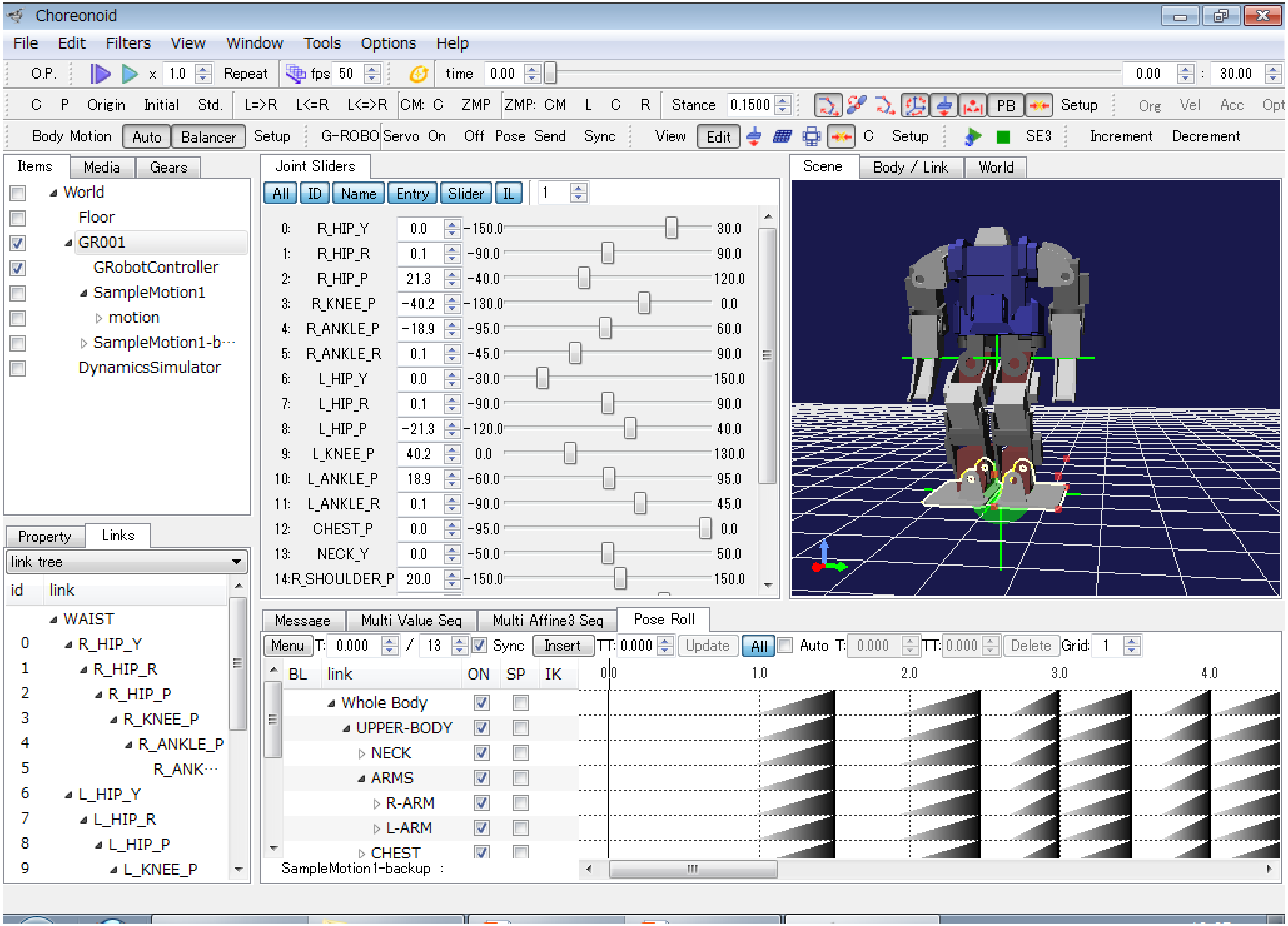
Choreonoid as a Software Framework for Implementing Graphical Robotics Applications

Shin’ichiro Nakaoka

AIST

Demonstration(1)



x 1.0 · fps 50 · time 0.00 · 0.00 : 30.00 · Stance 0.1500

- □ World
  - □ Floor
  - ☑ GR001
    - ☑ GRobotController
    - □ SampleMotion1
      - □ motion
    - □ SampleMotion1-b···
  - □ DynamicsSimulator

0:	R_HIP_Y	0.0	-150.0	30.0
1:	R_HIP_R	0.1	-90.0	90.0
2:	R_HIP_P	21.3	-40.0	120.0
3:	R_KNEE_P	-40.2	-130.0	0.0
4:	R_ANKLE_P	-18.9	-95.0	60.0
5:	R_ANKLE_R	0.1	-45.0	90.0
6:	L_HIP_Y	0.0	-30.0	150.0
7:	L_HIP_R	0.1	-90.0	90.0
8:	L_HIP_P	-21.3	-120.0	40.0
9:	L_KNEE_P	40.2	0.0	130.0
10:	L_ANKLE_P	18.9	-60.0	95.0
11:	L_ANKLE_R	0.1	-90.0	45.0
12:	CHEST_P	0.0	-95.0	0.0
13:	NECK_Y	0.0	-50.0	50.0
14:	R_SHOULDER_P	20.0	-150.0	150.0

link tree

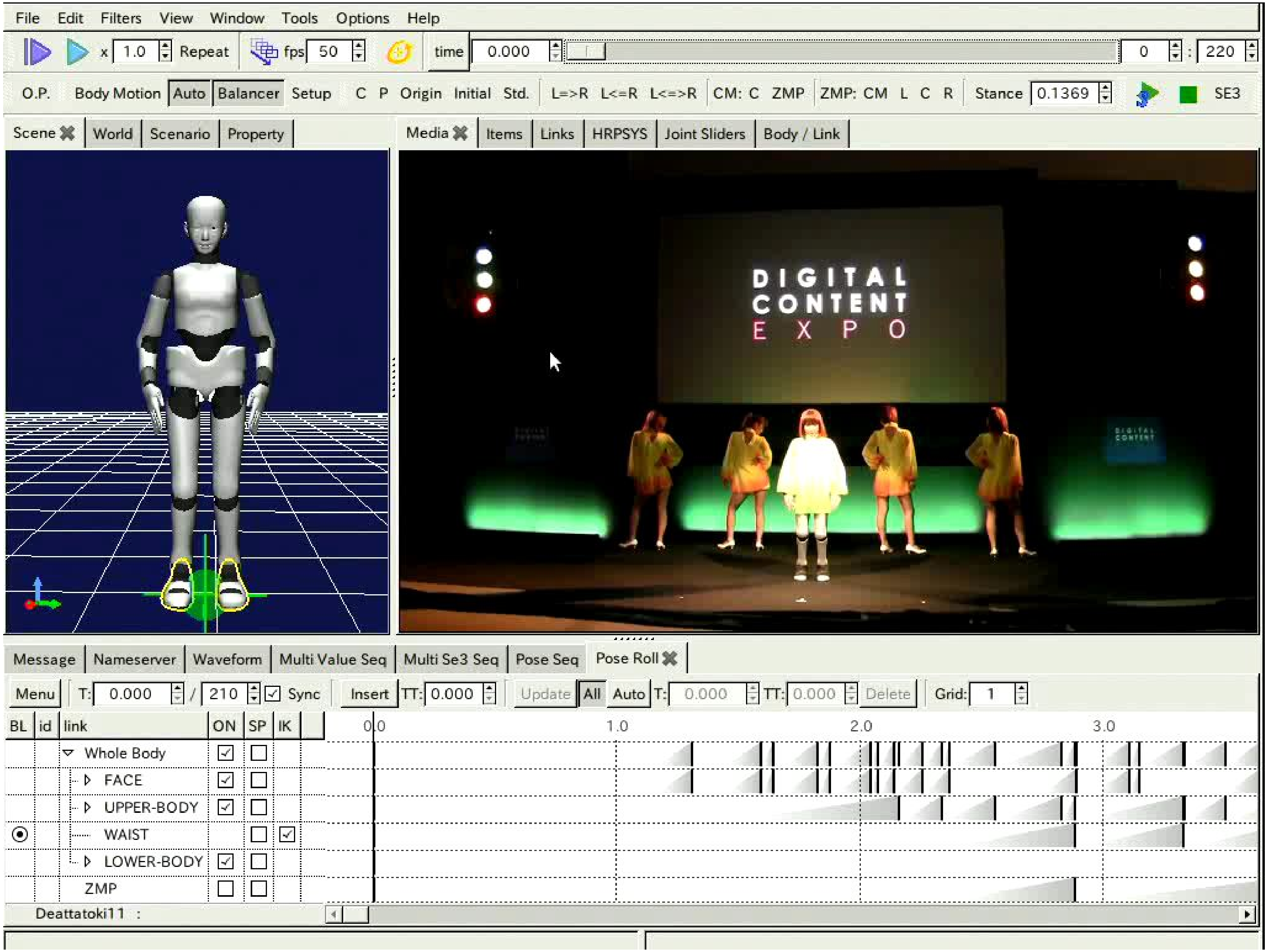
id	link
	WAIST
0	R_HIP_Y
1	R_HIP_R
2	R_HIP_P
3	R_KNEE_P
4	R_ANKLE_P
5	R_ANK···
6	L_HIP_Y
7	L_HIP_R
8	L_HIP_P
9	L_KNEE_P

T: 0.000 / 13 ☑ Sync TT: 0.000 □ Auto T: 0.000 TT: 0.000 Grid: 1

BL	link	ON	SP	IK
	Whole Body	☑	□	
	UPPER-BODY	☑	□	
	NECK	☑	□	
	ARMS	☑	□	
	R-ARM	☑	□	
	L-ARM	☑	□	
	CHEST	☑	□	

SampleMotion1-backup :

Demonstration(2)



x 1.0

fps 50

time 0.000

0 : 220

Stance 0.1369

T: 0.000 / 210 ☑ Sync

TT: 0.000

T: 0.000 TT: 0.000

Grid: 1

BL	id	link	ON	SP	IK
		▽ Whole Body	☑	□	
		▷ FACE	☑	□	
		▷ UPPER-BODY	☑	□	
◉		WAIST		□	☑
		▷ LOWER-BODY	☑	□	
		ZMP	□	□	

Deattatoki11 :

Choreonoid Framework Overview

A framework which covers GUIs and visualizations of robotics applications

High-performance monolithic (single process) structure all written in C++

Additional functions and their cooperation with existing functions can be flexibly implemented as plugins

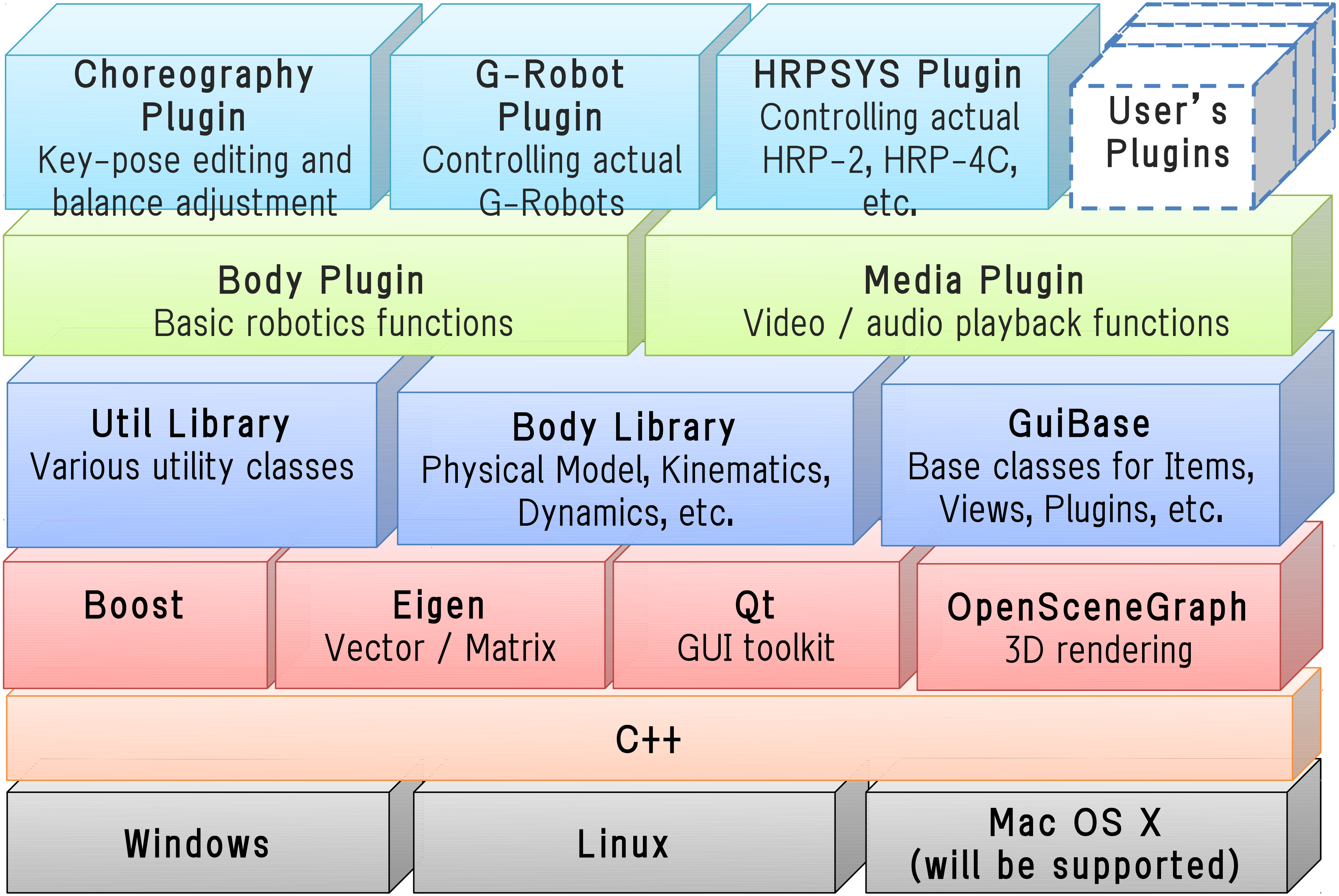
Basic Design of the framework

Items and item tree structure

Views and toolbars with flexible layout system

MVC-like architecture with signal-slot mechanism

Components



Choreography Plugin
Key-pose editing and balance adjustment

G-Robot Plugin
Controlling actual G-Robots

HRPSYS Plugin
Controlling actual HRP-2, HRP-4C, etc.

User's Plugins

Body Plugin
Basic robotics functions

Media Plugin
Video / audio playback functions

Util Library
Various utility classes

Body Library
Physical Model, Kinematics, Dynamics, etc.

GuiBase
Base classes for Items, Views, Plugins, etc.

Boost

Eigen
Vector / Matrix

Qt
GUI toolkit

OpenSceneGraph
3D rendering

C++

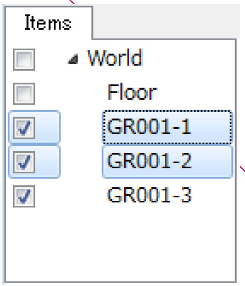
Windows

Linux

Mac OS X
(will be supported)

```
class SamplePlugin : public Plugin {
public:
    SamplePlugin() : Plugin(”Sample”) { depend(”Body”); }
    virtual bool initialize() {
        ToolBar* bar = new ToolBar(”Sample1”);
             ->addButton(”Increment“)
            ->sigClicked().connect(bind(&onButtonClicked, +0.04));
             ->addButton(”Decrement“)
            ->sigClicked().connect(bind(&onButtonClicked, -0.04));
          addToolBar(bar);
        return true;
    }
};
void onButtonClicked(double dq){
    ItemList<BodyItem> bodyItems =
        ItemTreeView::mainInstance()->selectedItems<BodyItem>();
    for(size_t i=0; i < bodyItems.size(); ++i){
        BodyPtr body = bodyItems[i]->body();
        for(int j=0; j < body->numJoints(); ++j){
            body->joint(j)->q += dq;
        }
        bodyItems[i]->notifyKinematicStateChange(true);
    }
}
```

clicked


Items

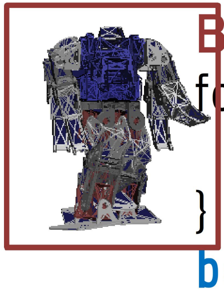
☐ World

☐ Floor

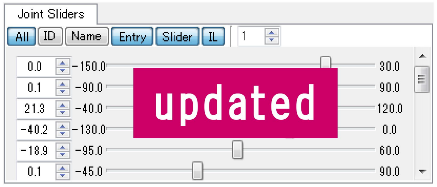
☑ GR001-1

☑ GR001-2

☑ GR001-3



signal



1

0.0 -150.0 30.0

0.1 -90.0 90.0

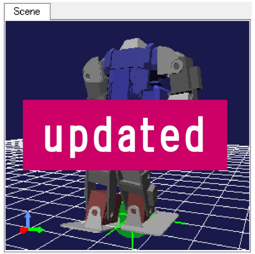
21.3 -40.0 120.0

-40.2 -130.0 0.0

-18.9 -95.0 60.0

0.1 -45.0 90.0

signal



Possible Applications

Simulators

Motion planning tools

Scenario scripting tools

Model design tools

Robot operation tools

Future Plan

We will soon open the official website www.choreonoid.org

We will release source / binary packages with an open-source license