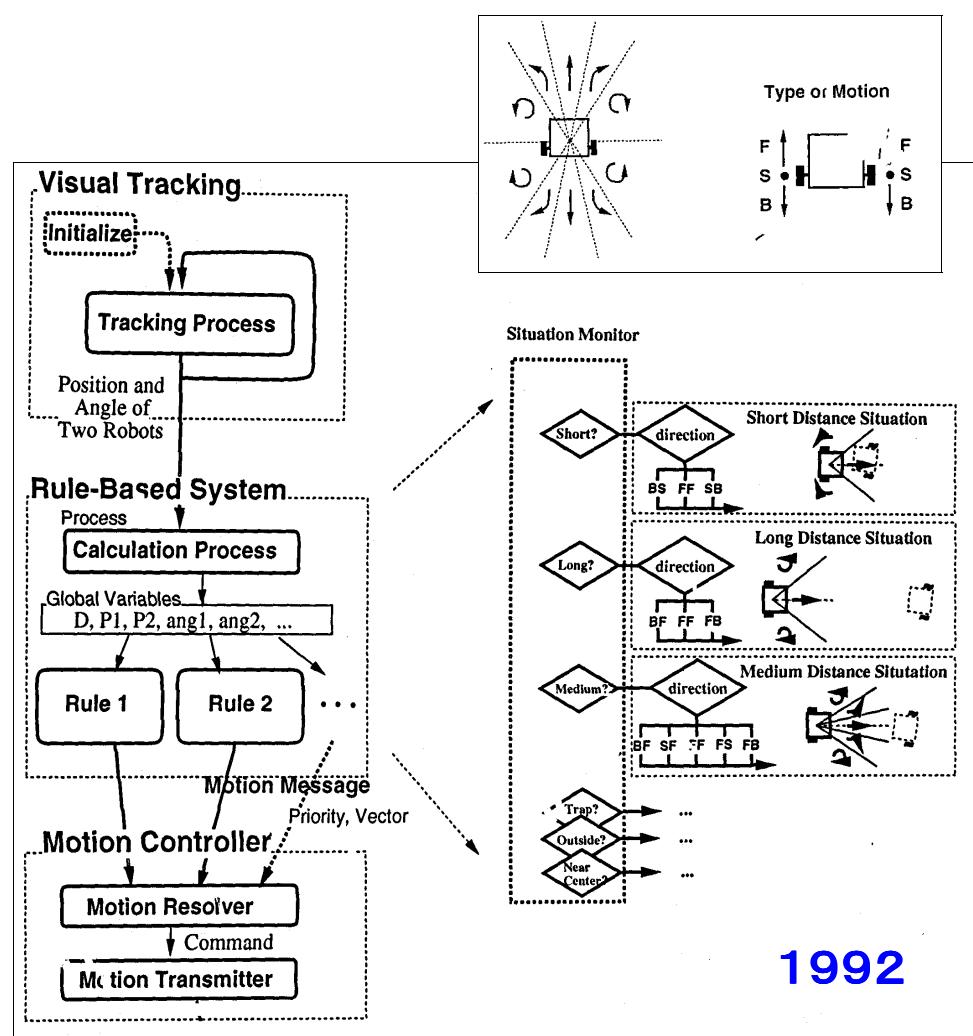


1994

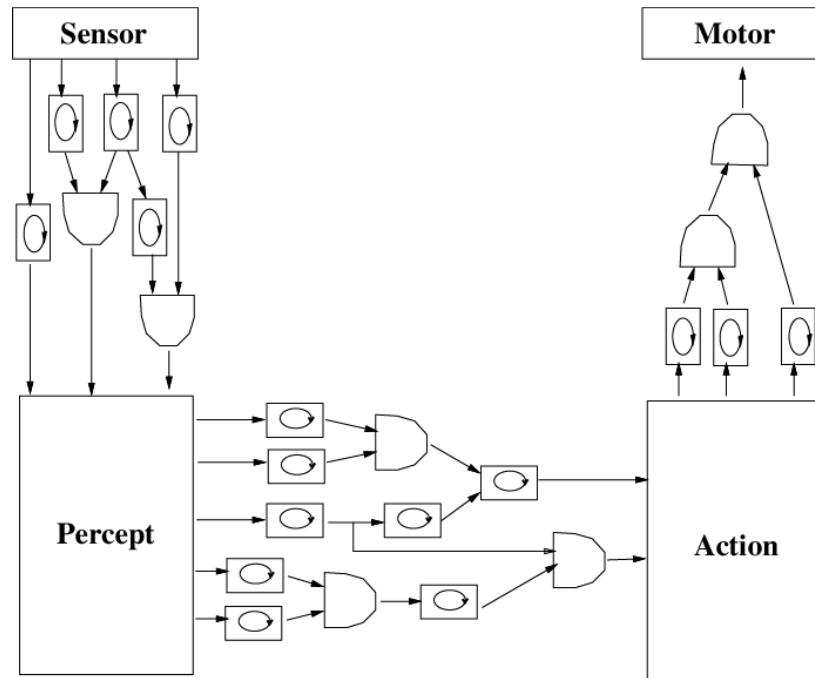
# 並行ルール判断 プロセス型



- Parallel Evaluation of Sensor States
- Action Arbitration



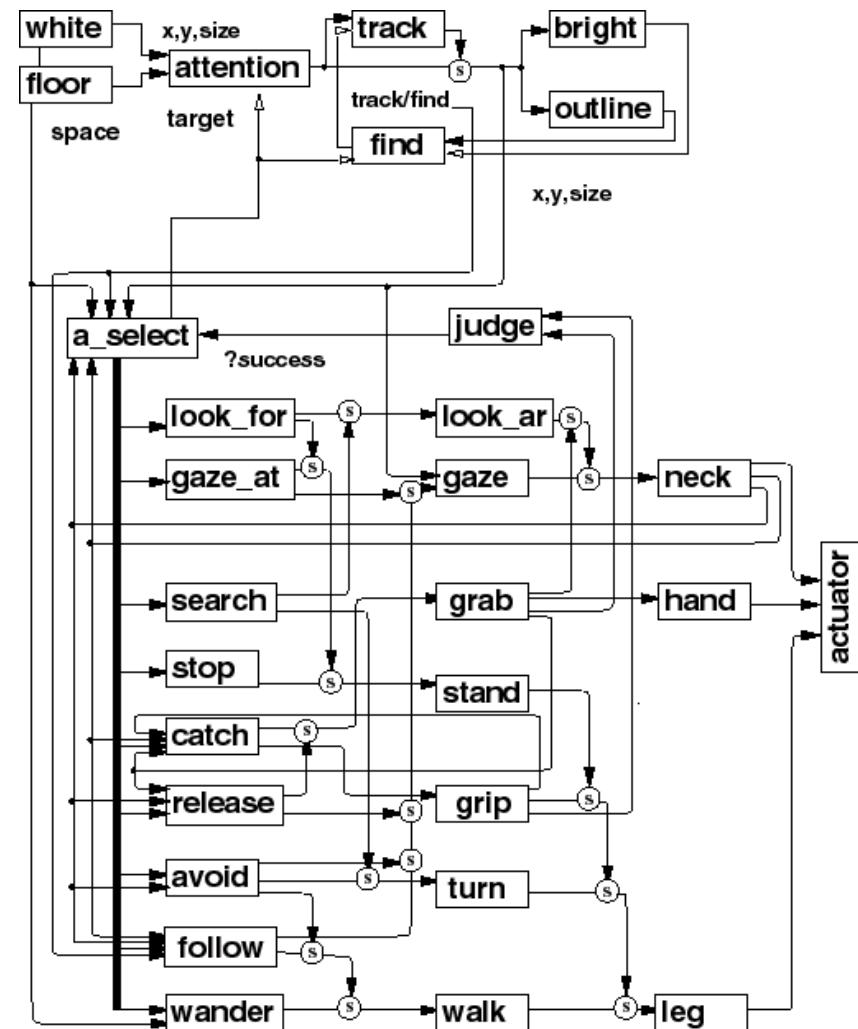
# プロセス群のネットワーク型



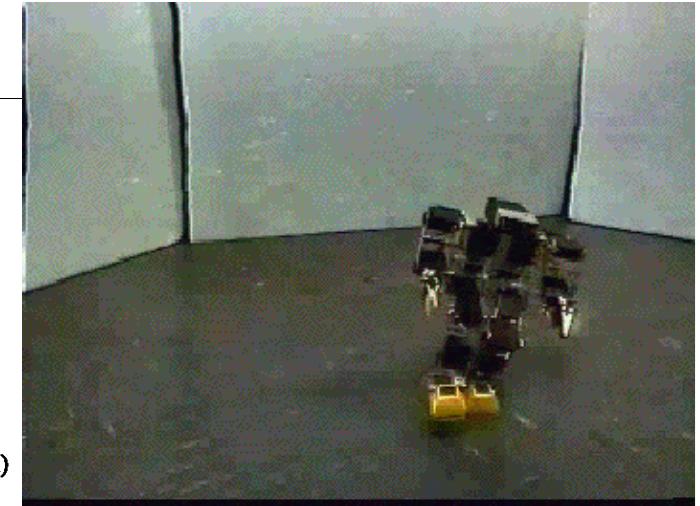
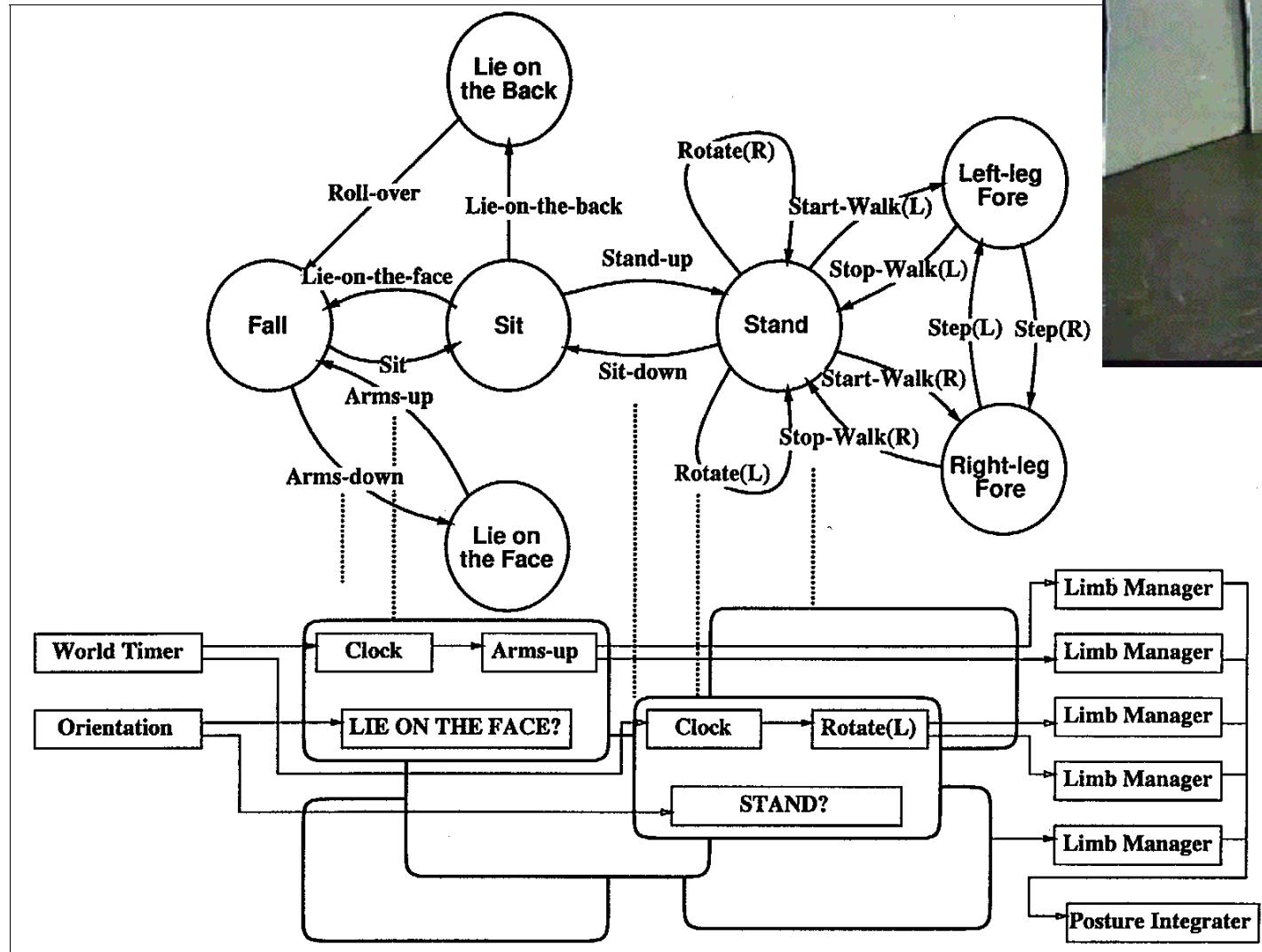
- Data-driven control,  
extensible with modularity
- Behavior Network (BeNet)



1995



# エラー復帰のためのプロセスネットワーク型 ステートネット



上位:  
State Net

下位:  
Behavior Net

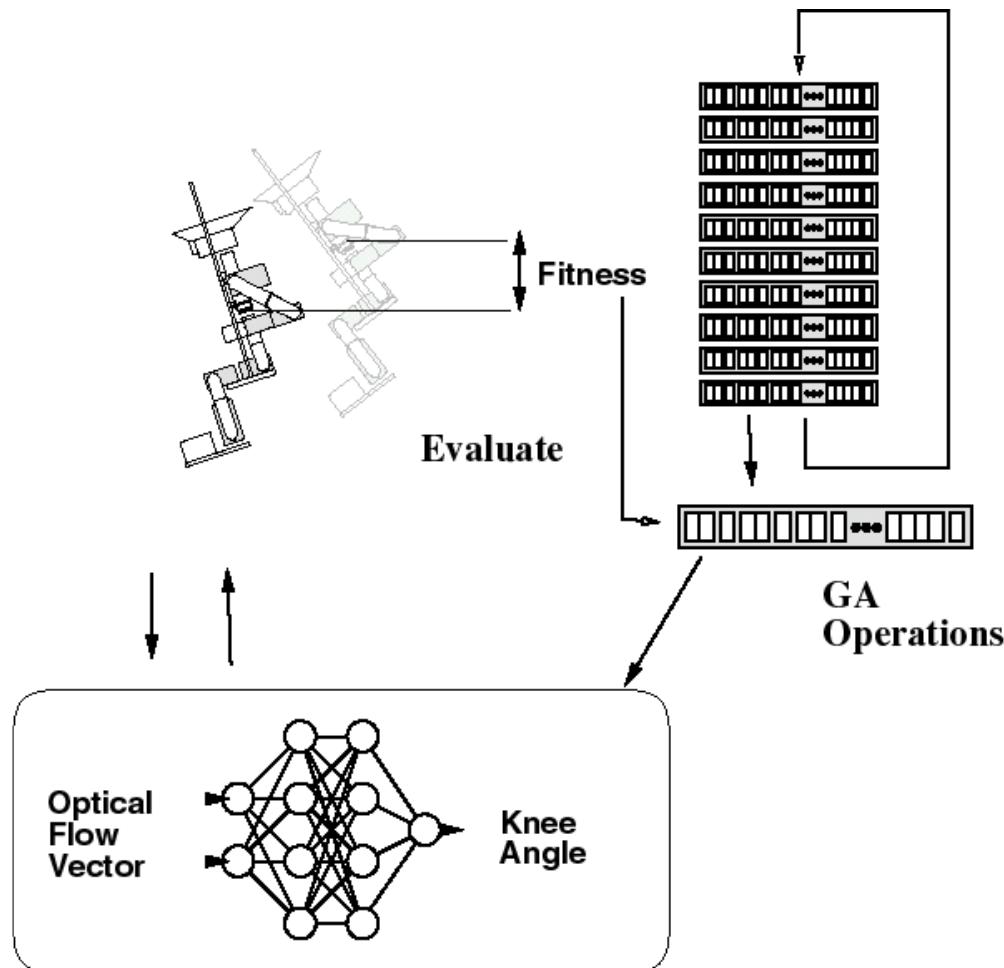
1995

# 歩くだけでなく、全身行動を 1994

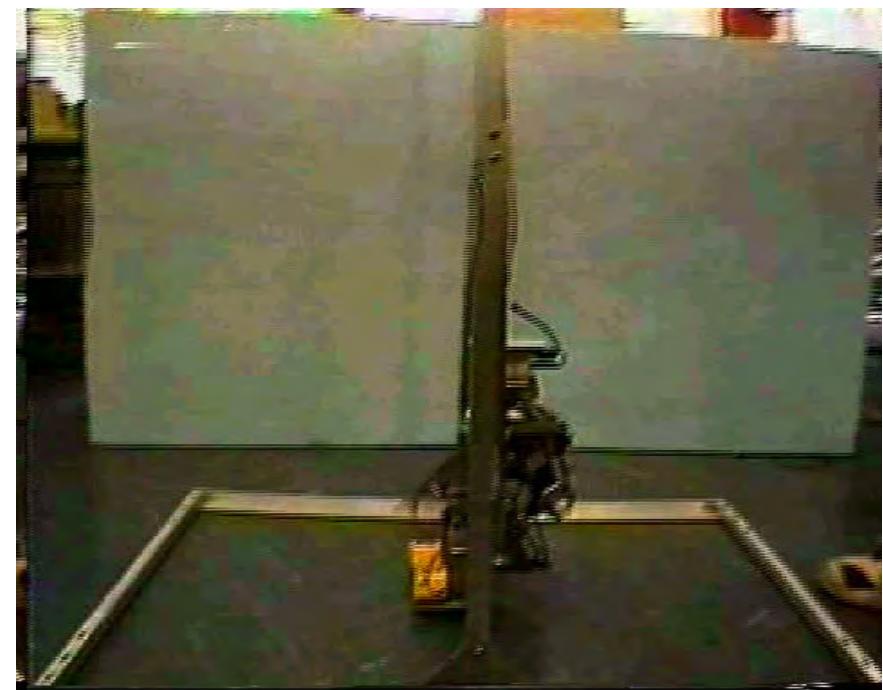


無線により、動きが自由になり、大きな計算機環境を利用可能

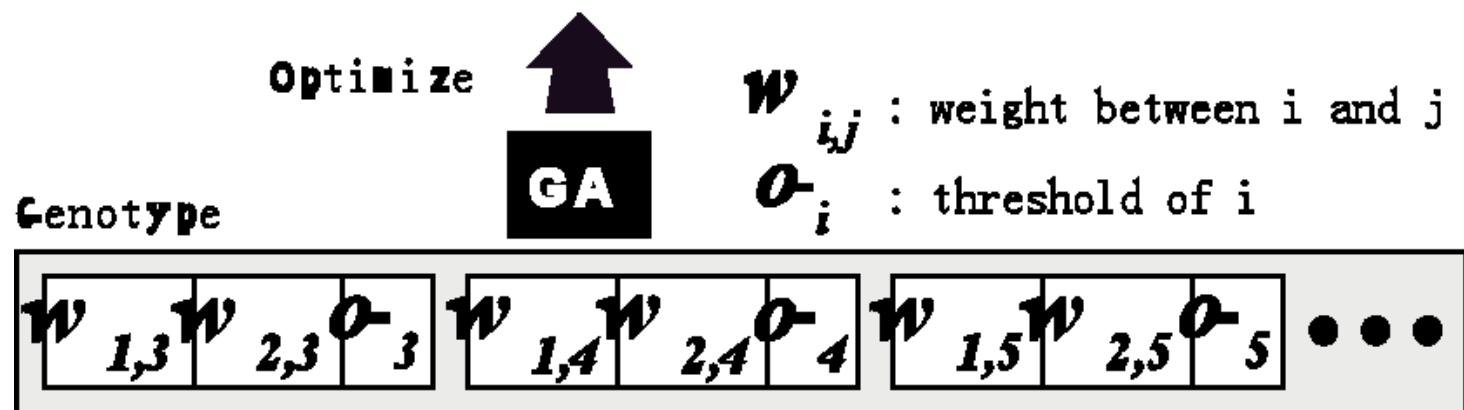
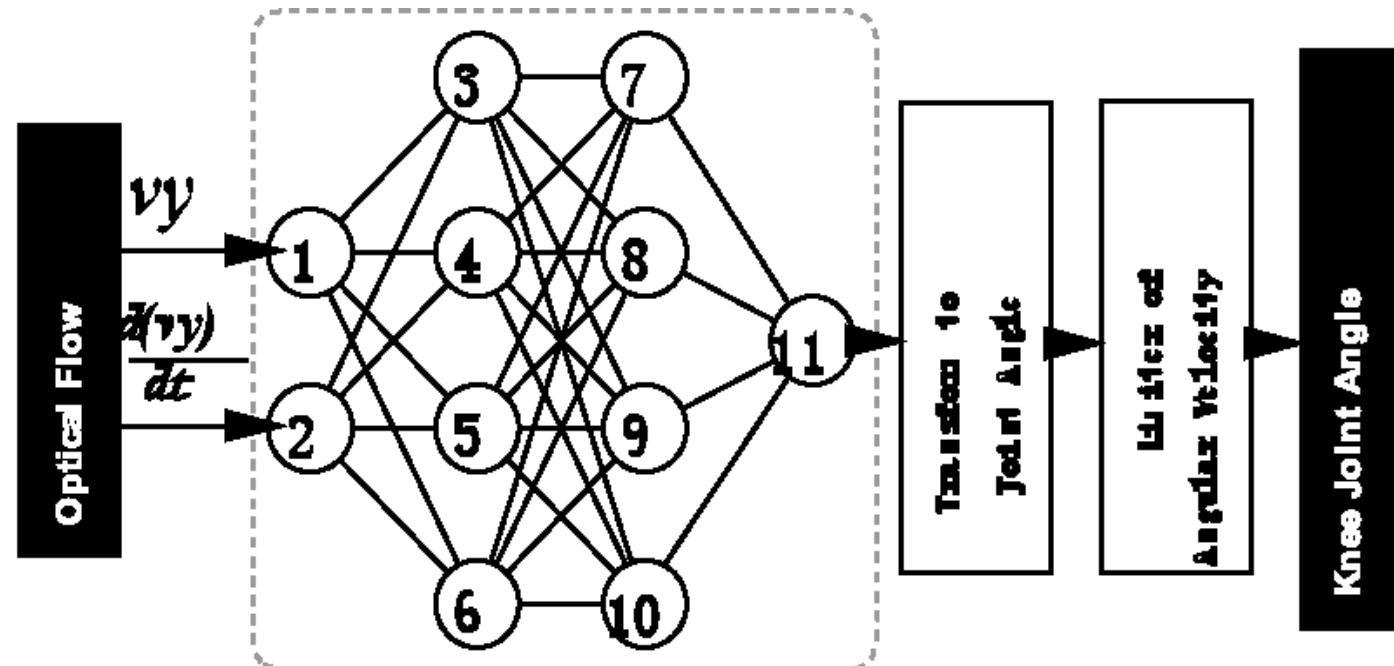
# 探索的手法による基本動作生成



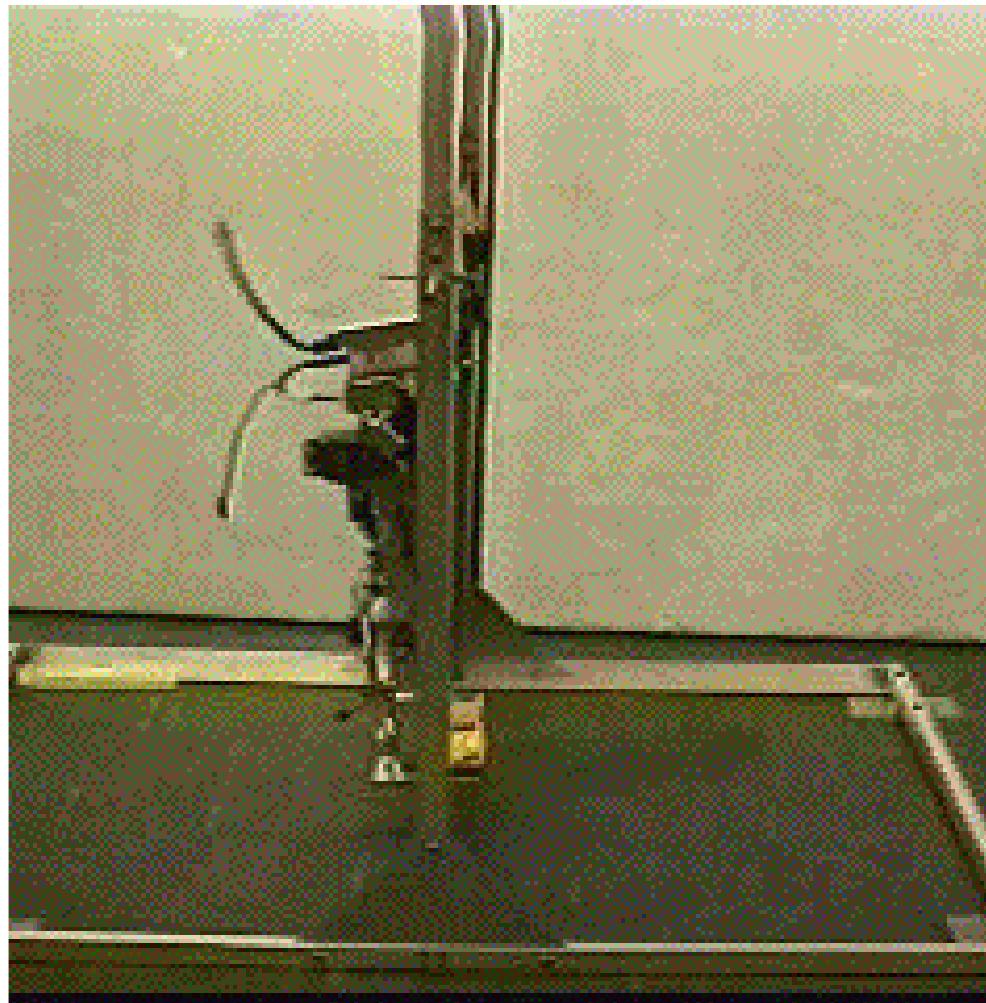
- 遺伝的アルゴリズムと  
ニューラルネットの組み  
合わせ

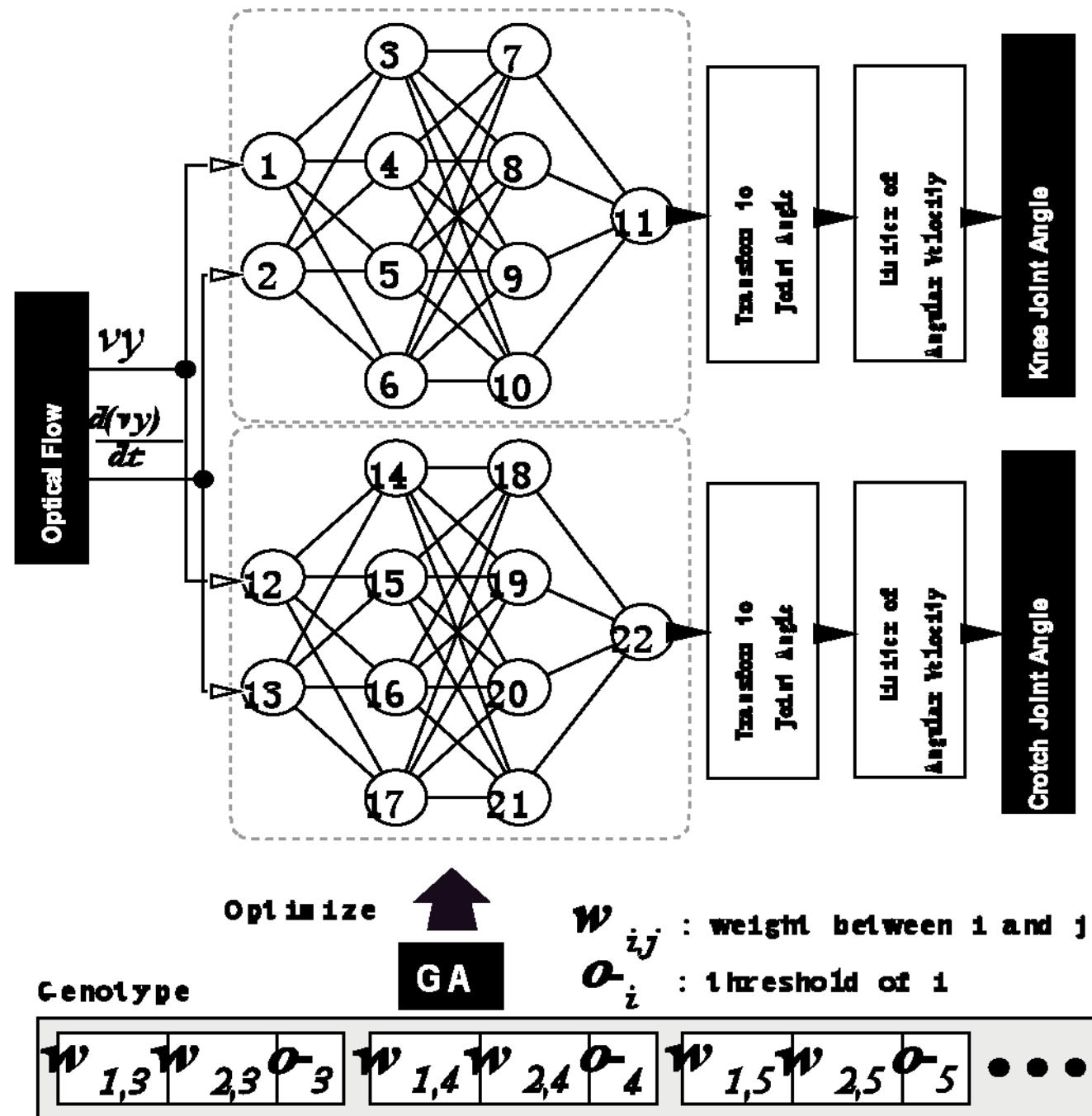


# ブランコ動作のプログラム



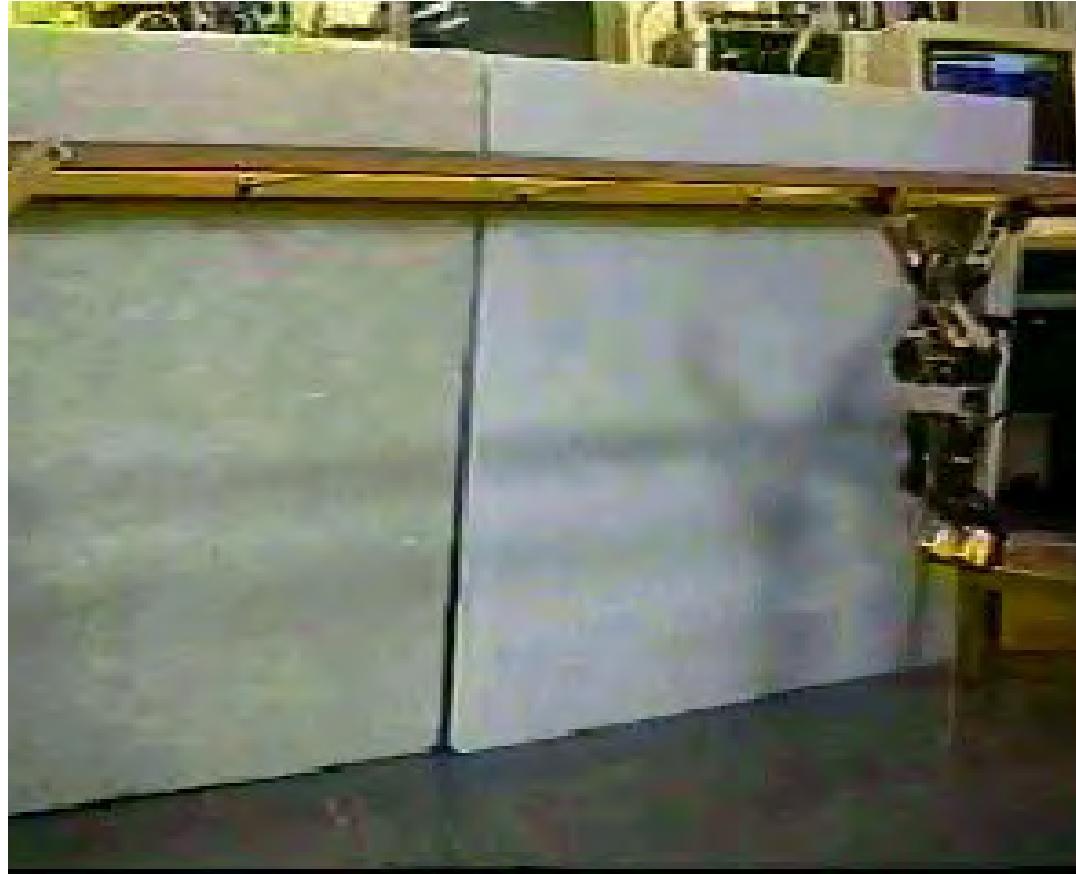
# 仮想環境内での イメージトレーニング





## 鉄棒の動作 プログラム

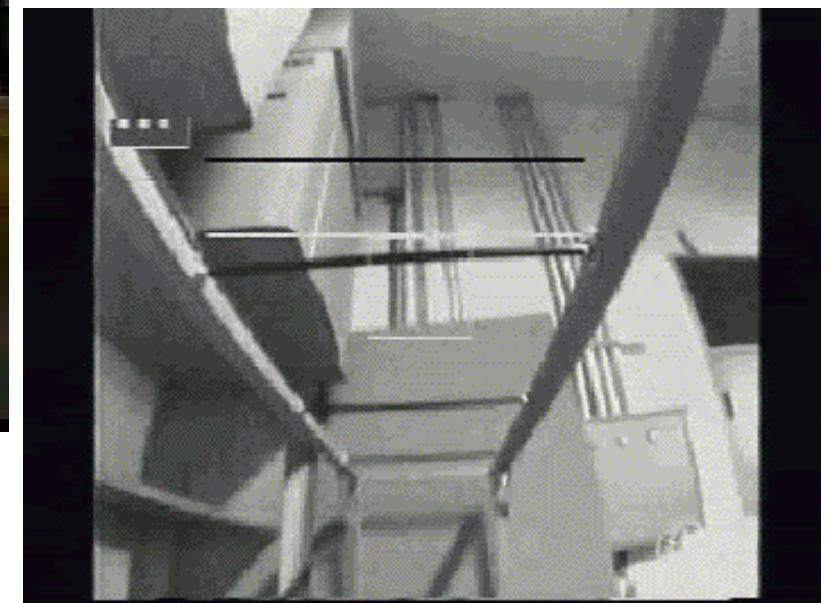
# もっと複雑な連携動作も 1997



何段かの運梯を渡ってゆく

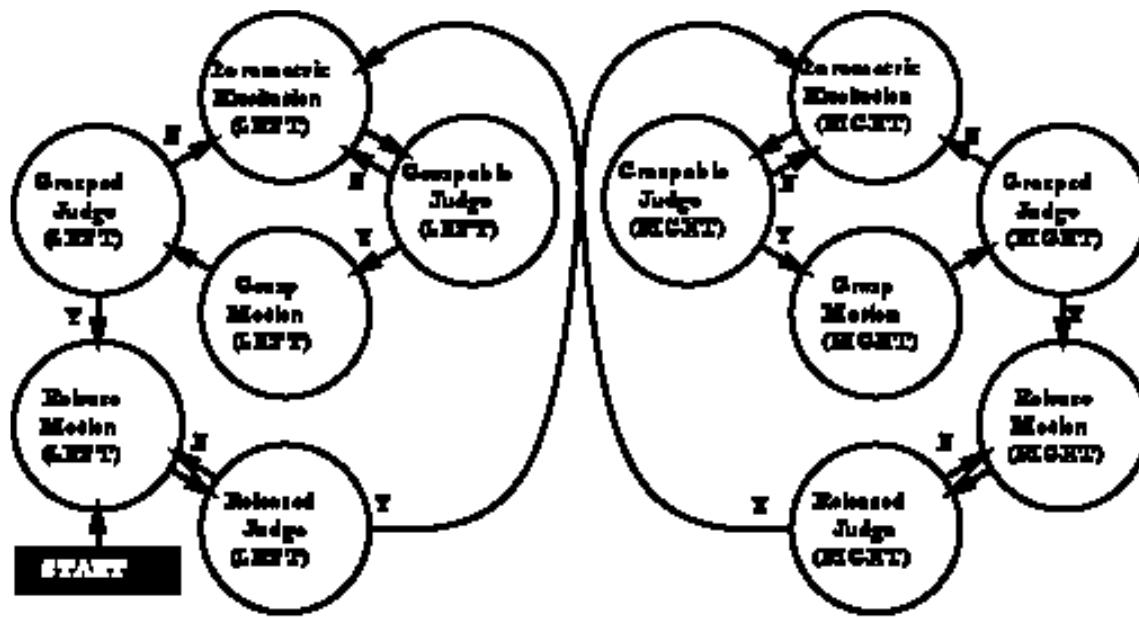


脚を使って揺れを増大

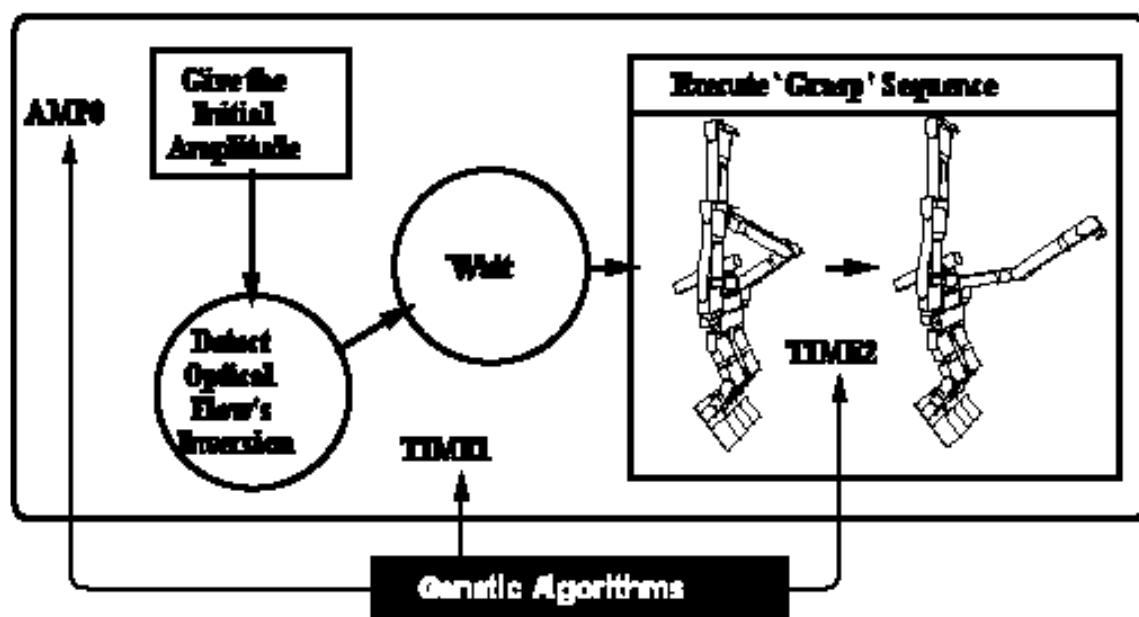


ロボットが見てている映像

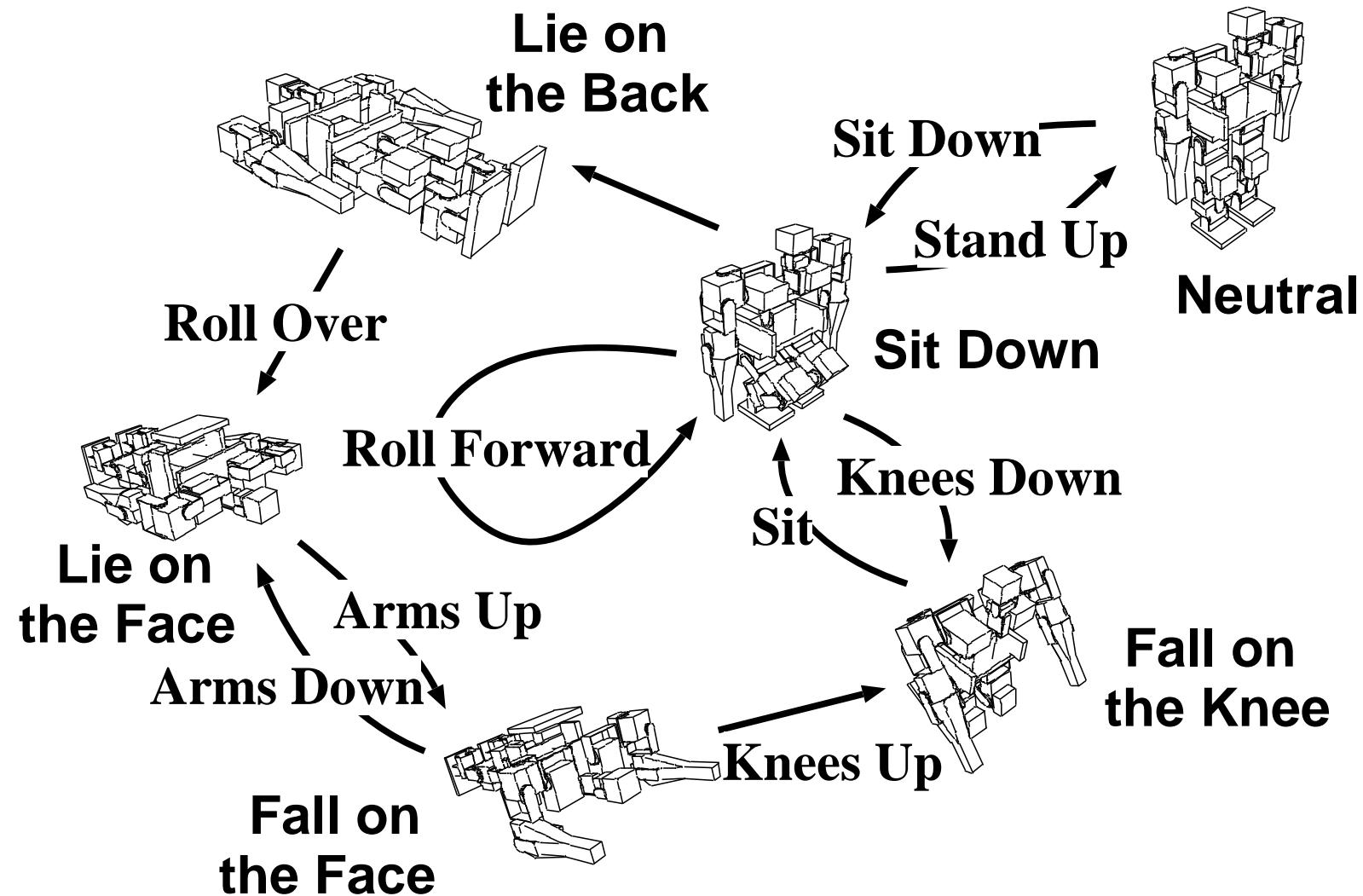
# 運梯動作 ブラキエー ション 動作の プログラム



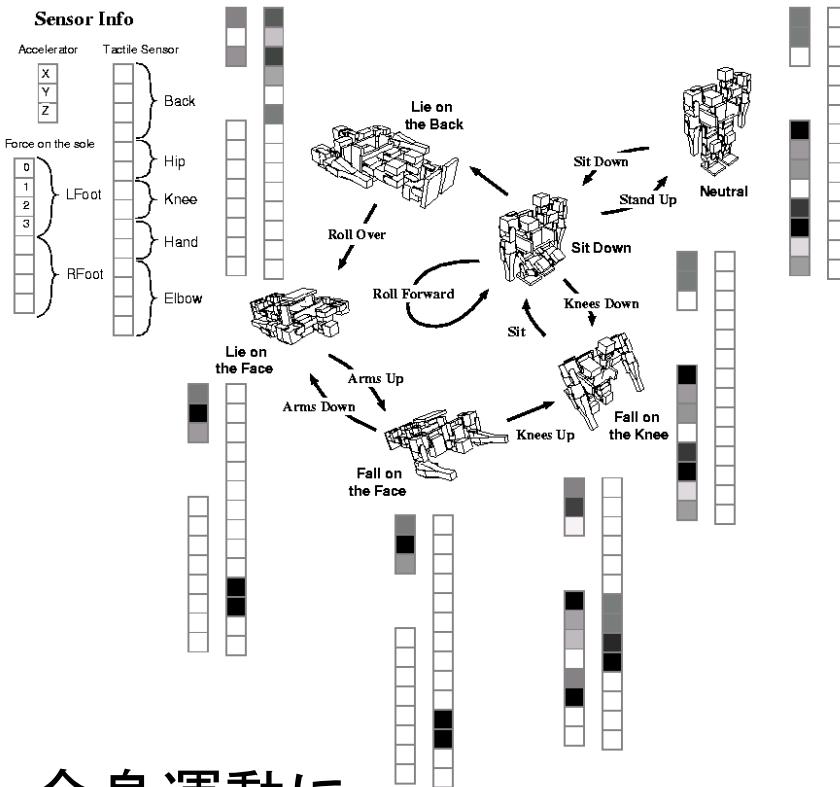
ex. Grasp Motion



# 倒れても起き上がる行動 ステートネット



# 動作のステートネットに基づく自動リカバリー行動の生成



全身運動における感覚空間における状態遷移状況

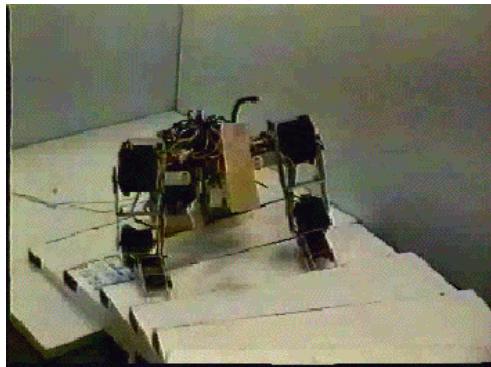


状態空間: 全身分布触覚, 姿勢情報の記述

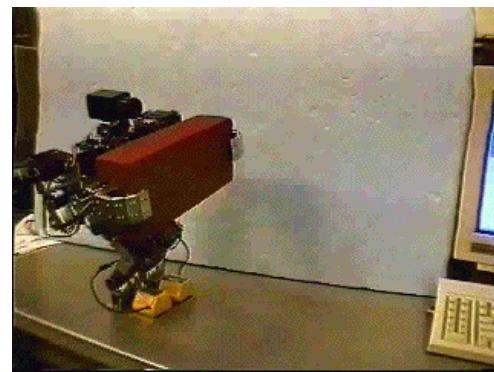


任意の状態から始まっても目標状態へたどりつく道を見つけることができる

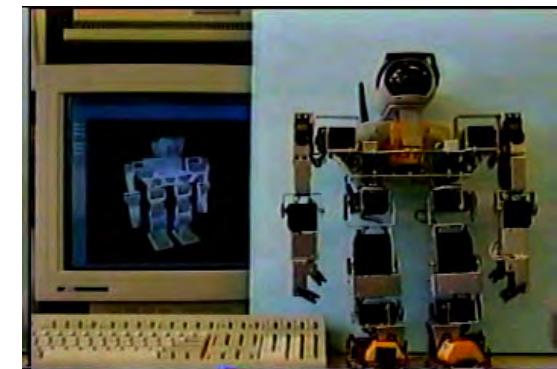
# 人間型の成長と進化の系図 1993–1997



1993 視覚誘導



1995 転倒復帰



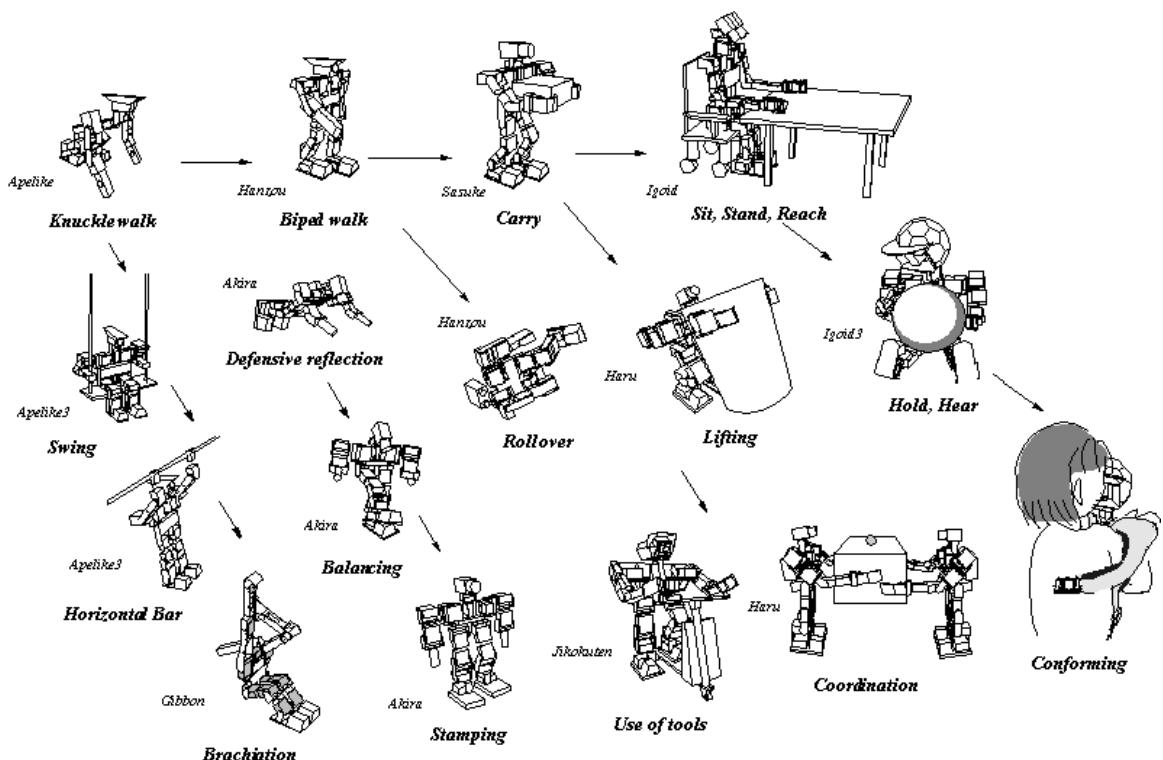
1996 体内神経系



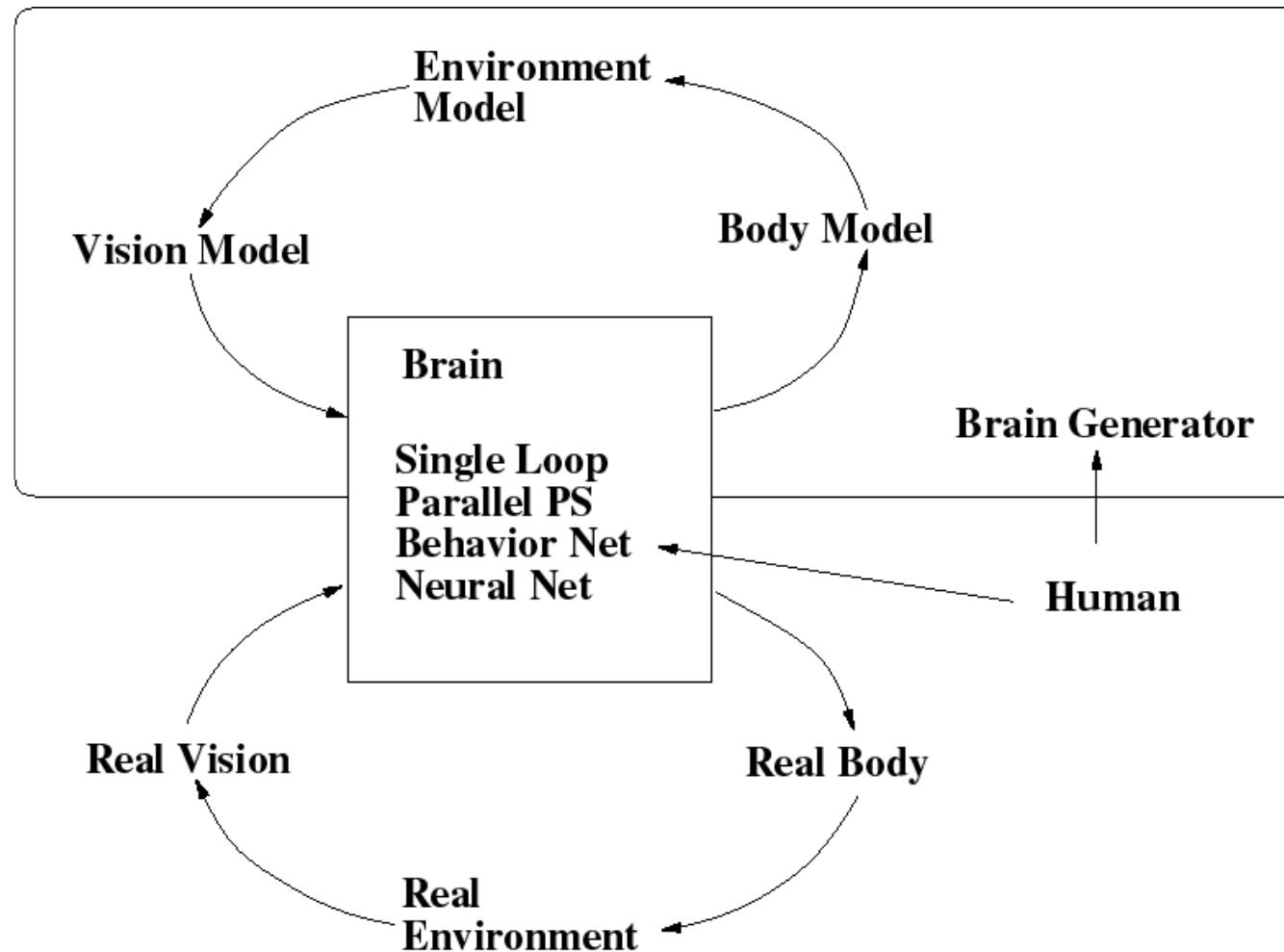
1994 動的運動制御



1996 全身行動制御



# 異なるブレインモデルを 同じボディに試すことができる環境



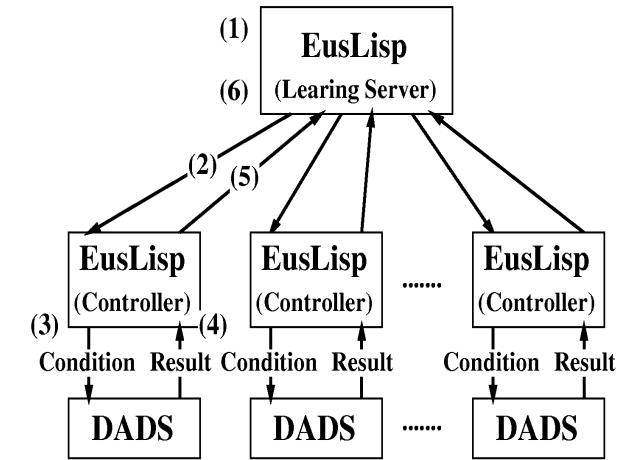
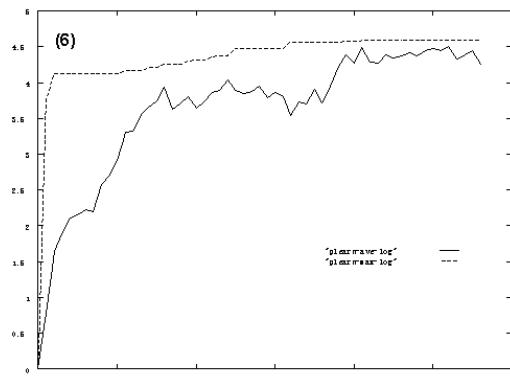
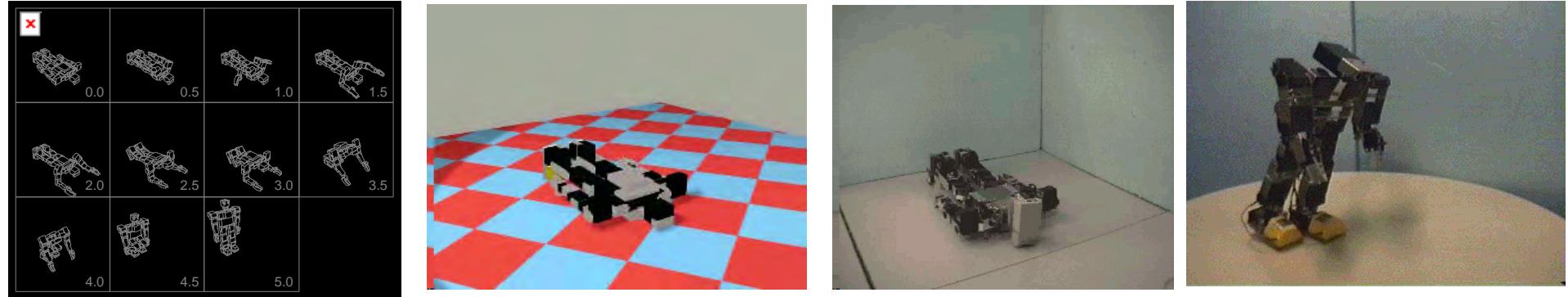
- (0) One-path neural network
- (1) Single Process
- (2) Parallel Process
- (3) Networked Processes  
(BeNet:  
Network of  
Behavior Unit)

# ロボットのシミュレーション環境

- シミュレーション環境
  - Microsoft Robotics Development Studio 2008
  - Player, Willow Garage ROS
- 物理シミュレーション
  - Havok
  - ODE: Open Dynamics Engine
  - PhysX (AGEIA), Rocket
  - sDIMs, OPENHRP
- プランニングシステム
  - PDDL (プランナ記述言語)、ff (前向き計算プランナ)
  - OpenRAVE (RRTプランナ)

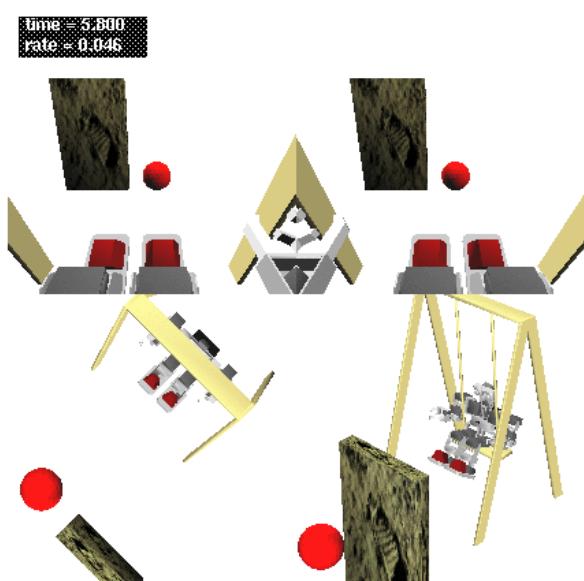
# 開発環境(Mother Environment)

- Trials in Various Feasible Motions
- Brain with Automatic Motion Search Mechanism
- Software Growth with Connecting Commercial Software

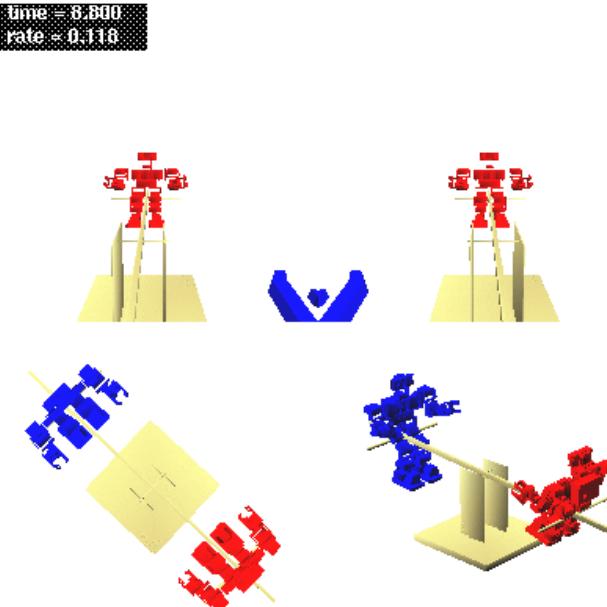


# ロボット行動プログラム例

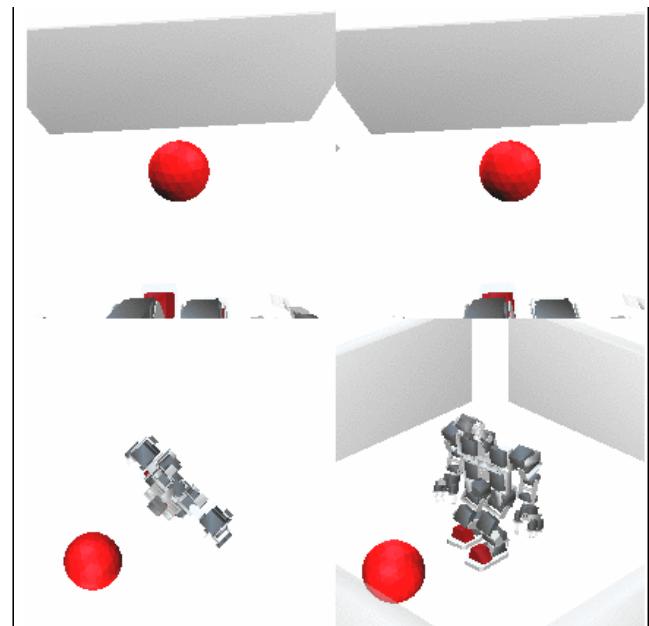
- ・ 視覚情報も同時にシミュレーション可能



ブランコこぎ



2台のヒューマノイド  
によるシーソー

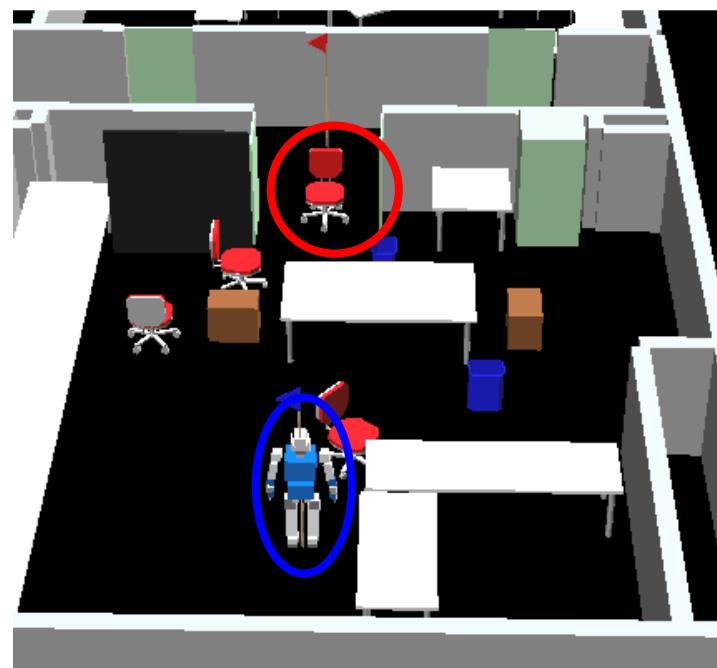
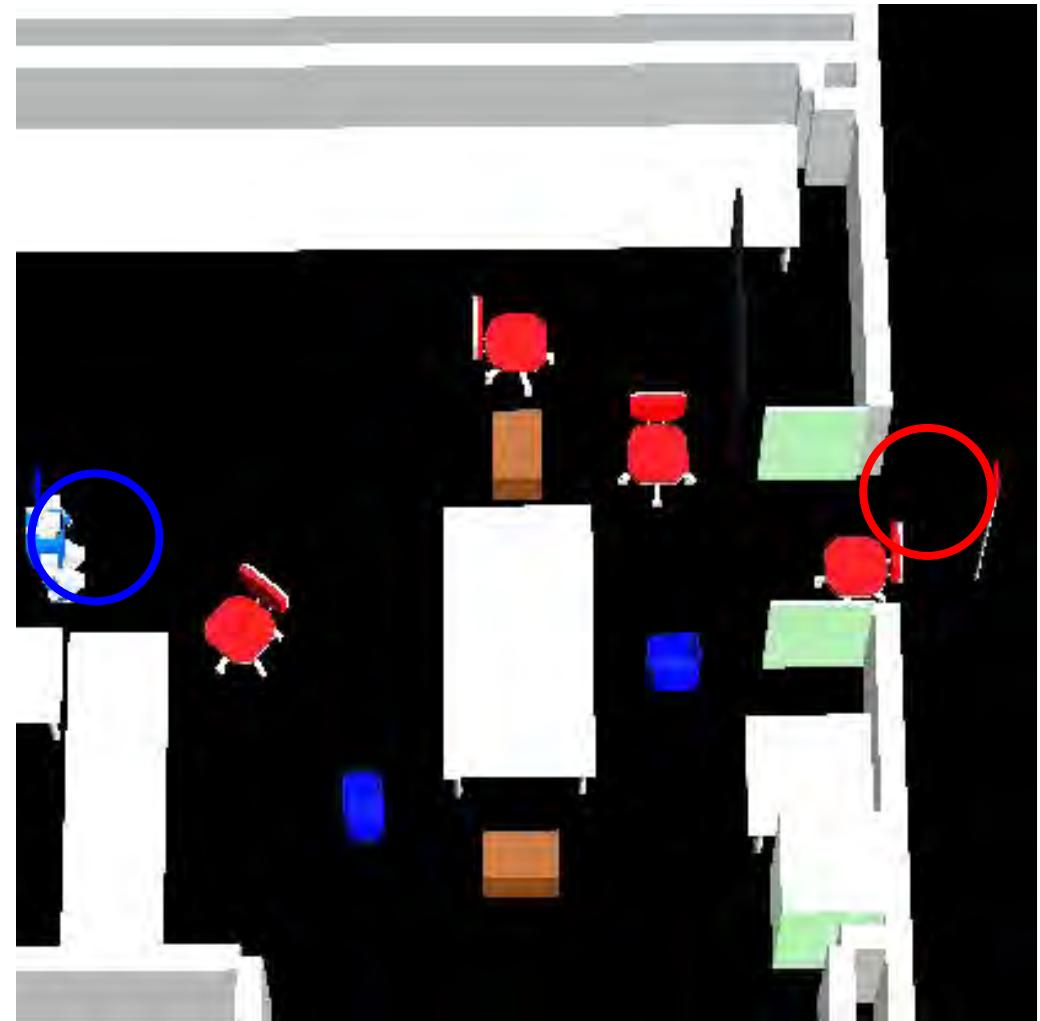


ボールを追いかけて蹴る

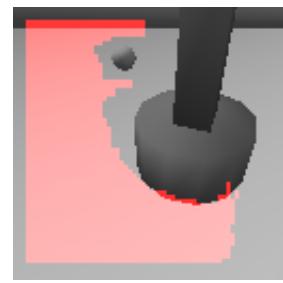
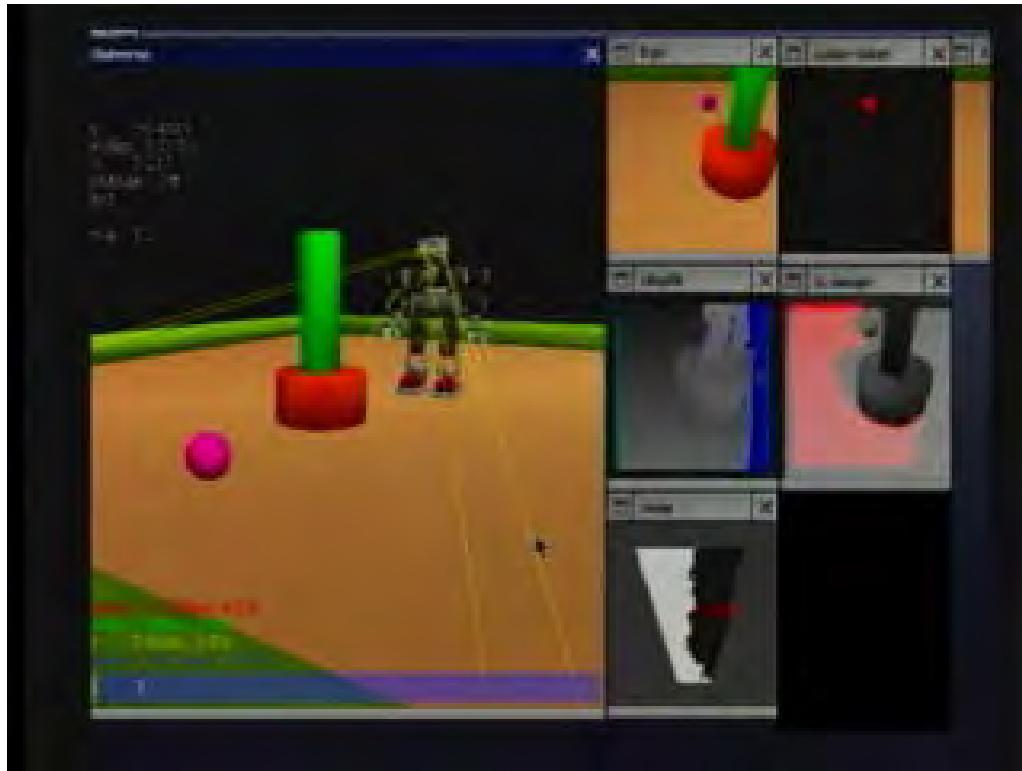
Mathengine -> ODE

1999-

## 可動物体の操作を考慮した行動計画生成



# Vision Based Motion Planner

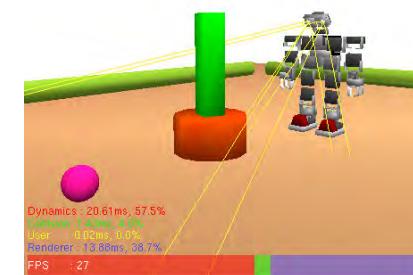


Ground Plane  
Detection



Generated plan

- Plane Segment Finder by Stereo Vision
- Path Finder on Ground Plane

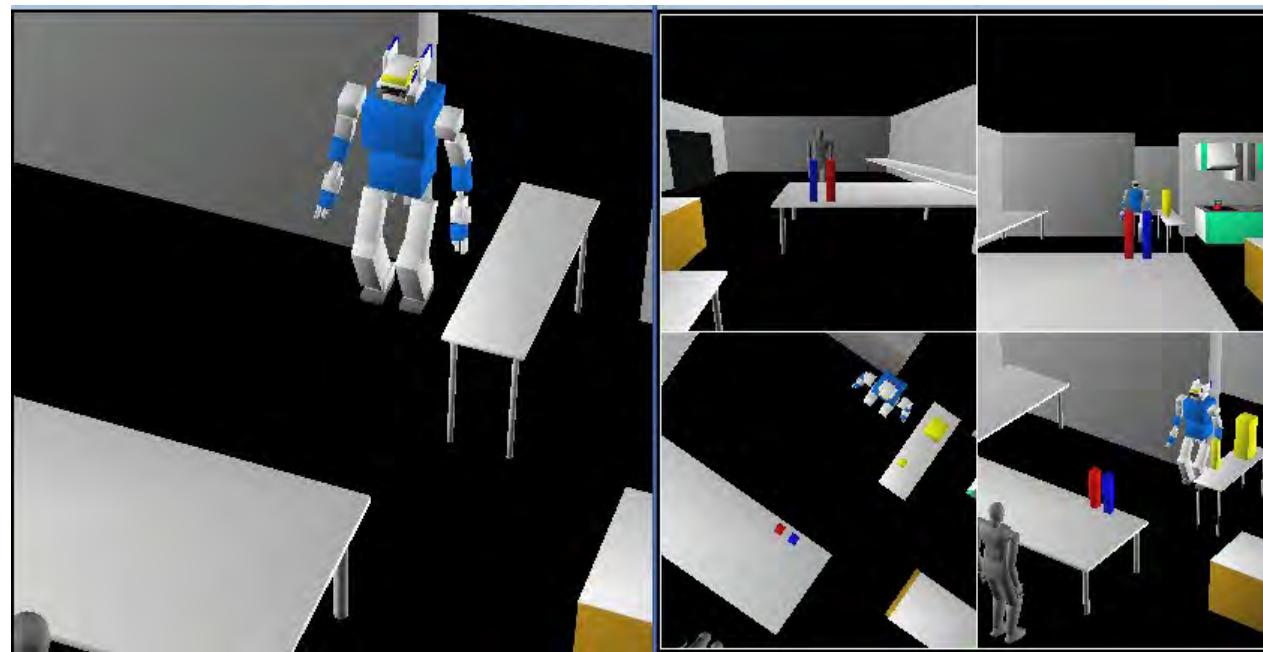


2002

Goal: Hold the Ball

# 対話・記憶・行動計画のシステム統合環境

- 状況に応じた質問・確認などの発話生成
- 視聴覚感覚情報に基づくオンライン空間表現形成  
とそれにに基づく対話のマネージメント
- オンサイト指示、動作計画系との統合基盤

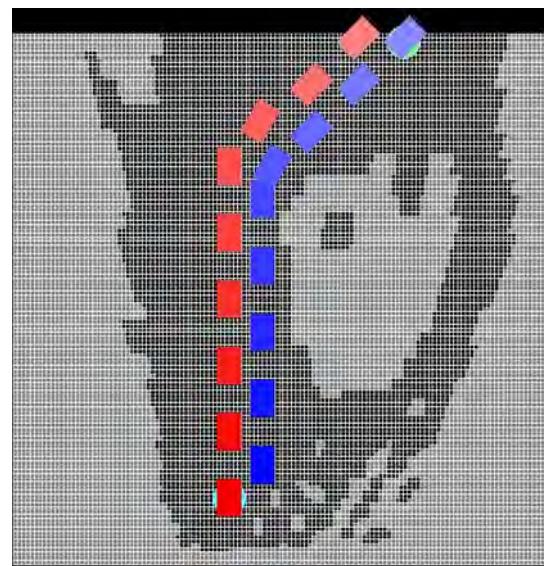
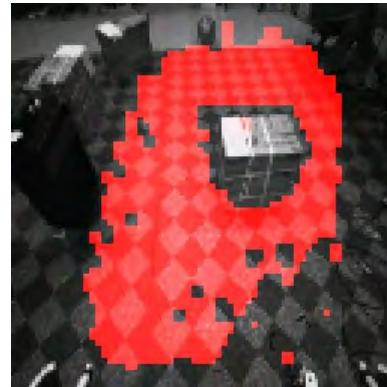


# 障害物回避歩行実験



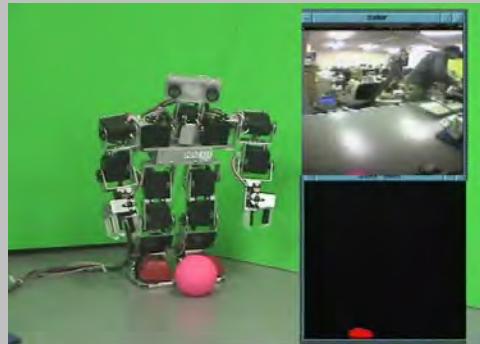
前方に近い、移動可能方向に目標移動を計画

# Vision Based Path Planning



Foot Step Plan

## 実身体



実時間制御層

# ヒューマノイドの ロボットシステム

## 認識判断行動生成系

- |      |        |
|------|--------|
| 対話制御 | 行動予測   |
| 感覚制御 | 物体認識   |
| 空間推論 | 移動プランナ |
| 行動推論 | 動作プランナ |
| 行動監視 | 動作制御   |

## 記憶系

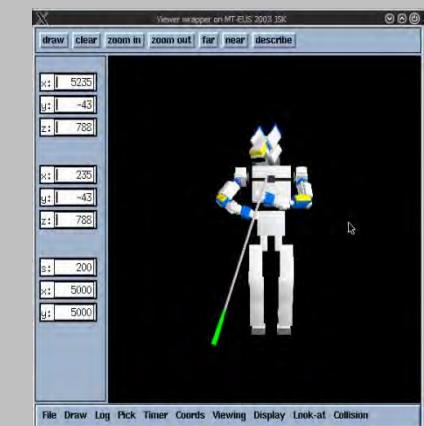
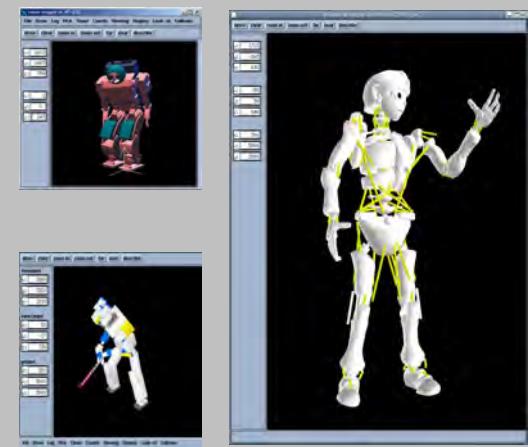
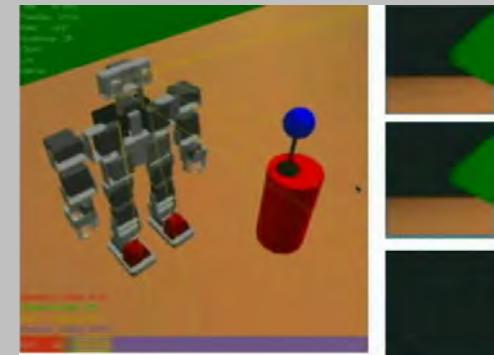
- 空間物体身体モデル  
目的行動モデル  
判断結果獲得データ

## 自律反応身体制御系

- 身体状態監視  
疲労度監視 順応 反射

感覚系 動作系

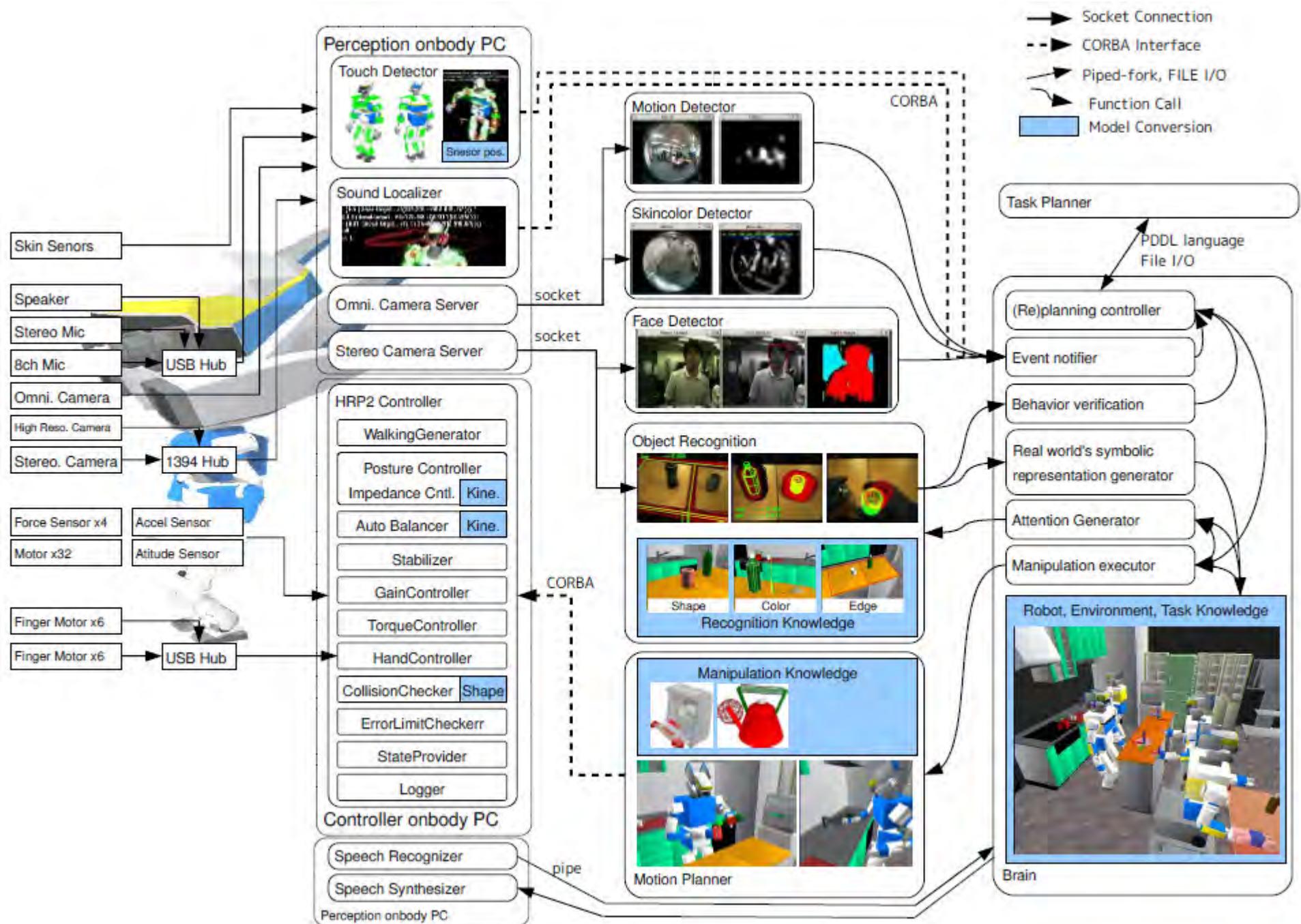
## 仮想身体



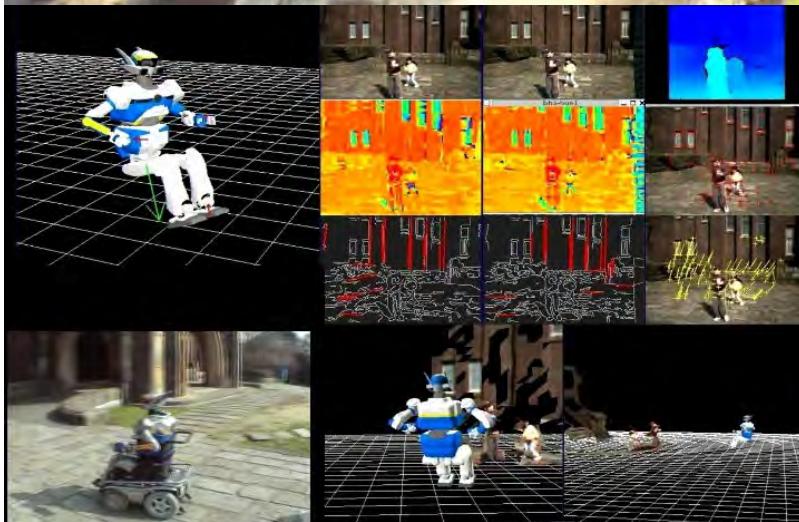
# 产学連携 IRT研究 2006



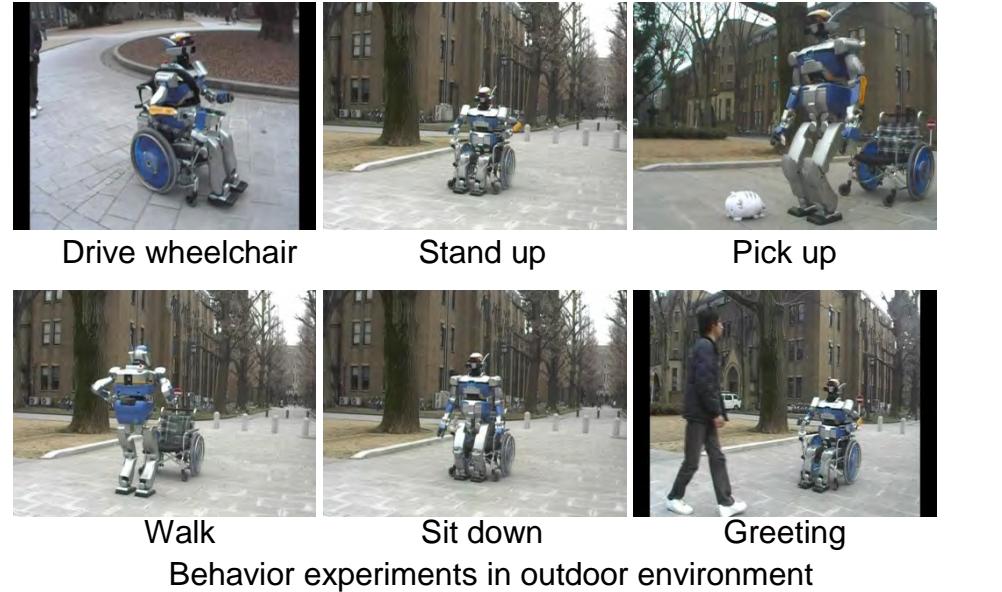
台所での支援行動  
2007.7.16 TV東京 カンブリア宮殿



# ヒューマノイド 手足のある計測計算機



Driving wheel chairs to acquire sensory information

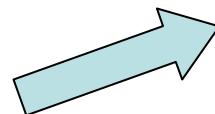
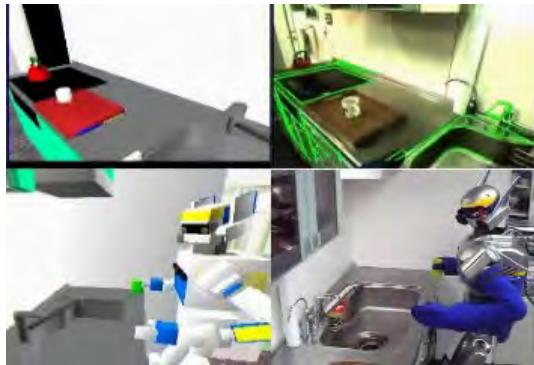


Falling down is unavoidable situation



-> self protection external skin

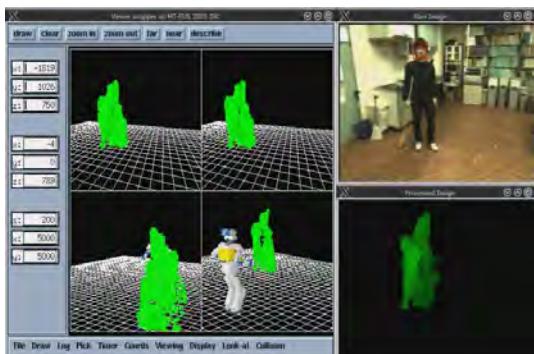
# ヒューマノイドの要素機能を発展応用



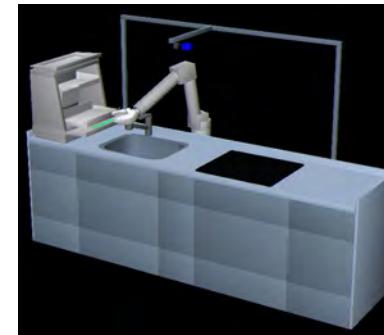
視聴覚応用



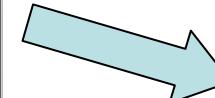
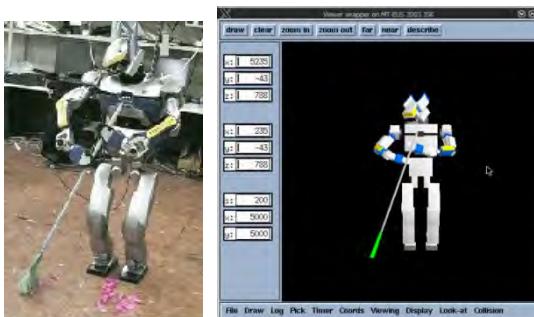
見守り: 富士通



MEMSセンサ  
手探し  
物体操作機能



キッチン支援: パナソニック



環境認識  
道具理解  
行動管理機能



掃除・片付け: トヨタ

ヒューマノイドにおける  
要素機能統合環境

# 要素機能応用の開発成果をヒューマノイドへ統合し維持



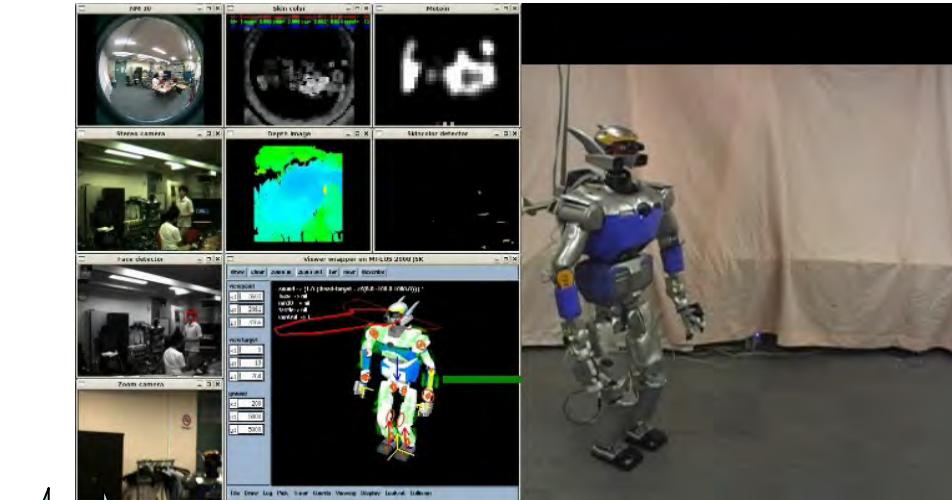
運搬 家事支援



見守り 思い出し支援

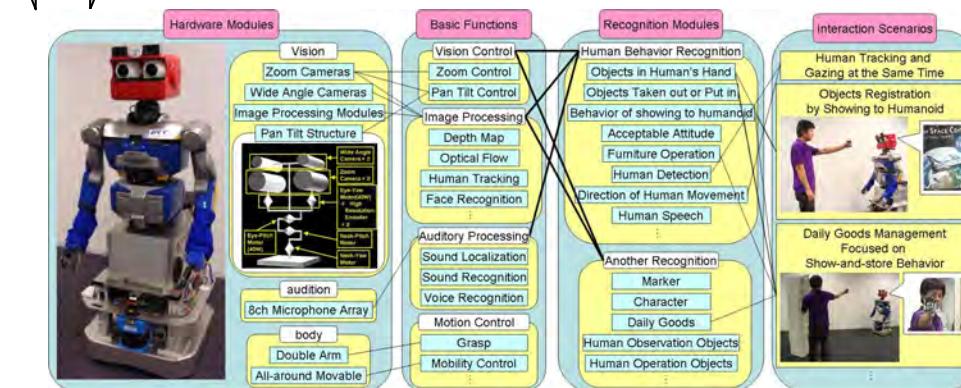


高機能化 家具・家電



HRP2JSK-NT

2脚  
2腕  
全身触覚  
聴覚

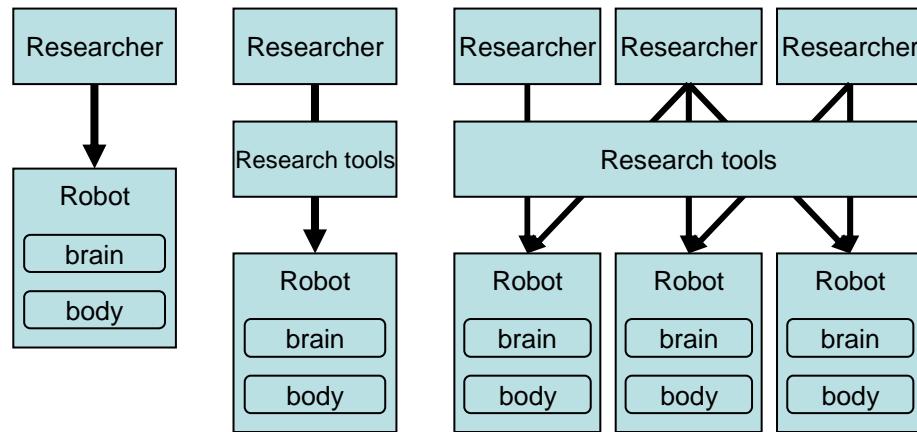


HRP2VZ

全方位車輪  
広視野  
高視力

**ヒューマノイドの要素機能**  
 -> 産学連携 企業と応用開発  
 -> 強化された要素機能をヒューマノイドに統合

# ロボットソフトウェアの発展的構成法研究



1993 ISRR

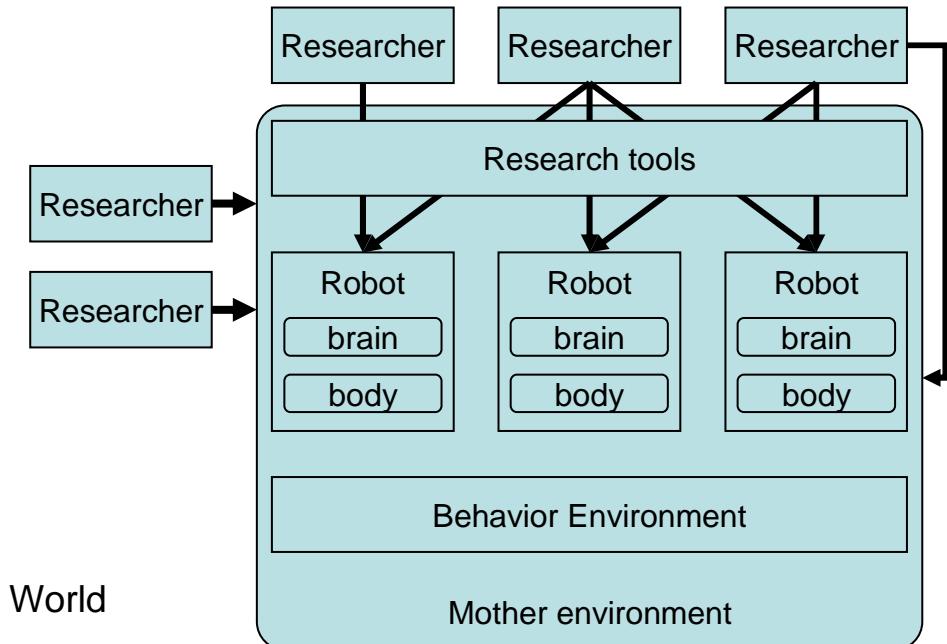
Masayuki Inaba:

Remote-Brained Robotics: Interfacing AI with Real World Behaviors,

in Takeo Kanade and Richard Paul (Eds.): *Robotics Research*, Vol.6, pp.335--344, The International Foundation for Robotcs Research, 1994.

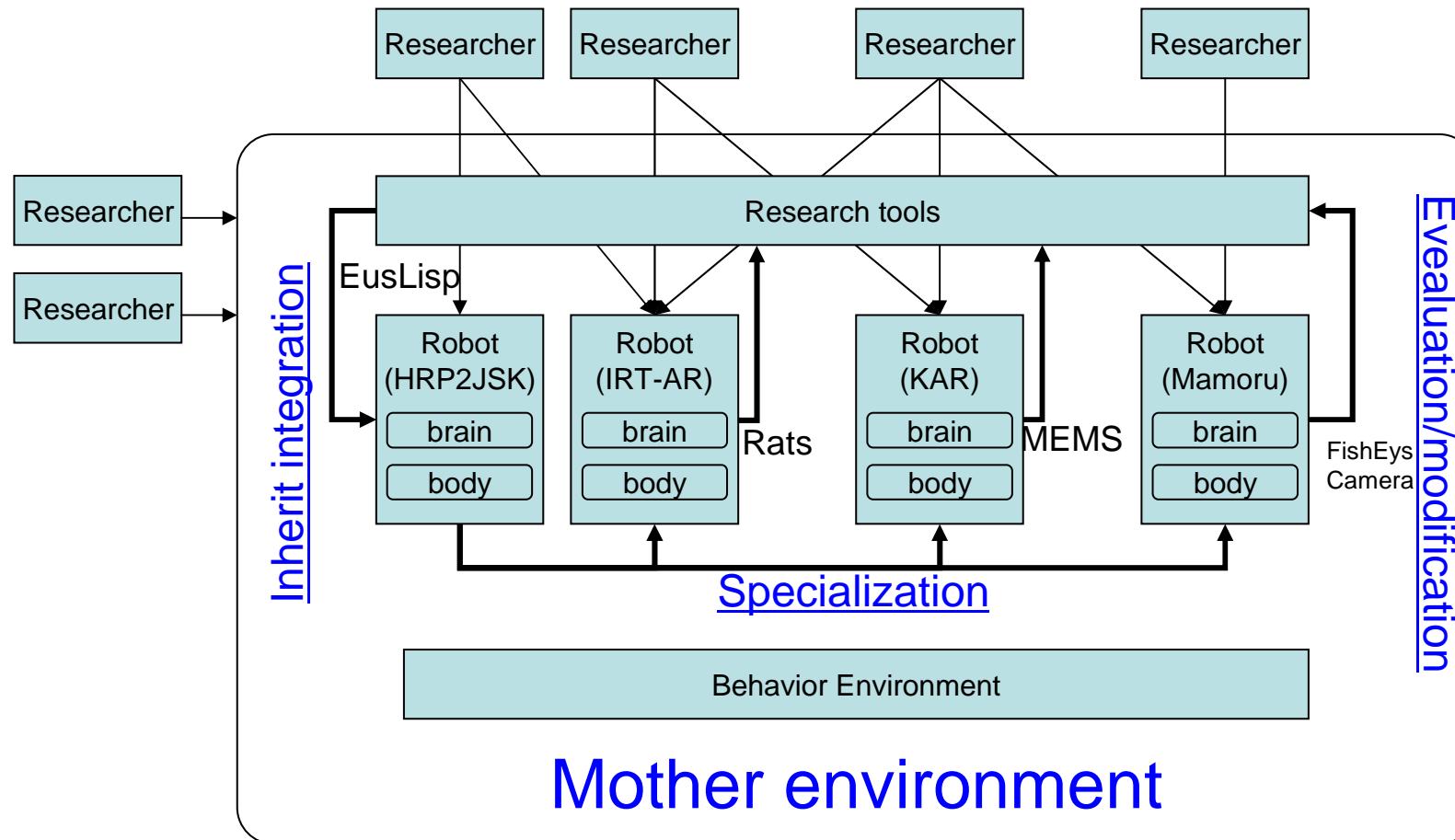
M. Inaba, K. Okada, T. Yoshikai, R. Hanai, K. Yamazaki, Y. Nakanish, H. Yaguchi, N. Hatao, J. Fujimoto, M. Kojima, S. Tokutsu, K. Yamamoto, Y. Kakiuchi, T. Maki, S. Nozawa, R. Ueda, I. Mizuuchi:  
Enhanced Mother Environment with Humanoid Specialization in IRT Robot Systems,

in Cedric Pradalier and Roland Siegwart and Gerhard Hirzinger (Eds.): *Robotics Research: The 14th International Symposium ISRR*, pp.379--396, Springer, 2011.



2009 ISRR

# JSK Mother Environment through Specialization and Integration

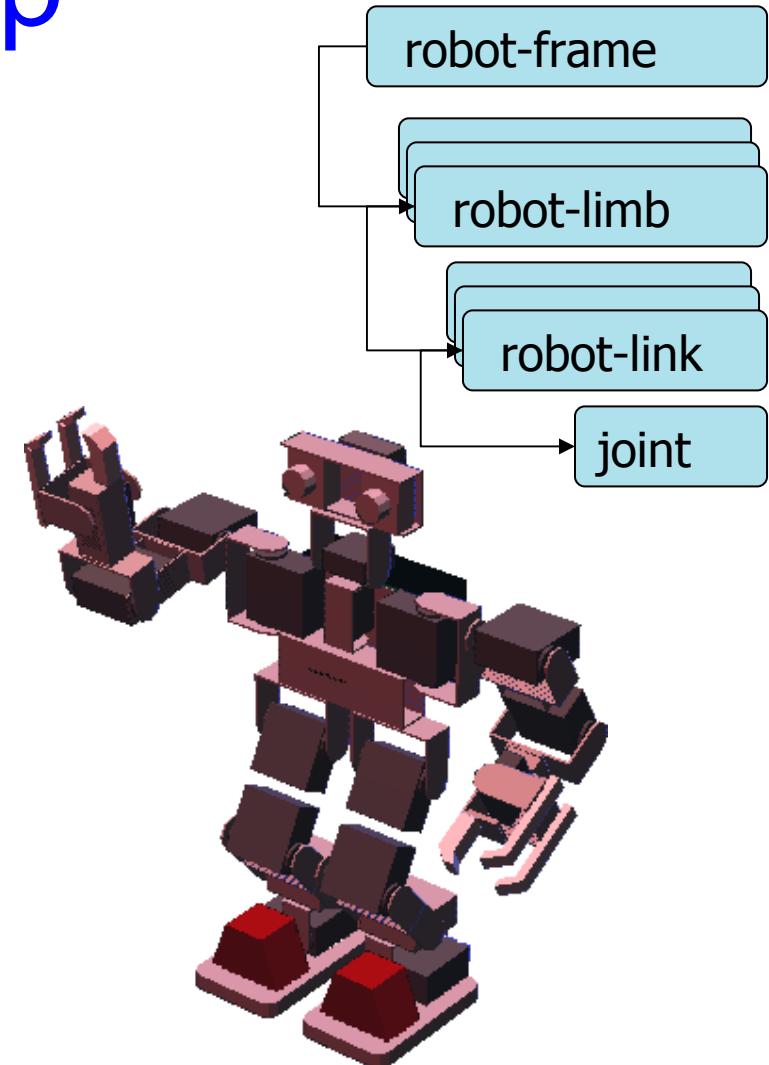


# ロボット用LISP言語: EusLisp

- 軽量言語 (Light Weight Language)
- 3次元形状モデル(3D Solid Modeling)
- 他言語インターフェース(Foreign Language Function)
- オブジェクト指向クラスシステム
- マルチスレッド, OpenGL, CORBA,,,,
- 1986 松井俊浩 @ 産業技術総合研究所
- BSD ライセンス
  - 検索: sourceforge euslisp
  - <http://sourceforge.net/projects/euslisp/>
- 東大拡張 jskeus
  - <http://sourceforge.net/projects/jskeus/>

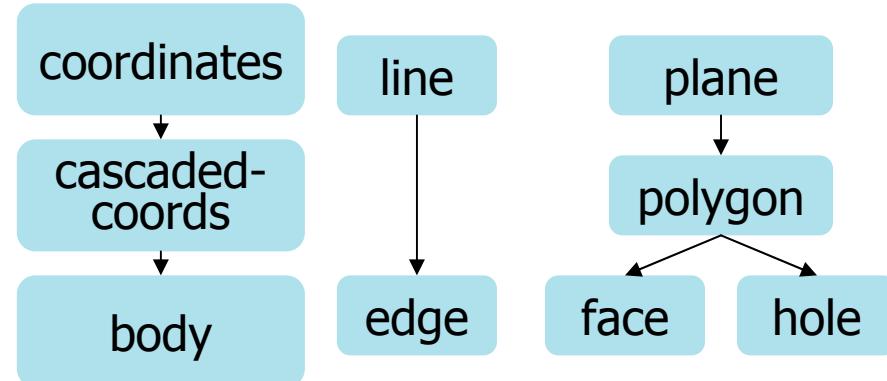
# Modeling humanoid robots in EusLisp

- Abstracted description of humanoid robots
- Joint space level commands
  - (send \*kaz3\* :larm :shoulder-p :joint-angle -80)
  - (send \*kaz3\* :larm :elbow-p :joint-angle -80)
- Work space level commands
  - (setq c (send \*kaz3\* :larm :end-coords :copy-worldcoords))
  - (send c :translate #f(30 0 30) :world)
  - (send \*kaz3\* :larm :inverse-kinematics c)



# ソリッドモデル in Euslisp

- coordinates
- :locate points
- :rotate theta axis
- Primitive Bodies
- (make-cube x y z)
- (make-prism points sweep)
- (make-cylinder radius height)
- Composition of Bodies
- (body+ body1 body2)
- (body- body1 body2)



Class hierarchy of solid modeler

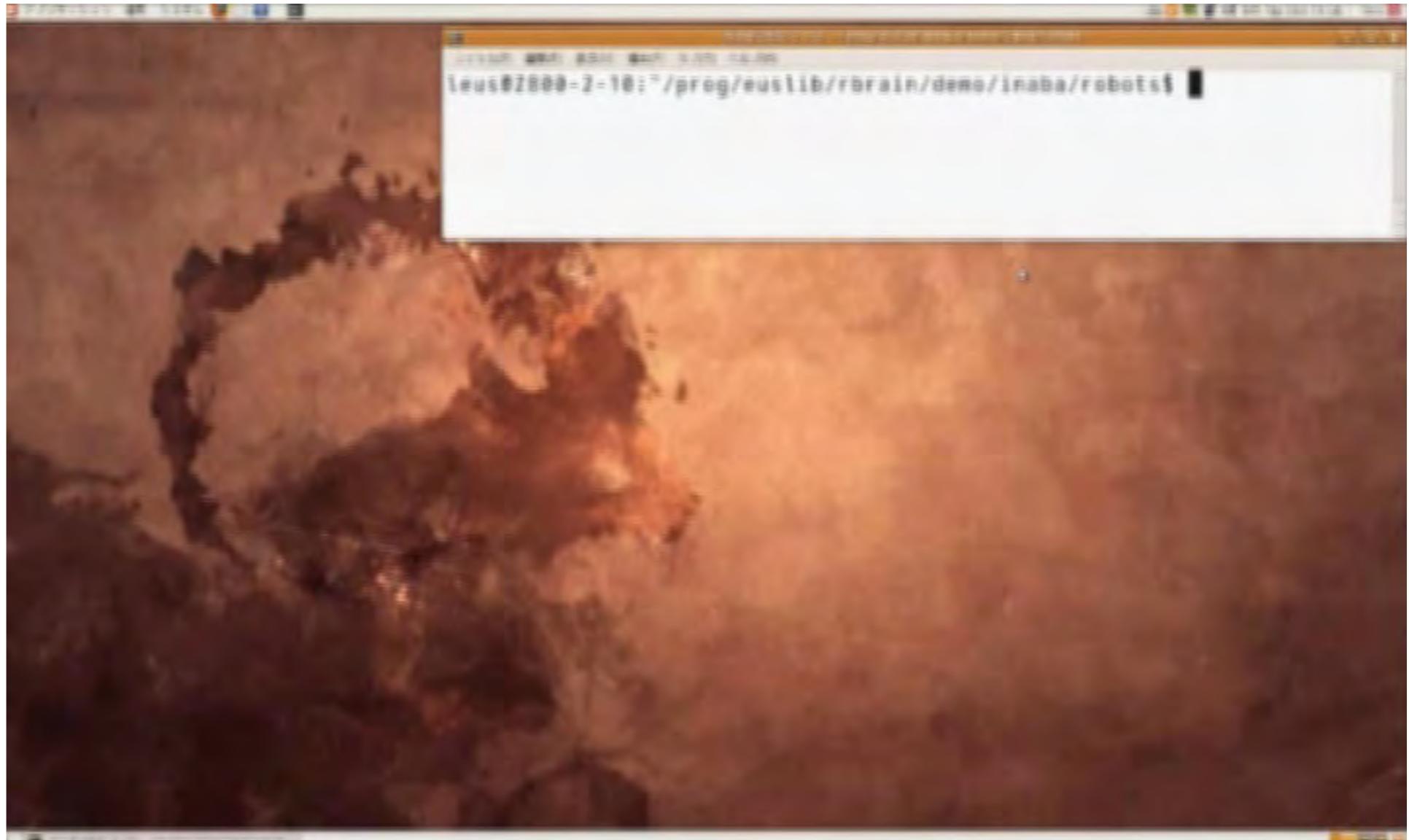
```

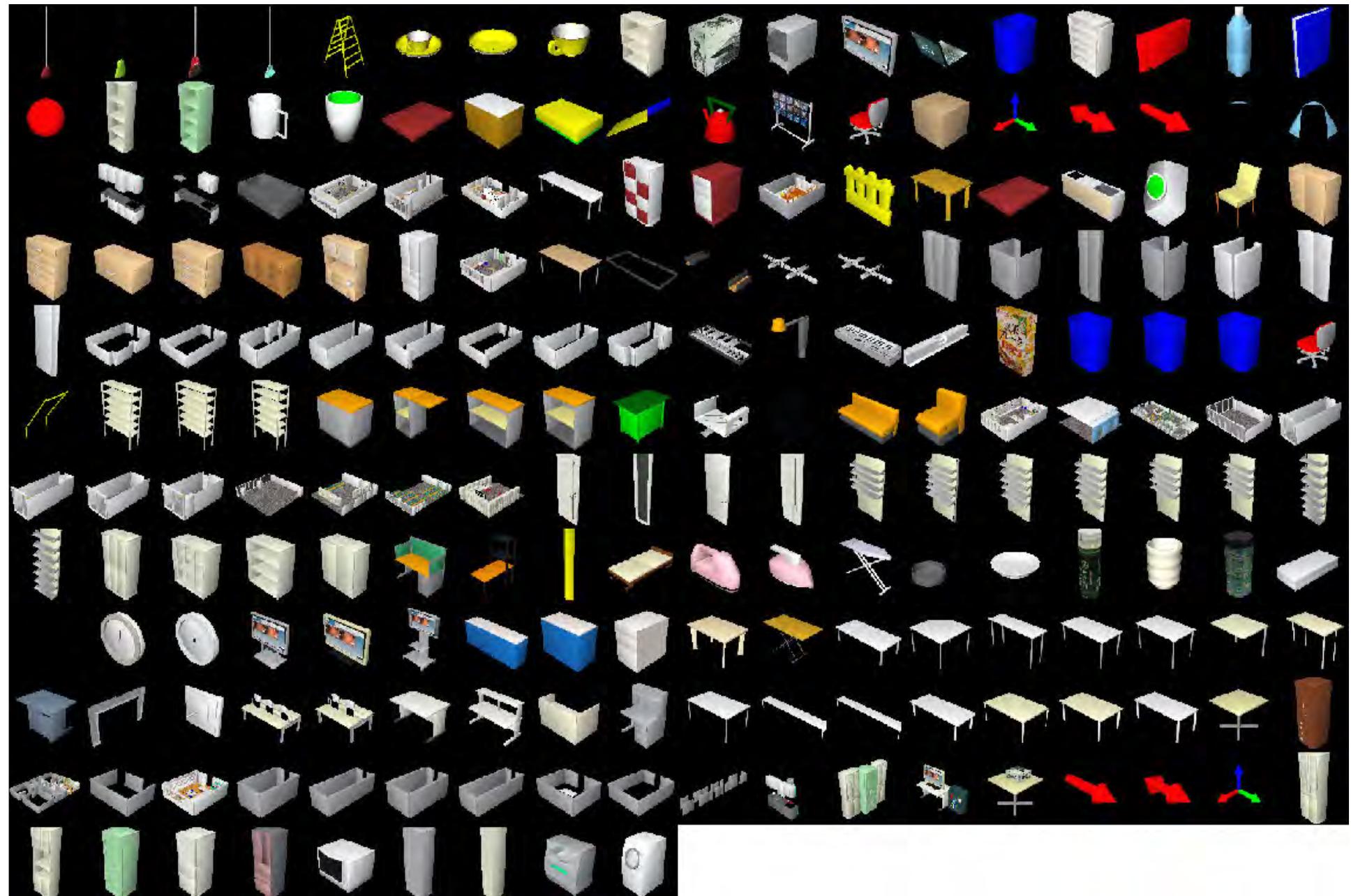
(setq *viewer* (view))
(setq cube1 (make-cube 50 50 50))
(setq cyl1 (make-cylinder 15 50))
(gldraw *viewer* cube1 cyl1)
(send cyl1 :rotate (deg2rad -50) :x)
(setq cubecyl1 (body+ cube1 cyl1))
(send cubecyl1 :rotate (deg2rad 60) :y)
(setq cubecyl2 (body- cube1 cyl1))
(send cubecyl2 :rotate (deg2rad -45) :z)
  
```

# ロボット行動実現研究システム環境 JSK

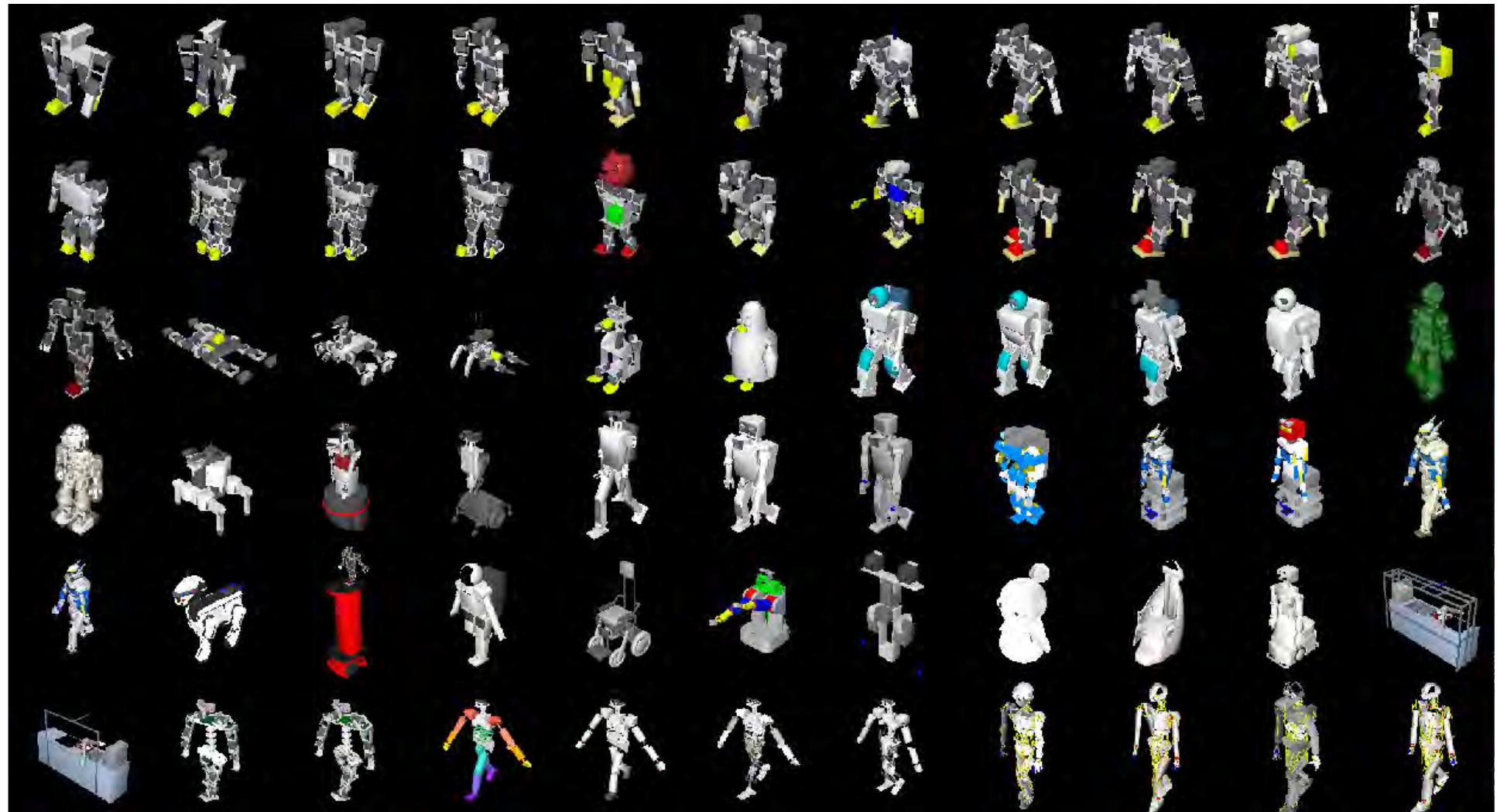


# Robots represented in our current Mother Environment

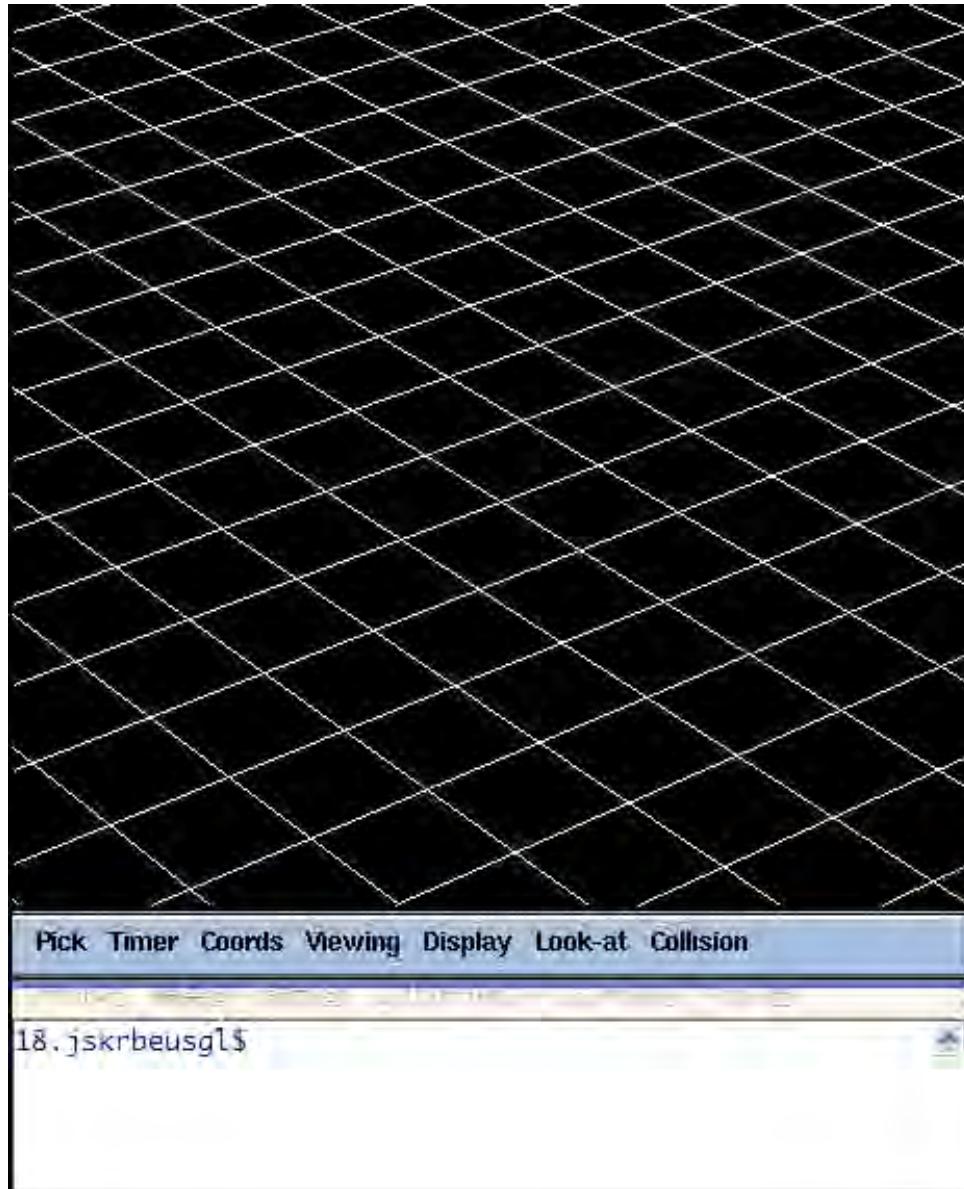




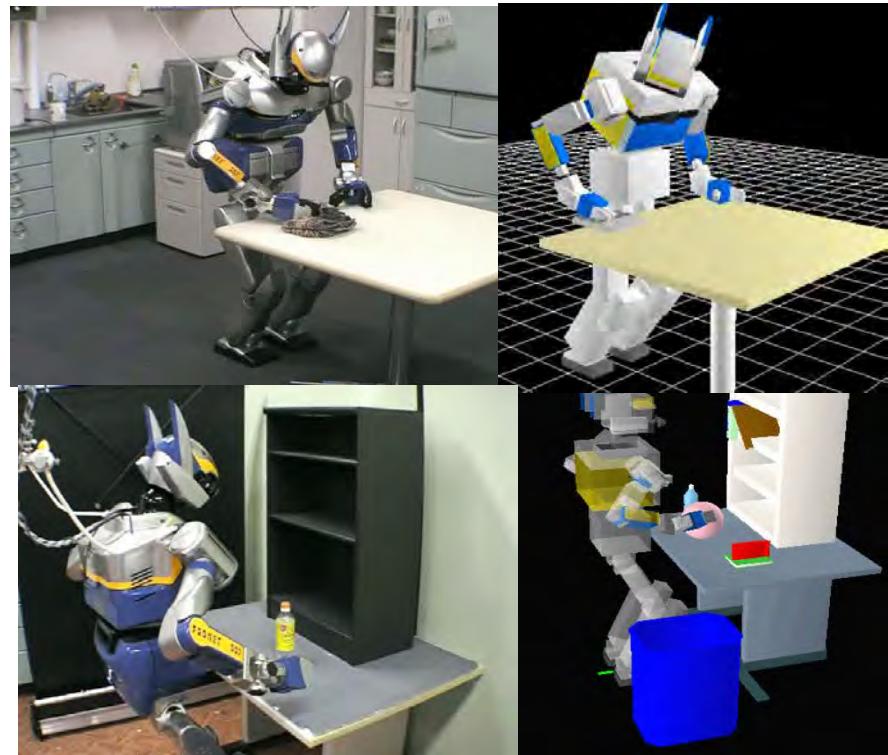
# ロボット身体モデル



# 動力学計算エンジン対応 in Euslisp

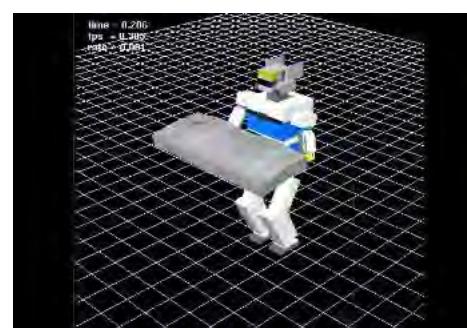
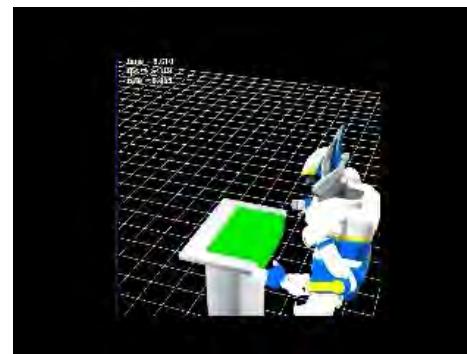
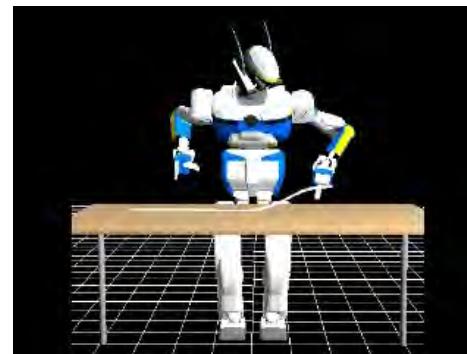
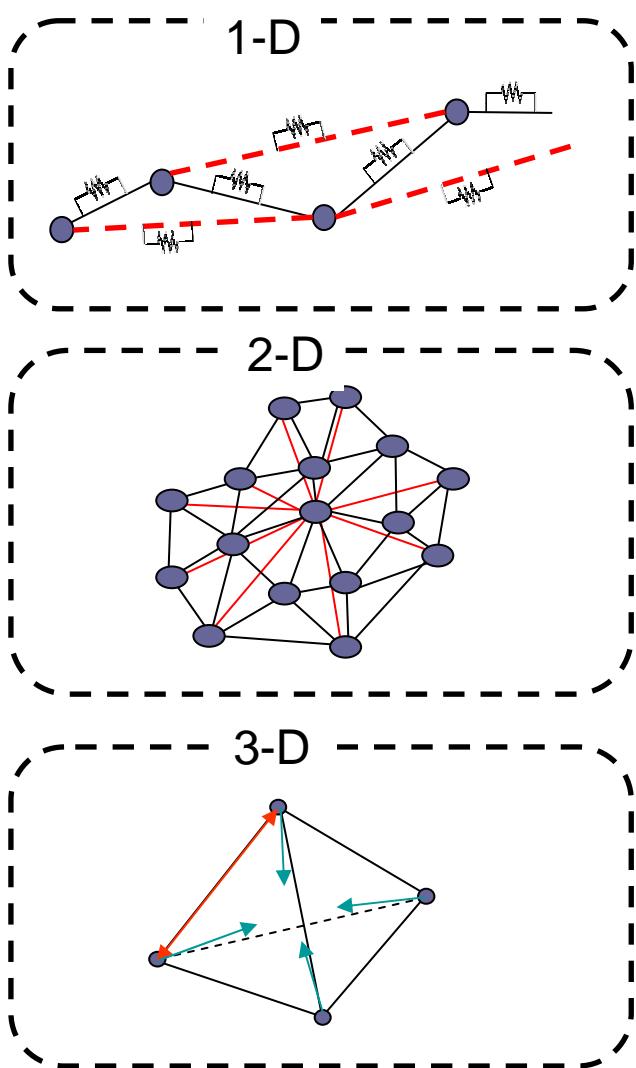


- Add an object
- Translate / rotate objects
- Toggle dynamics
- Control simulation loop



# 変形物体モデル in Euslisp

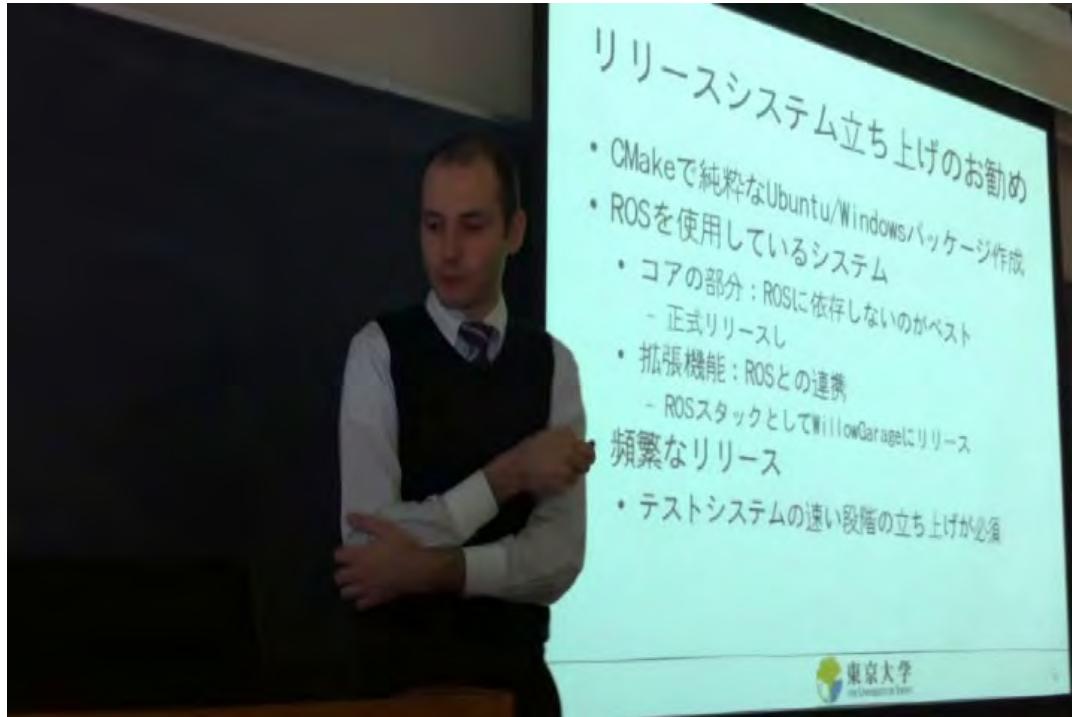
[IAS2008]



# 本講義 ロボットのソフトウェア RTM-ROS 相互運用

1. イントロ – 講義予定、ロボット知能化プロジェクト
  - 1) 講義の位置付け・背景 稲葉
  - 2) RTM チュートリアル 吉海
  - 3) ROS チュートリアル 岡田
2. ロボットソフトウェアの相互運用 吉海、岡田
3. RTミドルウェア: RTM 吉海、矢口
4. ソフトウェアツール: ドキュメント、テスト検証 魯仙
5. 動作計画、開発システム: OpenRAVE 魯仙(R. Diankov)
6. Robot オペレーティングシステム: ROS 岡田
7. RTM/ROS相互接続、ビジョンRTM 吉海
8. RTMビルドツール、三次元ビジョン: RTMExTender 矢口
9. 双腕マニピュレーション: HIRONX, Vpython 花井
10. インタラクティブシステム: OpenHRI, Choreonoid 原(産総研)
11. 実時間対応: OpenHRP, OpenRTM HRP4 金広(産総研)
12. ロボットソフトウェアの発展的構成法 稲葉

# Rtm-Ros-Robotics 講義



2011. 4. 20—7. 20 (水)

全12回、補習3回

13:00—14:30

本郷、秋葉原 TV会議接続



**講義ホームページ**

<http://code.google.com/p/rtm-ros-robotics/>

**ドキュメントWiki**

<http://code.google.com/p/rtm-ros-robotics/wiki/>

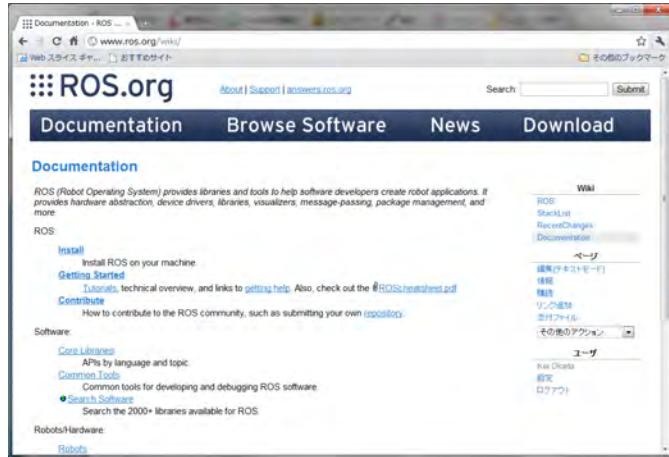
**ソースコードレポジトリ**

<http://rtm-ros-robotics.googlecode.com/svn/trunk>

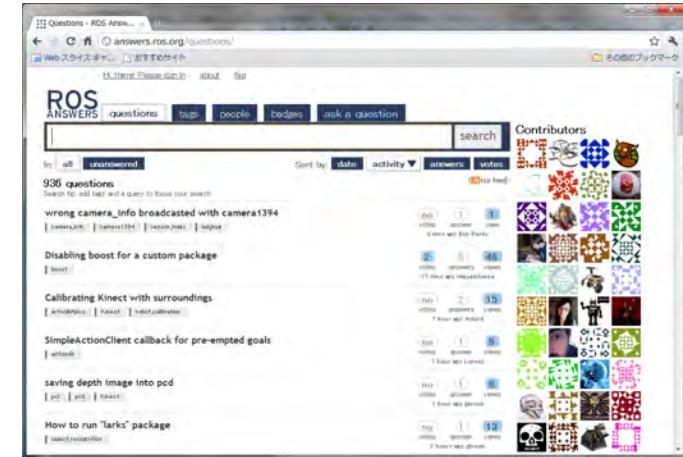
**uStream 撮影放映状況**

<http://ustre.am/xKBj>

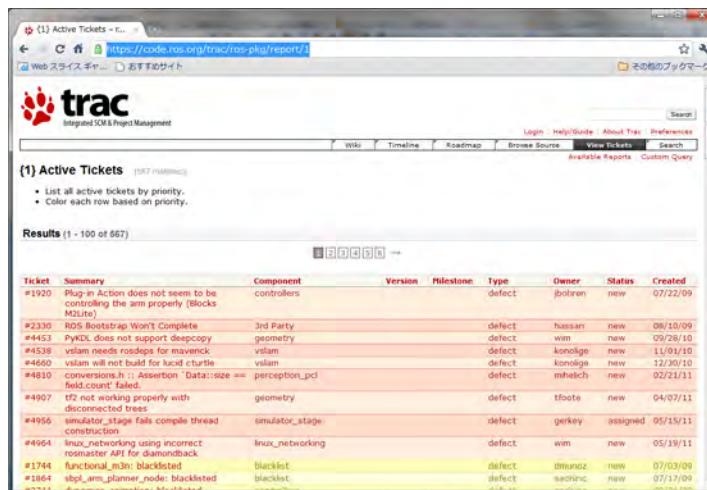
# コミュニティ型開発の例



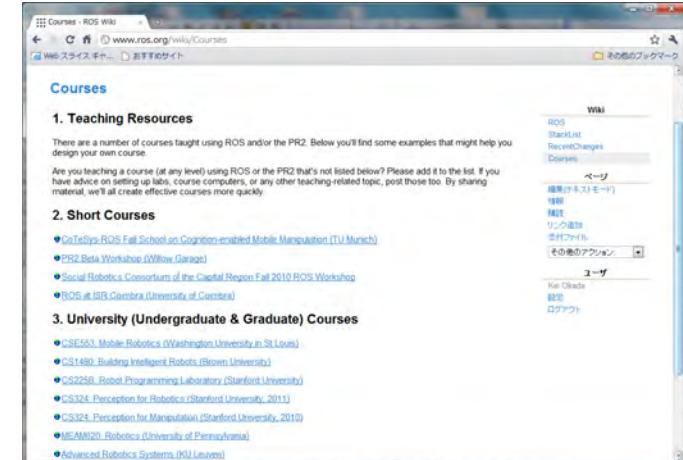
誰も編集可能なwiki  
<http://www.ros.org/wiki/>



質問共有サイト  
<http://answers.ros.org/>

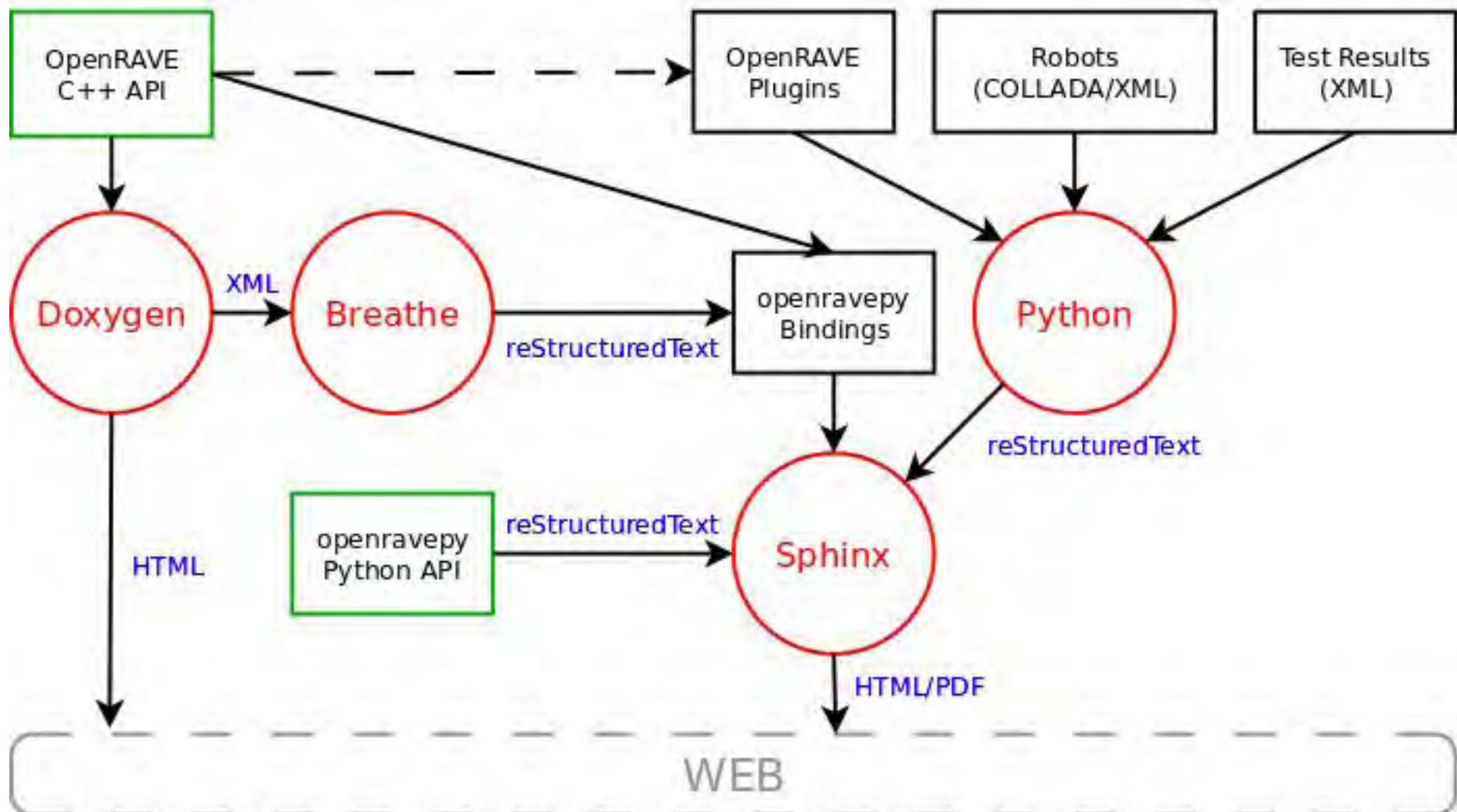


バグ報告, 機能提案  
<https://code.ros.org/trac/ros-pkg/report/1>

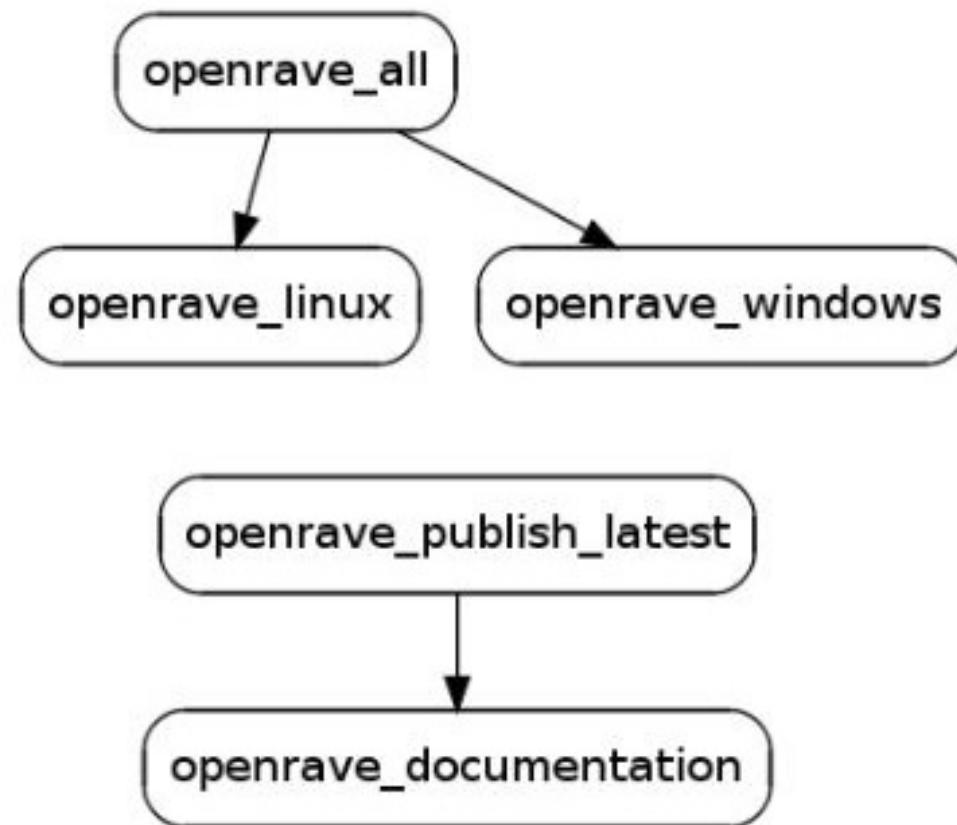


講義スライドも共有  
<http://www.ros.org/wiki/Courses>

# ドキュメント生成



# プログラムテスト Jenkins



OpenRAVE Jenkinsのジョブの例

# OpenRTM-OpenHRP-ROS 統合環境インストール

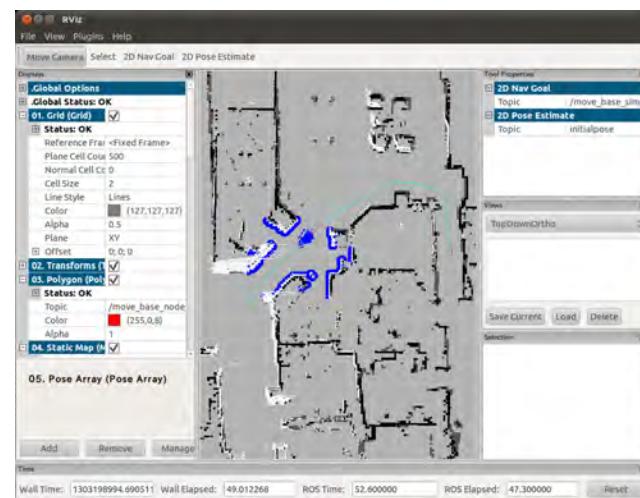
```
# ソースコードのインストール
$ rosinstall --continue-on-error
  ~/prog/rtm-ros-robotics /opt/ros/diamondback
  http://rtm-ros-robotics.googlecode.com/svn/tags/latest/agentsystem\_rostutorials/rtm-ros-robotics.rosinstall

# 環境のセットアップ
$ source ~/prog/rtm-ros-robotics/setup.sh

# RTM-ROS 統合環境のコンパイル
$ rosmake --pre-clean --rosdep-install --rosdep-yes
  hrpsys_ros_bridge
```

# ナビゲーション ROS

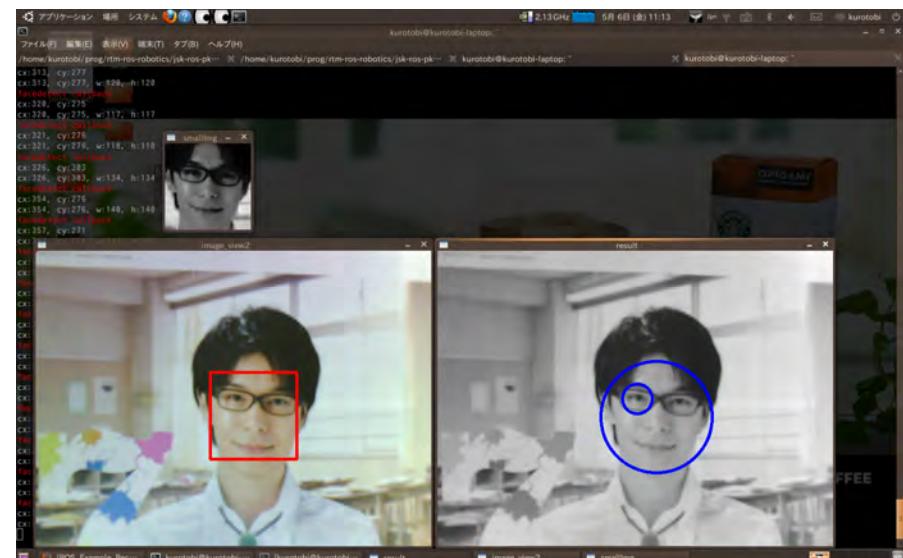
- rosrun  
move\_base\_stage\_tutorial  
robot.launch
- Roscd  
move\_base\_stage\_tutorial
- rosrun rviz rviz -d  
config/rviz.vcg
- rosrun pr2\_teleop  
teleop\_pr2\_keyboard



Rviz画面

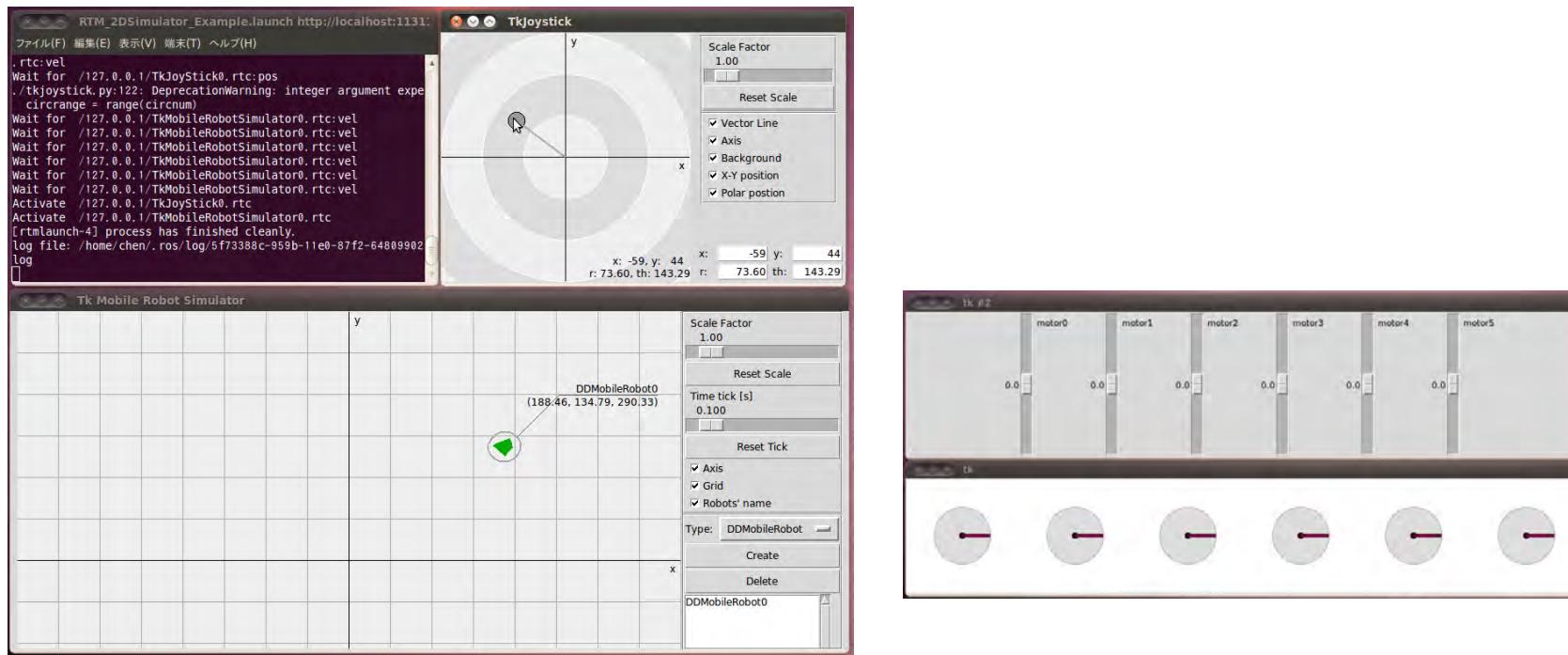
# 認識処理

- **roslaunch roseus\_tutorials usb-camera.launch**
- **roslaunch roseus\_tutorials image-view.launch**
- **roslaunch roseus\_tutorials face-detector-mono.launch**
- **rosrun roseus\_tutorials face-detector-mono.l**
  - (1)
  - **rosrun roseus\_tutorials vision-action-example1.l**
  - (2)
  - **rosrun roseus\_tutorials vision-action-example2.l**
  - (3)
- **roslaunch roseus\_tutorials checkerboard-pose.launch**
- **rosrun roseus\_tutorials vision-action-example3.l**



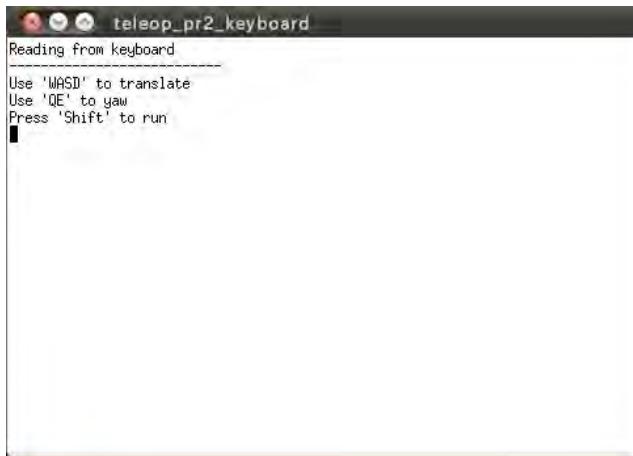
# 2D RTM シミュレータ

- roslaunch openrtm MobileRobot\_2DSimulator\_Example.launch
- roslaunch openrtm Slider\_and\_Motor\_Example.launch
- roslaunch openrtm Seq\_Example.launch



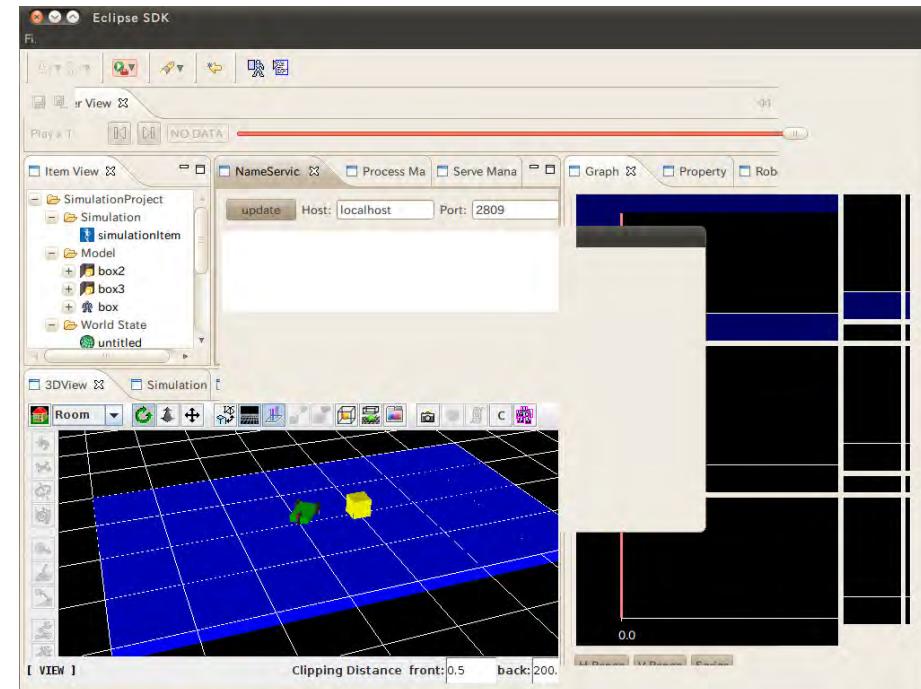
# 移動ロボット mrobot

- **roslaunch mrobot\_ros\_bridge mrobot\_simulator.launch**



```
teleop_pr2_keyboard
Reading from keyboard
Use 'WASD' to translate
Use 'QE' to yaw
Press 'Shift' to run
```

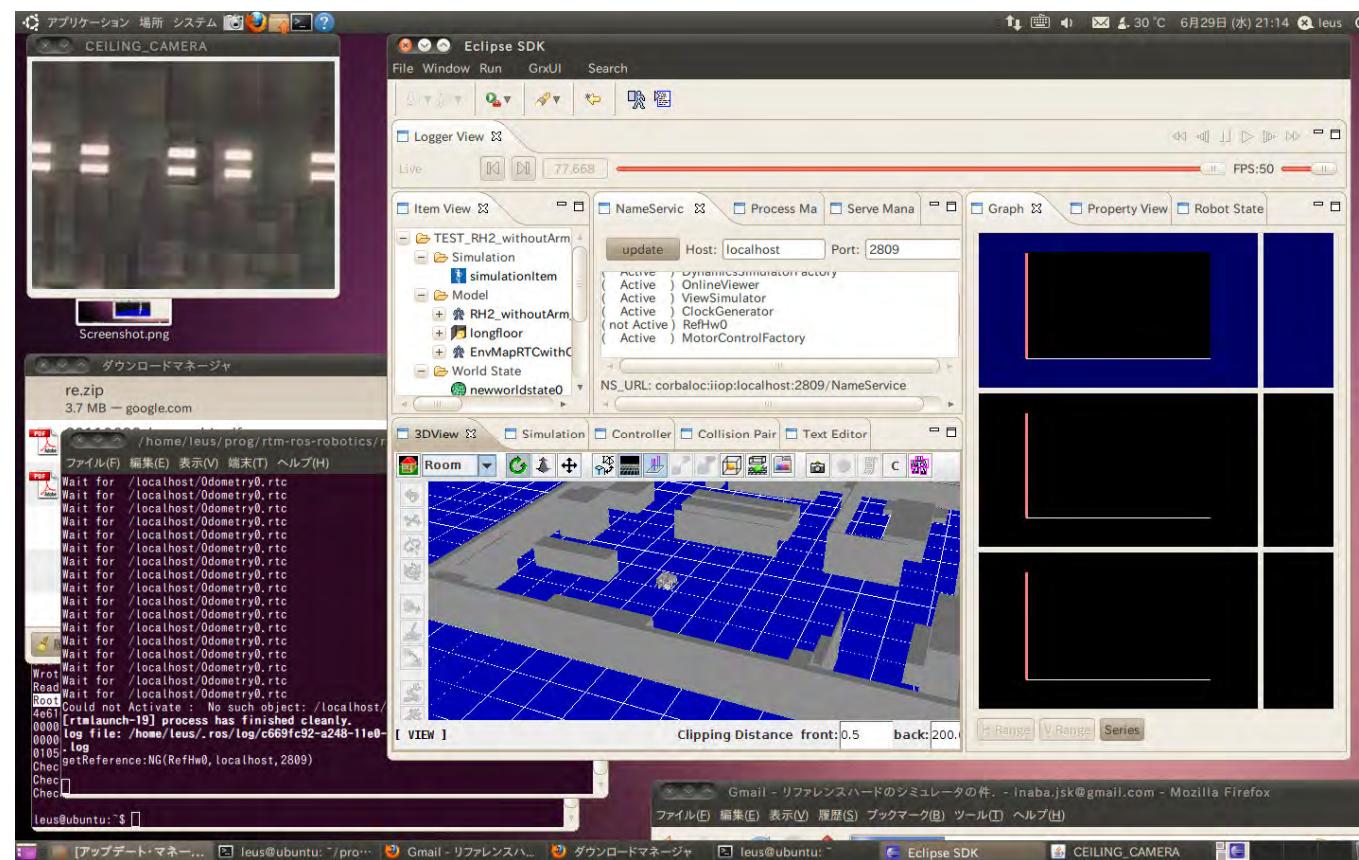
キーボード操作



OpenHRP

# 移動マニピュレータ RH

- roslaunch RS003 rh.launch



# 双腕ヒューマノイド HIRONX

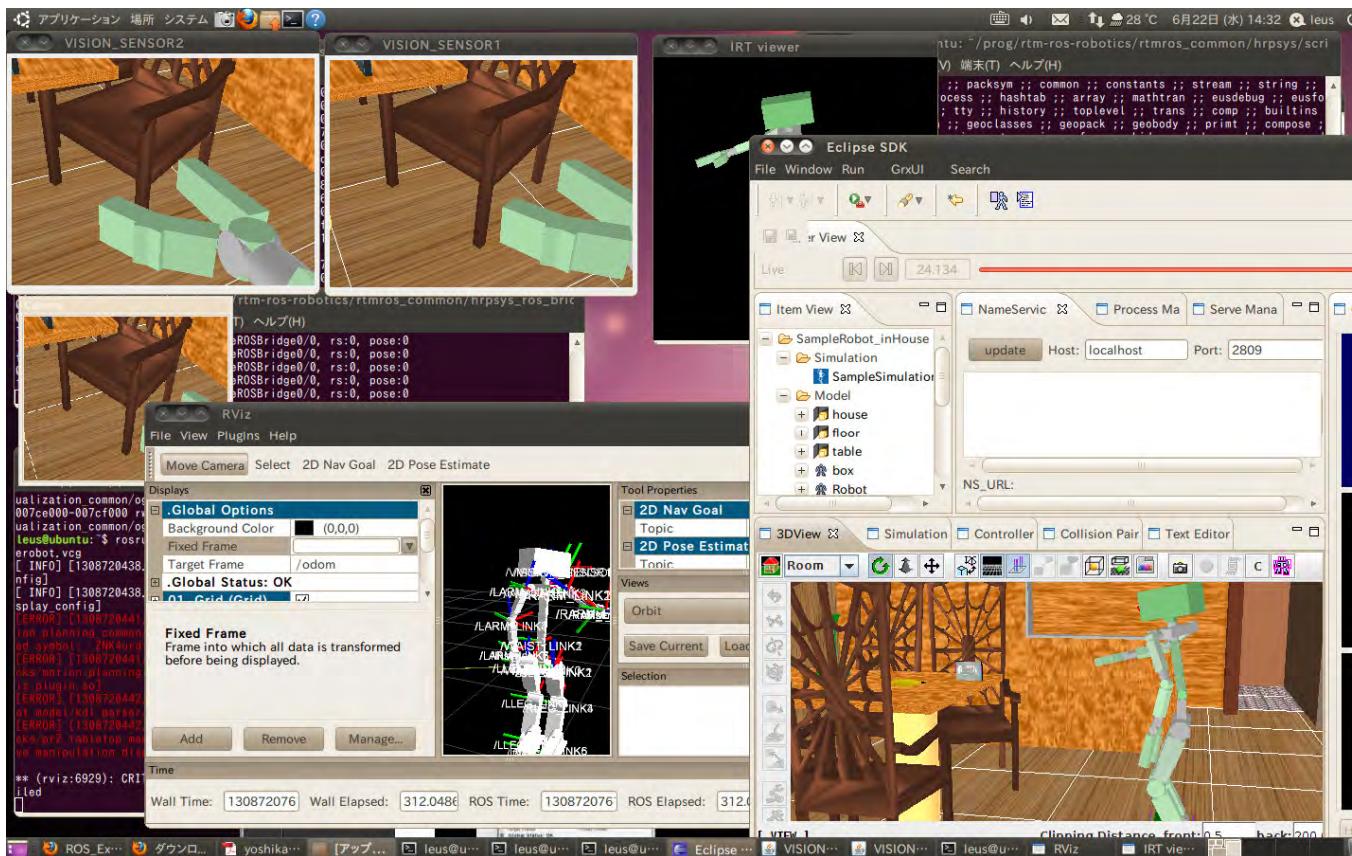
- **\$ roscd iv\_plan/examples**  
**\$ ipython pickbox.py**  
**putbox()**
- **Pick()**
- **Place()**



Vpythonでの表示

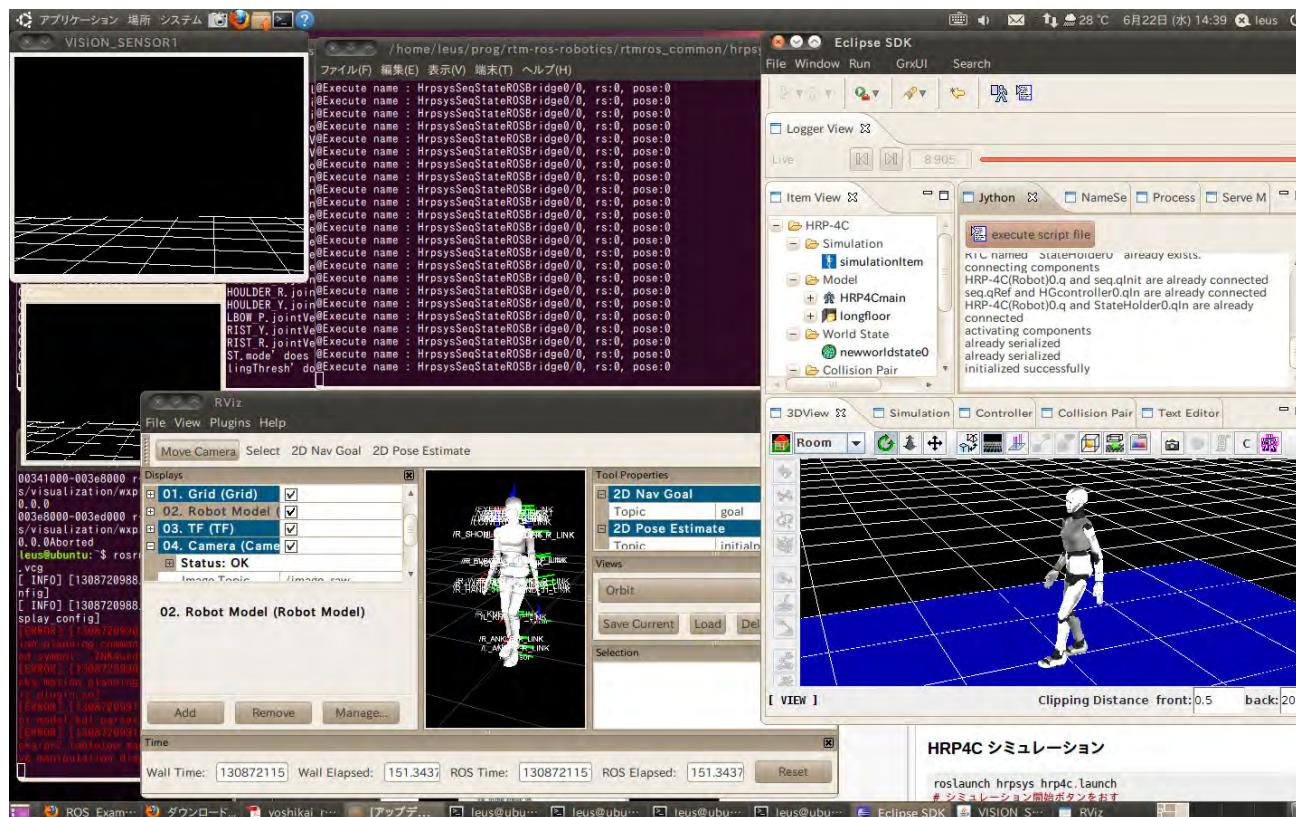
# サンプルヒューマノイド

- `roslaunch hrpsys samplerobot.launch`
- `roslaunch hrpsys_ros_bridge samplerobot_ros_bridge.launch`
- `rosrun rviz rviz -d `rospack find hrpsys_ros_bridge`/launch/samplerobot.vcg`



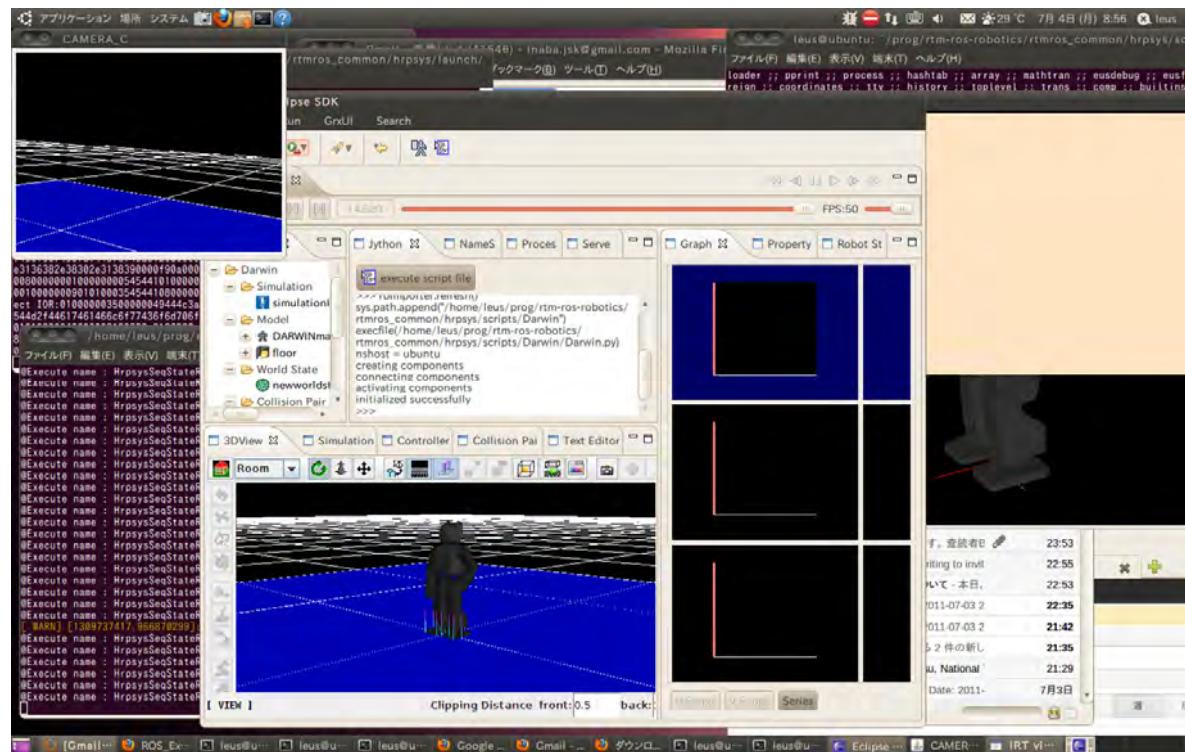
# HRP4C

- `roslaunch hrpsys hrp4c.launch`



# 小型ヒューマノイド DARWIN

- rosrun hrpsys darwin.launch
- rosrun hrpsys\_ros\_bridge darwin\_ros\_bridge.launch
- roscd hrpsys/scripts/Darwin
- rosequa sample.l



# 統合環境実行ツール

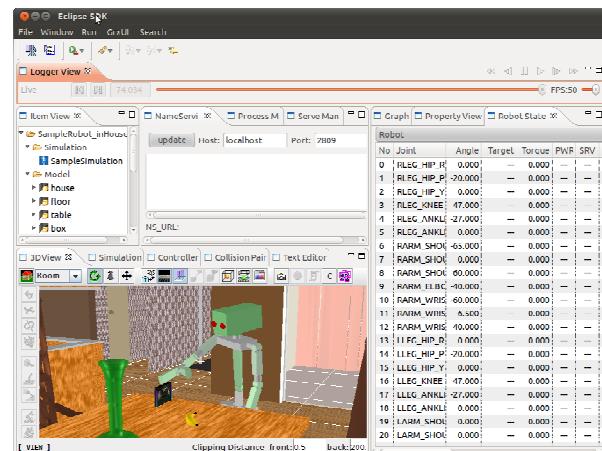
# RTM環境の立ち上げ

```
$ rosrun hrpsys samplerobot.launch
```

# ROS環境の立ち上げ

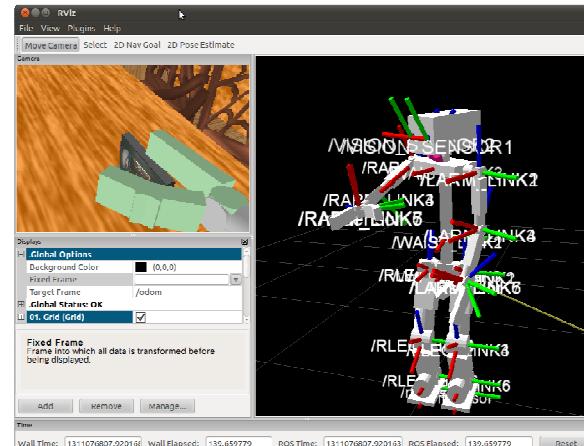
```
$ rosrun hrpsys_ros_brikdge samplerobot_ros_brikdge.launch
```

それぞれ9個のプロセスを立ち上げ, RTCの接続, 開始, アクティベイトを行う.



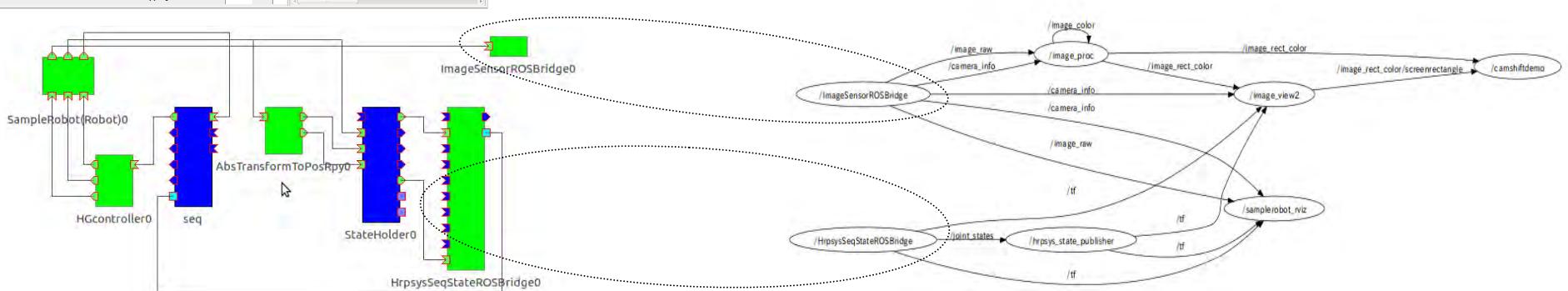
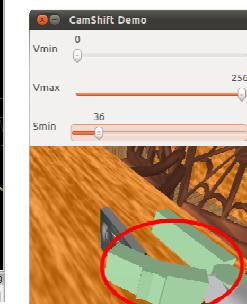
## RTM環境

- ・物理シミュレーション  
(干渉計算,  
モデルローダ)
- ・ロボットモデル
- ・コントローラ
- ・シーケンサ
- ・センサーholder

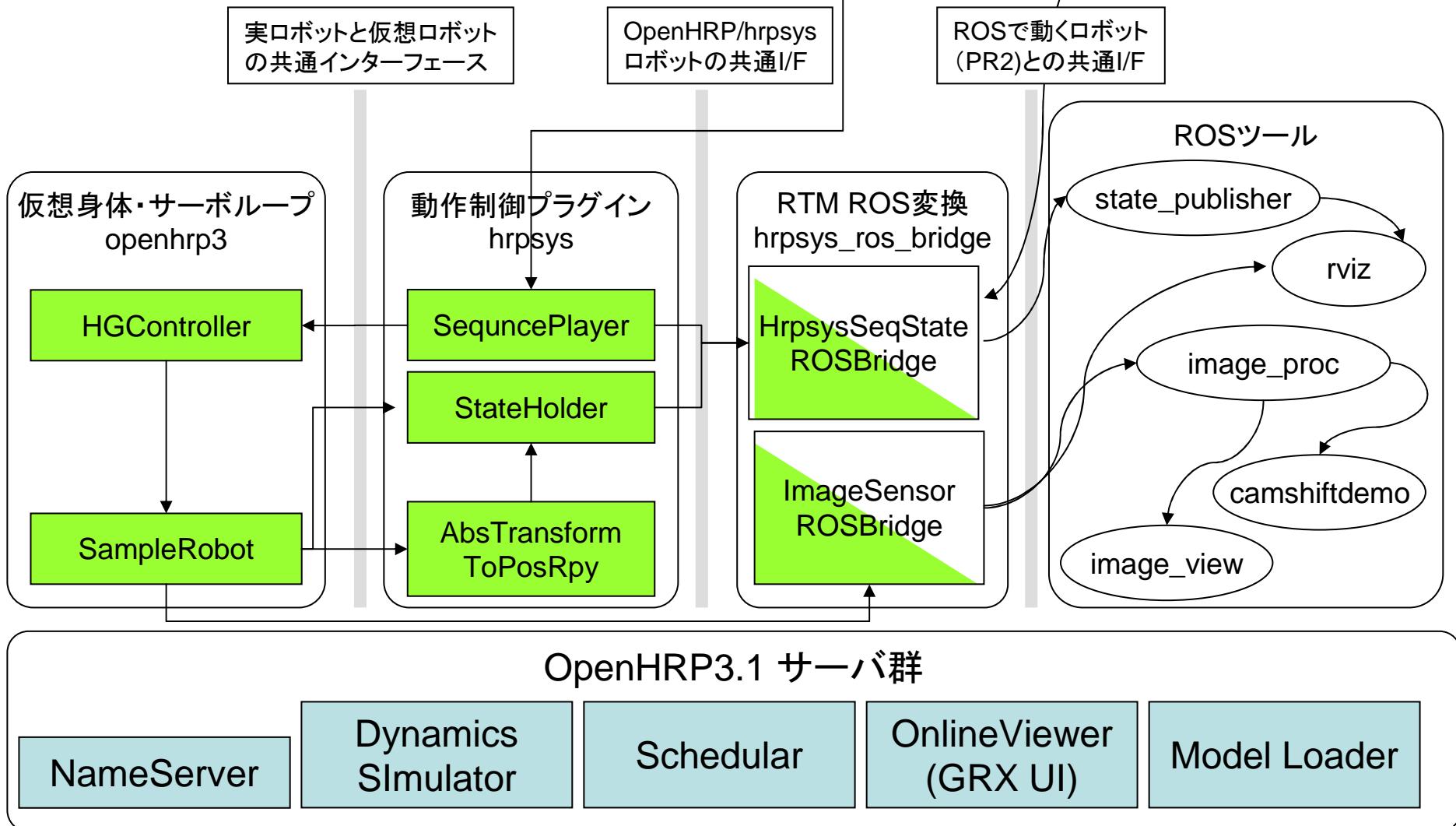


## ROS環境

- ・センサービューア
- ・画像処理

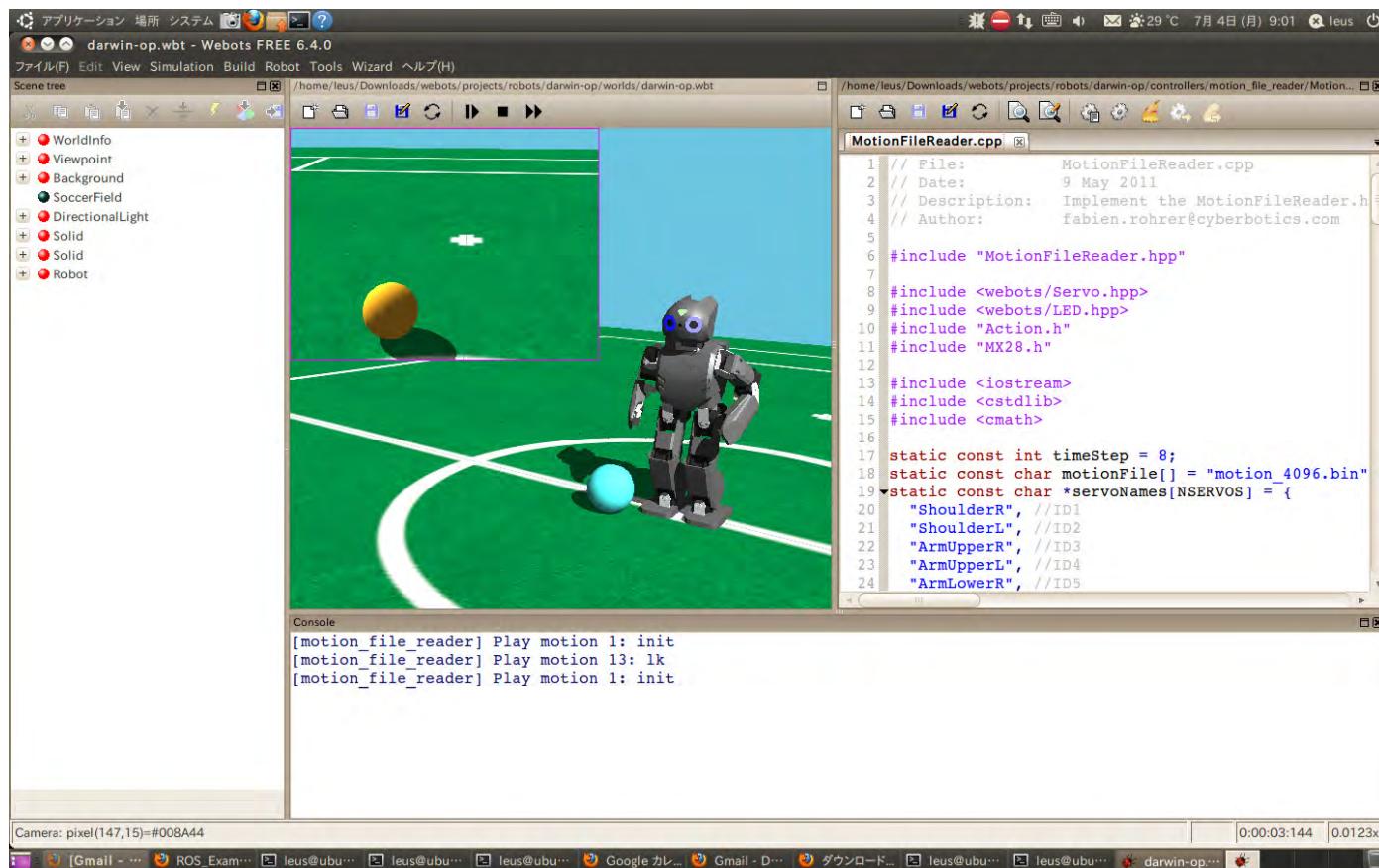


# OpenRTM-OpenHRP ROS 統合環境まとめ



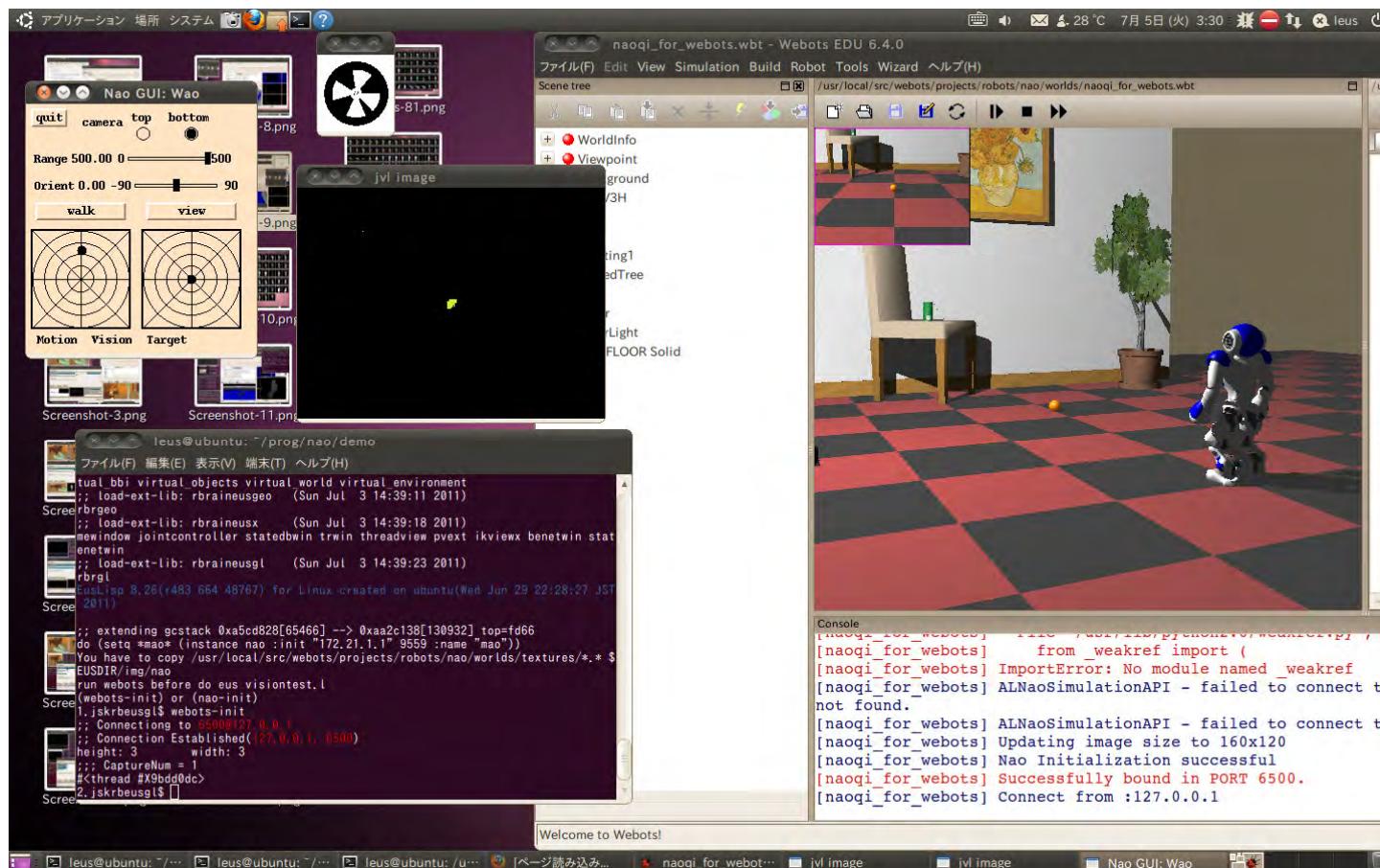
# シミュレータ

- <http://www.cyberbotics.com/download>



# シミュレータ

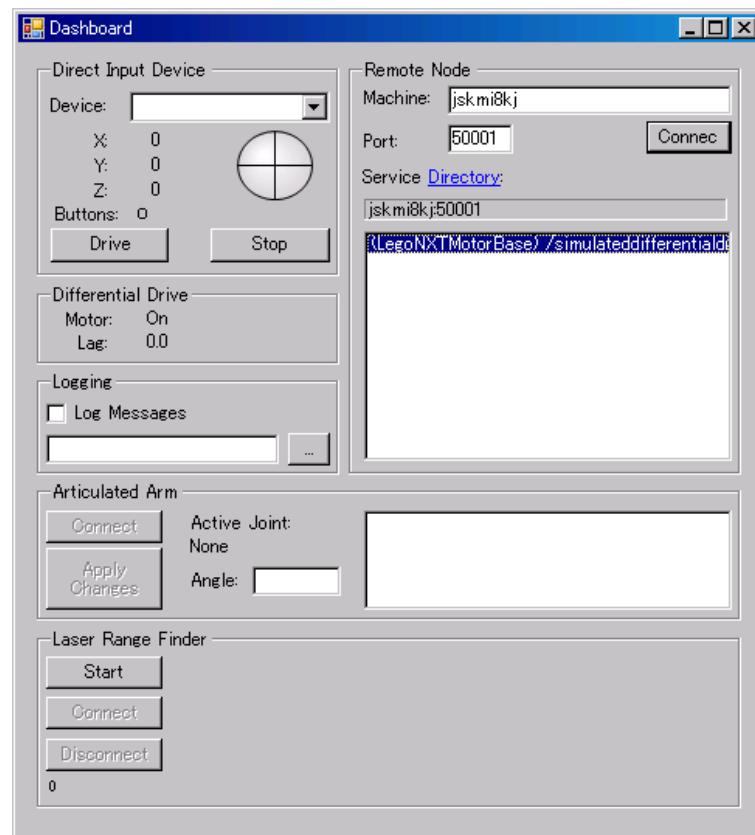
- <http://www.cyberbotics.com/download>



## NAOのWebots対応環境

# Microsoft Robotics Studio

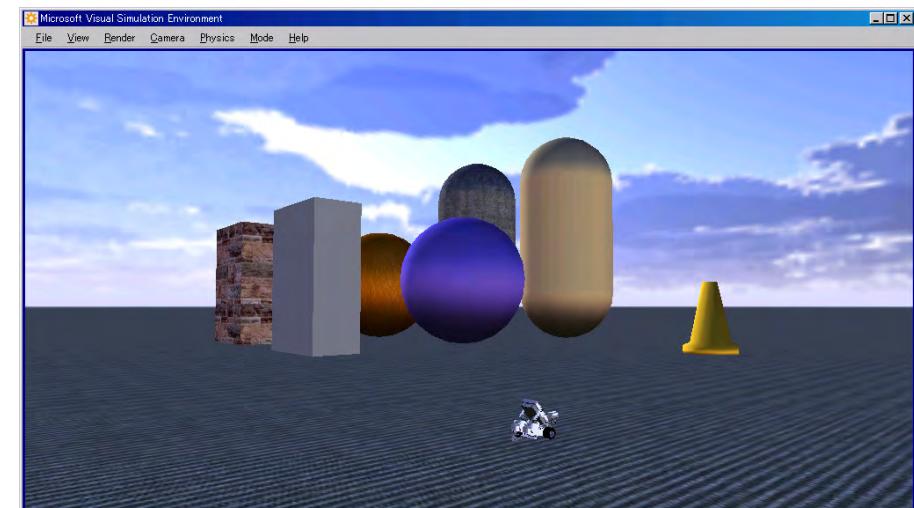
- <http://msdn.microsoft.com/robotics/>



```

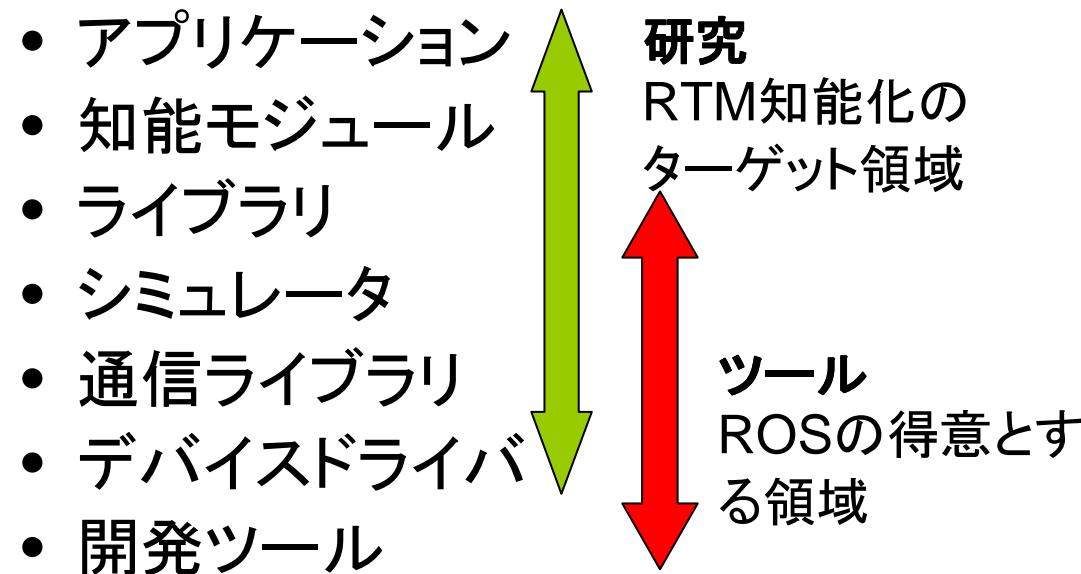
LeGO NXT Tribot Simulation
P:\20\April\%202007%\samples\config\LEGO.NXT.Tribot.Simulation.manifest.xml [04/25/2007 14:22:24][http://jskmi8kj:50000/manifest!loaderclient]
* Manifest load complete [04/25/2007 14:22:29][http://jskmi8kj:50000/manifest!loaderclient]
* Starting loading of BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\street_cone.bos
* Finished loading BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\street_cone.bos
* Starting loading of BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\LegоНXTtribot.bos
* Finished loading BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\LegоНXTtribot.bos
* Starting loading of BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\OmniLight.bos
* Finished loading BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\OmniLight.bos
* Starting loading of BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\Arrow.bos
* Finished loading BOS file:C:\Microsoft Robotics Studio 1.5 (CTP April 2007)\store\media\Arrow.bos
* Service uri: [04/25/2007 14:22:36][http://jskmi8kj:50000/simulateddifferentia
ldrive/b3cb9e9-9b7f-4704-b856-bf59a49df66d]
* Service uri: [04/25/2007 14:22:37][http://jskmi8kj:50000/simulatedbumper/b8
fa99ef-80d4-422e-ae63-889c0e7f6f7d]

```



# RTM-ROS統合方式

- ROSのオープンソース資産の上にRTMによる知能ロボット環境を構築



ツールと研究の比率の違い

Steve Cousins speaking at  
Robo Development Tuesday  
より

# 書類配達支援 6階から1階へ



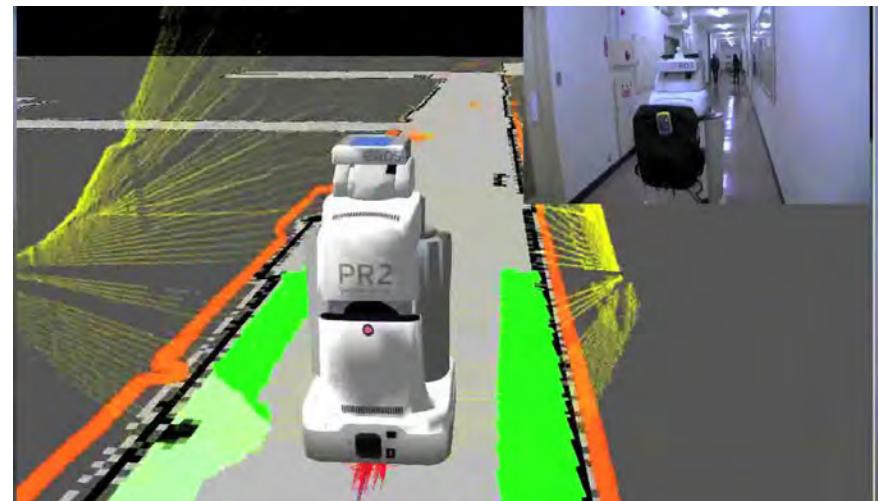
6階で書類を渡す



エレベータボタンを認識



何階に居るのかを認識する

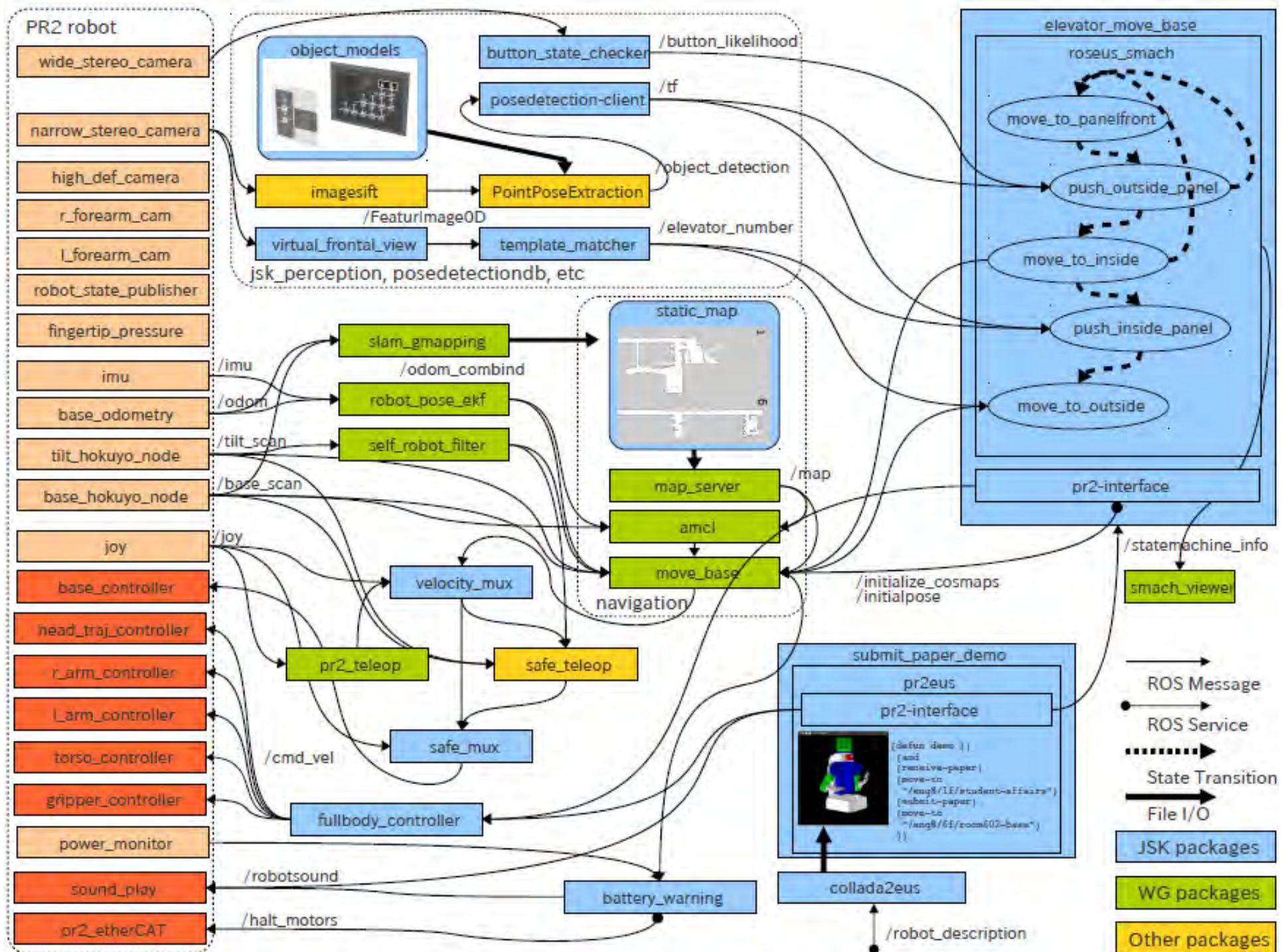


ロボットの自律移動時の環境認識

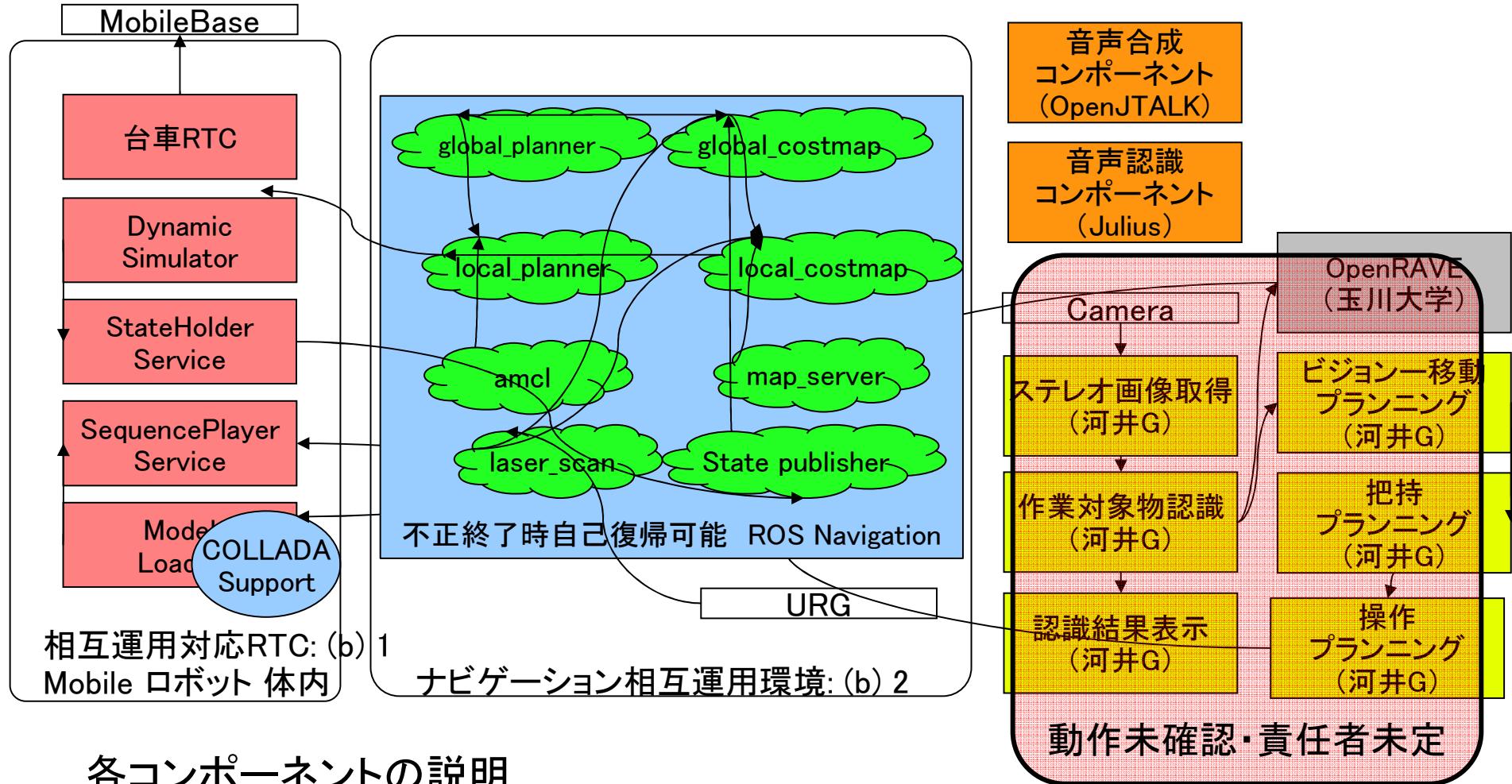
# 家具操作の教示



ロボットはどこにいるかを知っているためその場での家具操作教示で再利用可能



# 相互運用計画



## 各コンポーネントの説明



# 本講義 ロボットのソフトウェア RTM-ROS 相互運用

1. イントロ – 講義予定、ロボット知能化プロジェクト
  - 1) 講義の位置付け・背景 稲葉
  - 2) RTM チュートリアル 吉海
  - 3) ROS チュートリアル 岡田
2. ロボットソフトウェアの相互運用 吉海、岡田
3. RTミドルウェア: RTM 吉海、矢口
4. ソフトウェアツール: ドキュメント、テスト検証 魯仙
5. 動作計画、開発システム: OpenRAVE 魯仙(R. Diankov)
6. Robot オペレーティングシステム: ROS 岡田
7. RTM/ROS相互接続、ビジョンRTM 吉海
8. RTMビルドツール、三次元ビジョン: RTMExTender 矢口
9. 双腕マニピュレーション: HIRONX, Vpython 花井
10. インタラクティブシステム: OpenHRI, Choreonoid 原(産総研)
11. 実時間対応: OpenHRP, OpenRTM HRP4 金広(産総研)
12. ロボットソフトウェアの発展的構成法 稲葉

東京大学大学院 情報理工学系研究科  
平成23年度 夏学期 大学院講義

創造情報学特論I (4860-1014 創造情報学専攻)  
無かったものを生み出すクリエイティブなステップ  
のための構成法とは

エージェントシステム(4850-1010 知能機械情報学専攻)  
感覚と行動、入力と出力をどう連結してゆき  
実環境、仮想環境での行動はどう作られるべきか

IRT通論(4891-1001 博士後期課程IRTプログラム)  
目的を与える社会があり、それを産業として支える  
技術を構築する方法論としてはどう評価できるか

東京大学大学院 情報理工学系研究科  
平成23年度 夏学期  
最終回

# ソフトウェアの発展的構成法について

東京大学大学院  
情報理工学系研究科 創造情報学専攻  
稻葉雅幸

Department of Creative Informatics  
<http://www.jsk.t.u-tokyo.ac.jp/>