A New Architecture for Deriving Dynamic Brain-Machine Interfaces

José Fortes, Renato Figueiredo, José Principe
Dep. of Electrical and Computer Eng.

Linda Hermer-Vazquez
Dep. of Psychology

Justin Sanchez
Dep. of Pediatrics

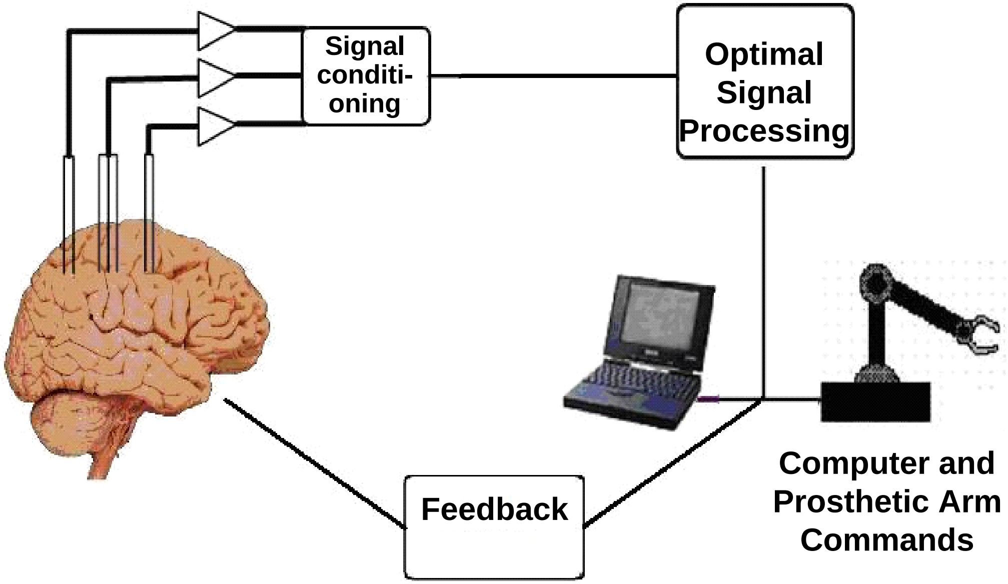
Outline

- What is BMI
- DDDBMI architecture vs. classical BMI
  - Switching among inverse-forward model pairs
- Movemes as motion primitives/models
- Distributed experimental setup
- Computational support
  - Algorithm structure
  - Reservation and QoS middleware
- Closing remarks

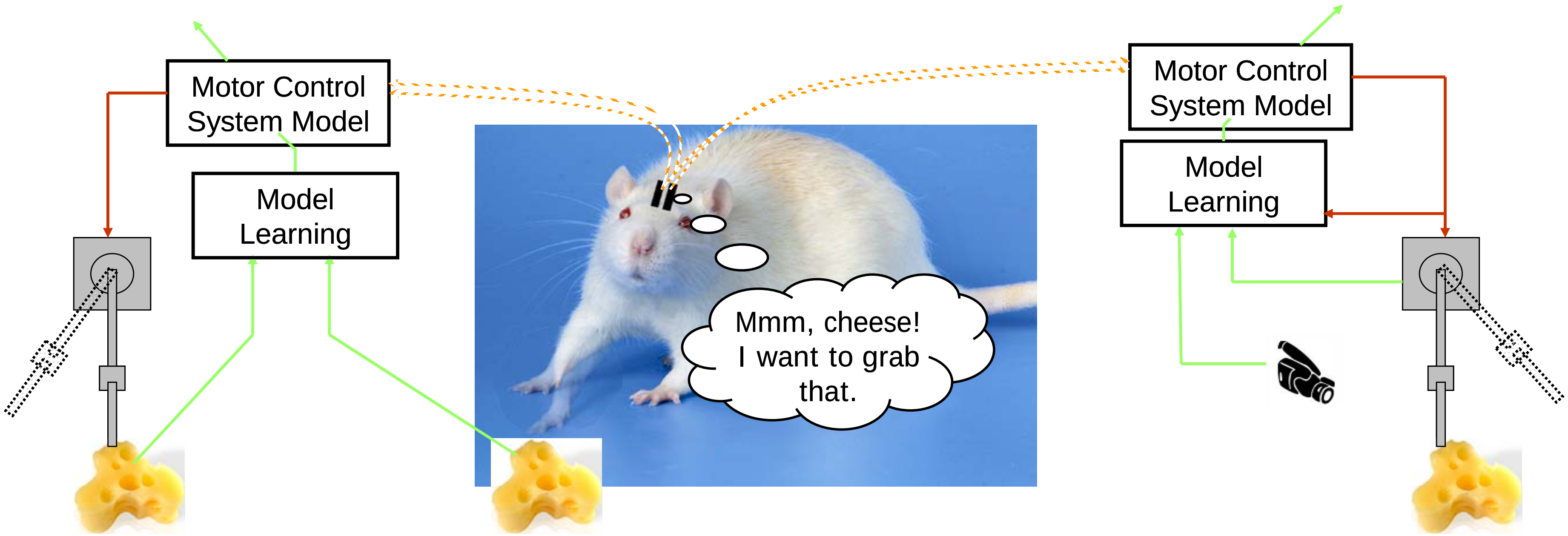
Brain Machine Interfaces (BMIs)

- Motor BMIs
  - Translate brain electrical activity into commands to external devices
  - Command BMIs or BCIs– EEG-based
  - Trajectory control BMIs – based on neuronal firings/fields

- Signal processing
  - Many possible models
  - Real-time (20-200 ms)
  - Feedback and training



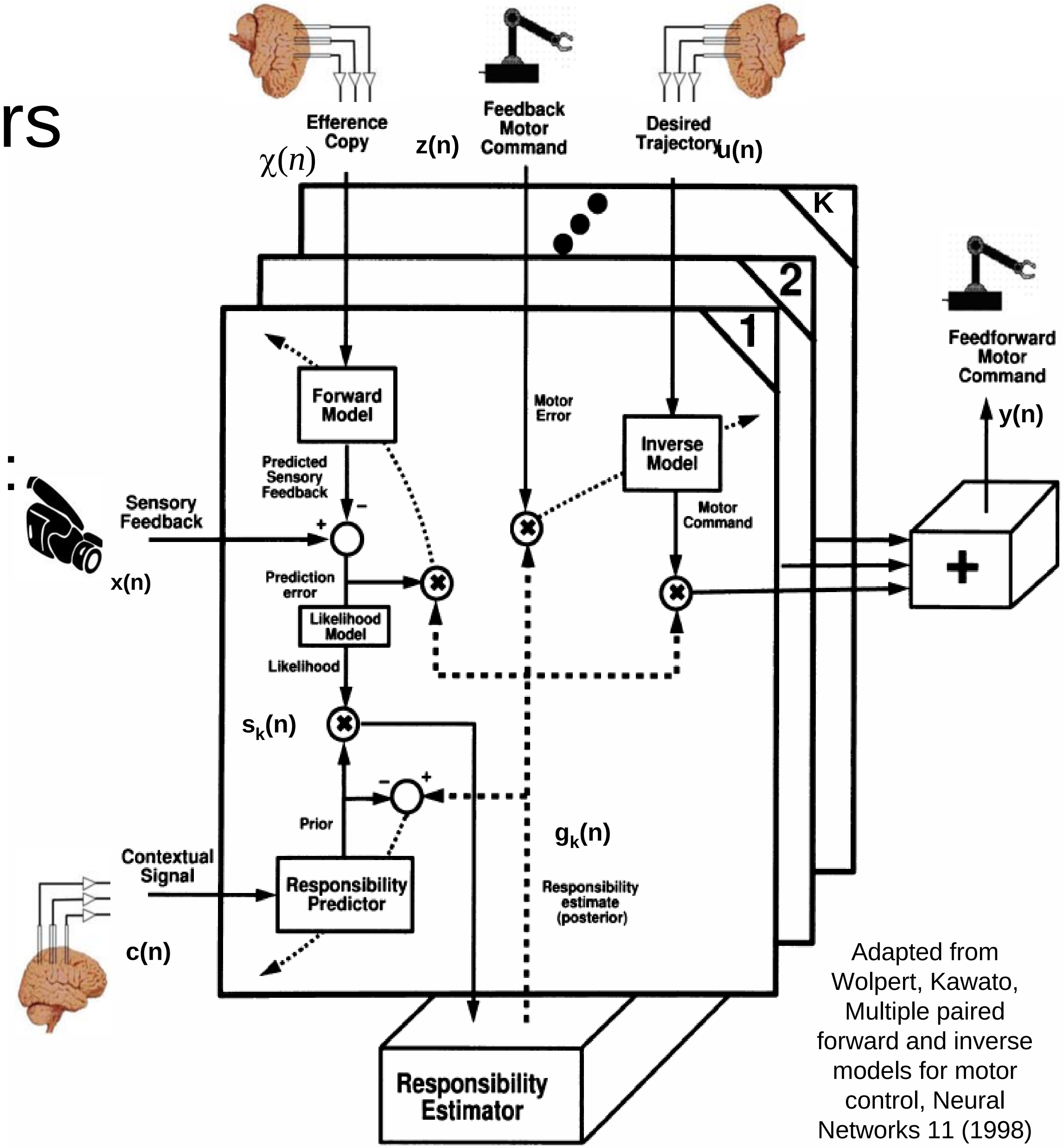
Traditional vs. DDDBMI model



- Desired signal provided by the model, not the patient
- Internal error estimates from forward-inverse models
- Switching between pairs as in a “mixture of experts”

DDDBMI model (inspired by Kawato's brain model)

- multiple model pairs
  - forward (planning): sensory input from motor commands
  - inverse (execution): motor commands from trajectory info
- output combines several models
  - data dependent
  - dynamic

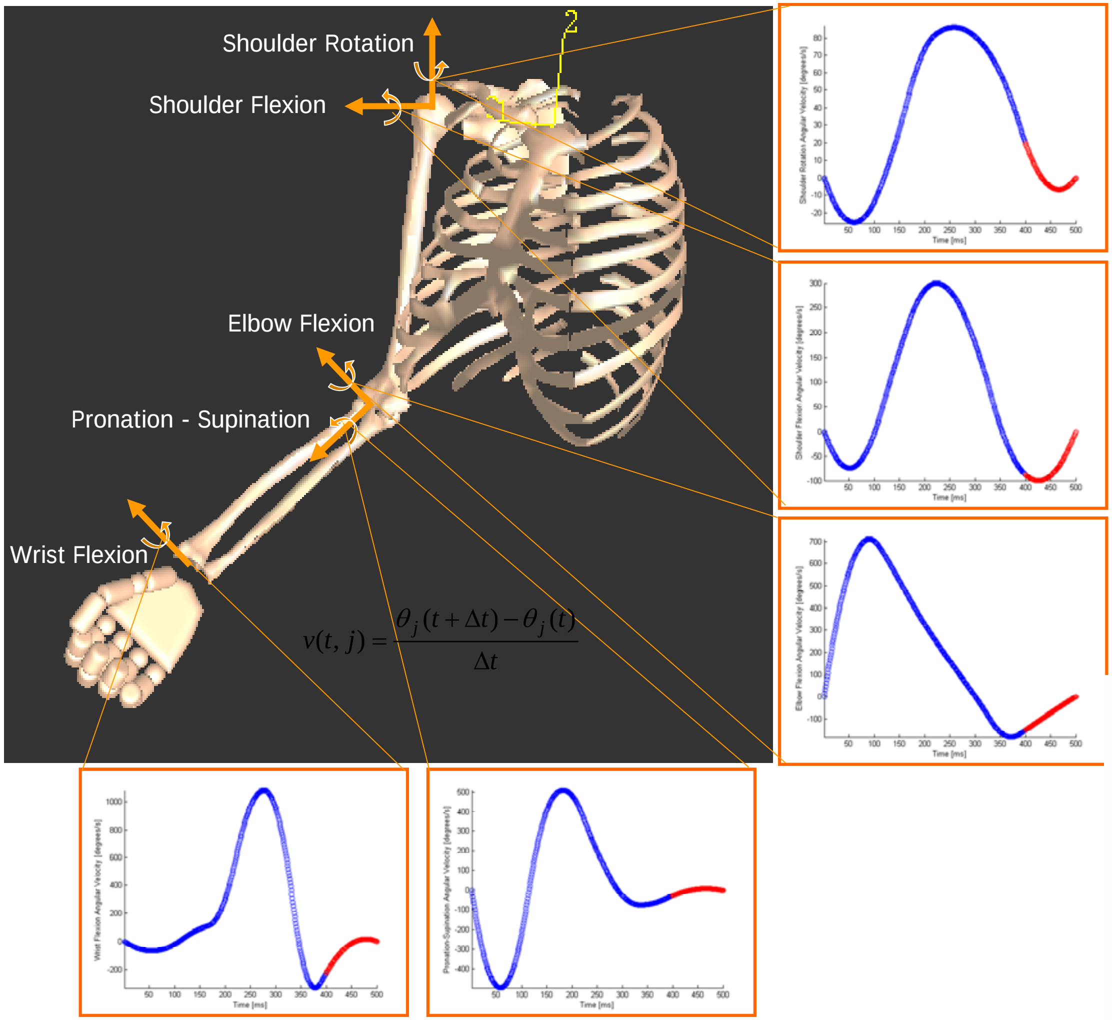


Adapted from Wolpert, Kawato, Multiple paired forward and inverse models for motor control, Neural Networks 11 (1998)

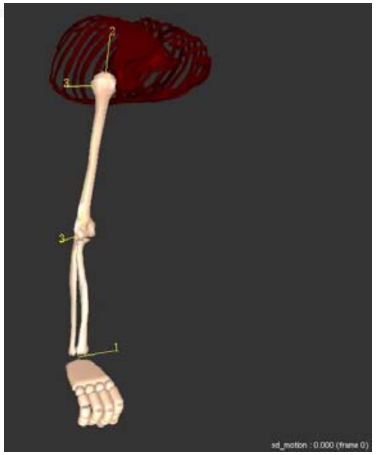
General considerations

- Number of pairs of “internal” models
  - 10s – 100s for simple tasks (e.g. press lever)
  - 1000s (?) for complex tasks
- Types of “internal” models
  - Linear (filters): Wiener, NLMS, PVA, …
  - Nonlinear (neural nets): TDNN, RMLP, RNN, NMCLM
  - State-based: Kalman filters, Bayesian classifiers, HMMs
- Complexity of models
  - $O(n)$, $O(n^2)$, $O(mn^2)$, $O(n^3)$, …
  - for n neurons, m models

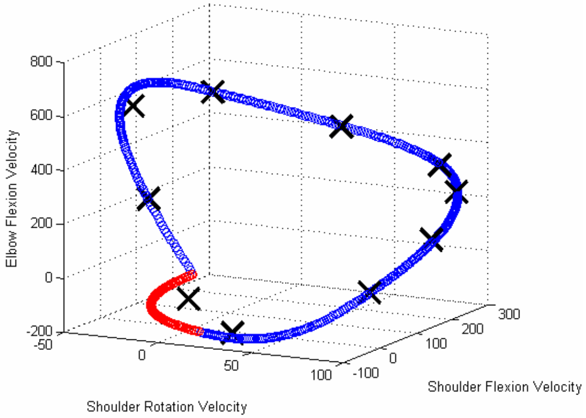
Movemes



$$v(t,j)=\frac{\theta_j(t+\Delta t)-\theta_j(t)}{\Delta t}$$

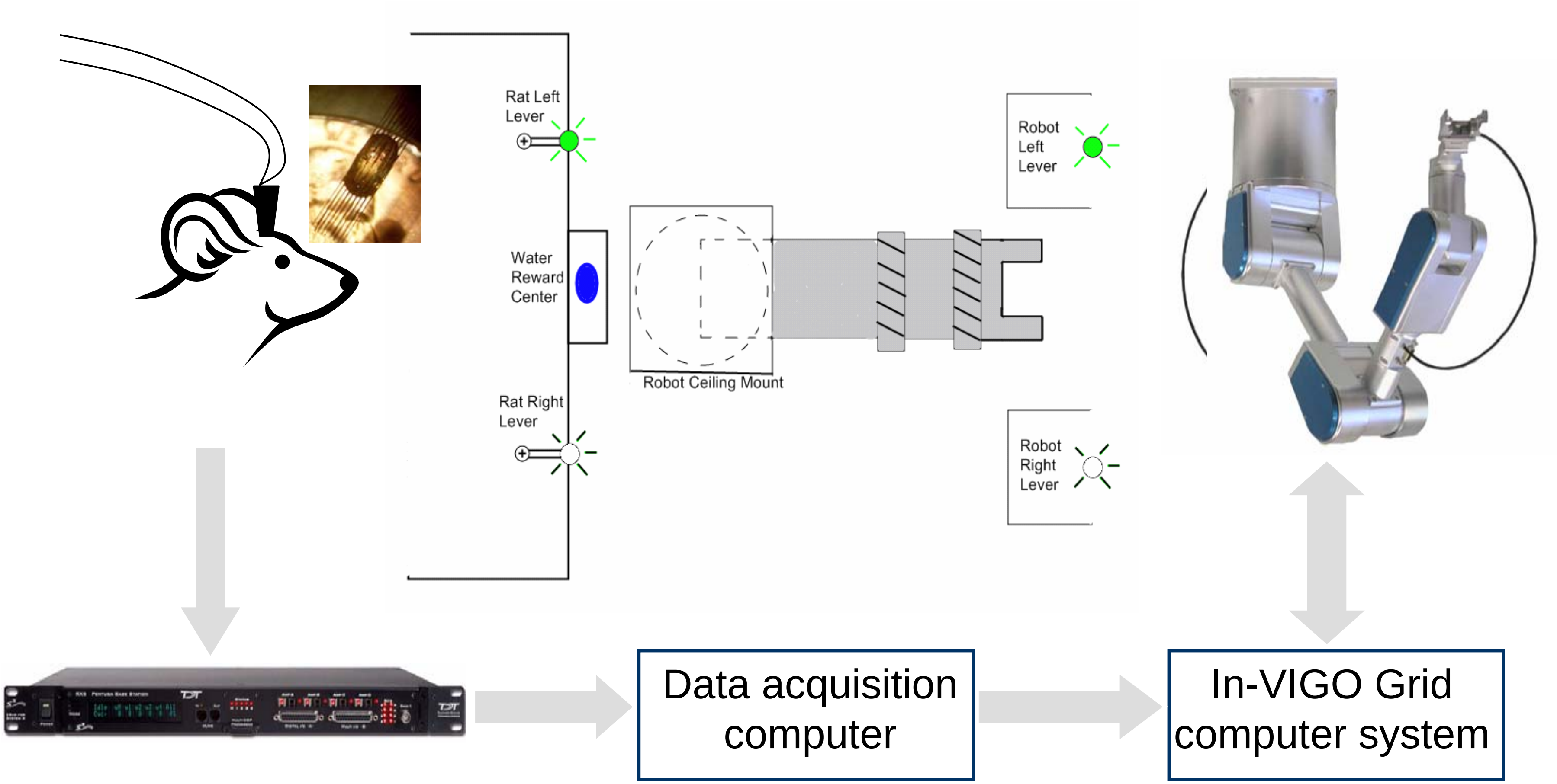


Clustering



[DiGiovanna, Sanchez, Fregly, and Principe, "Arm motion reconstruction via clustering in joint angle space" IJCNN, 2006]

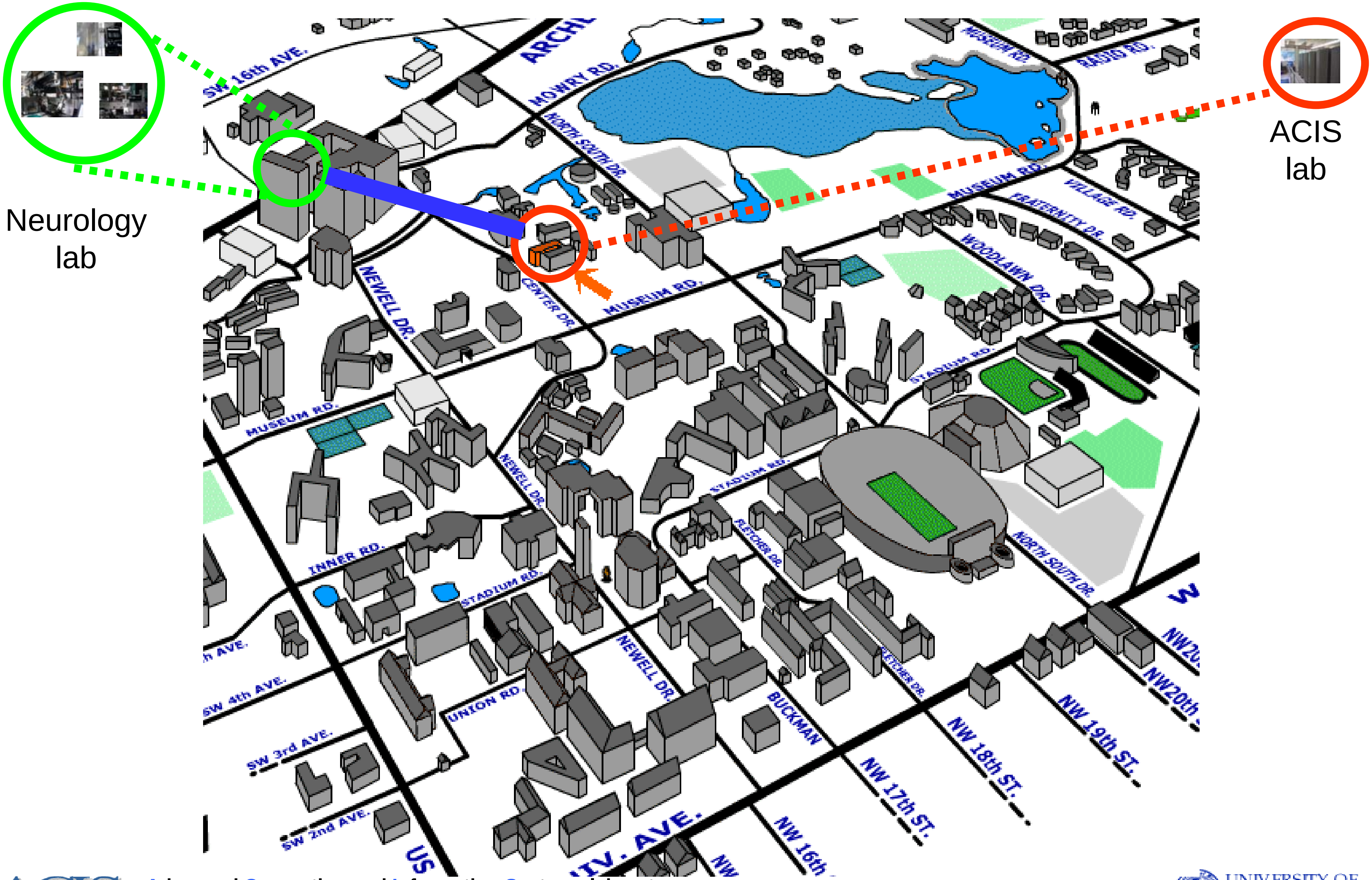
Experimental Setup



Data acquisition computer

In-VIGO Grid computer system

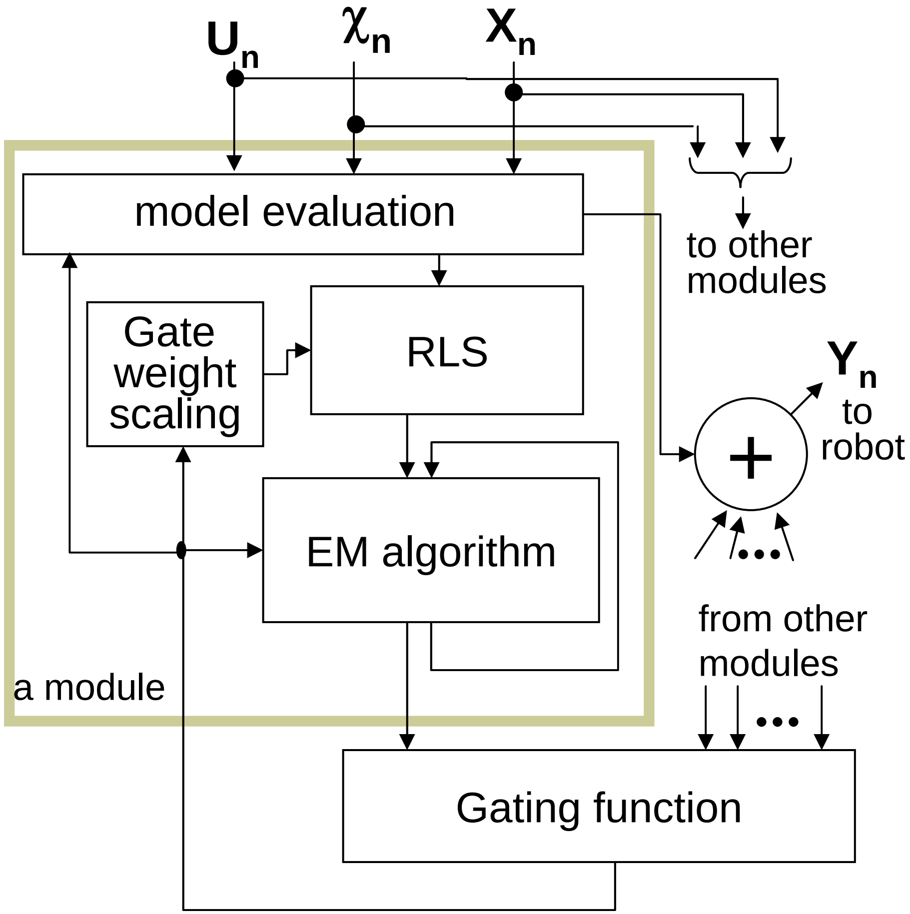
Experimental setup distribution

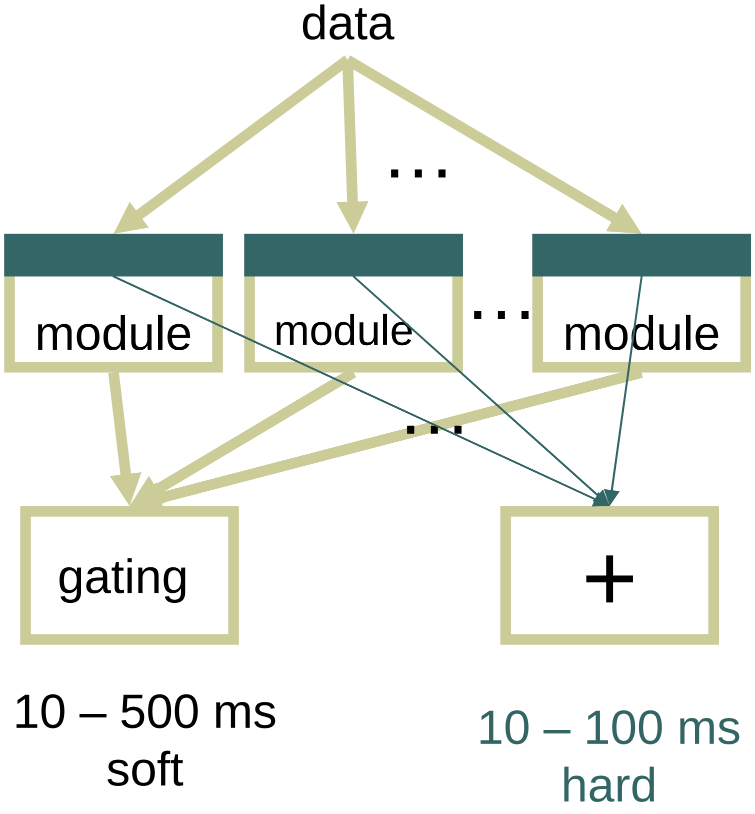


Neurology lab

ACIS lab

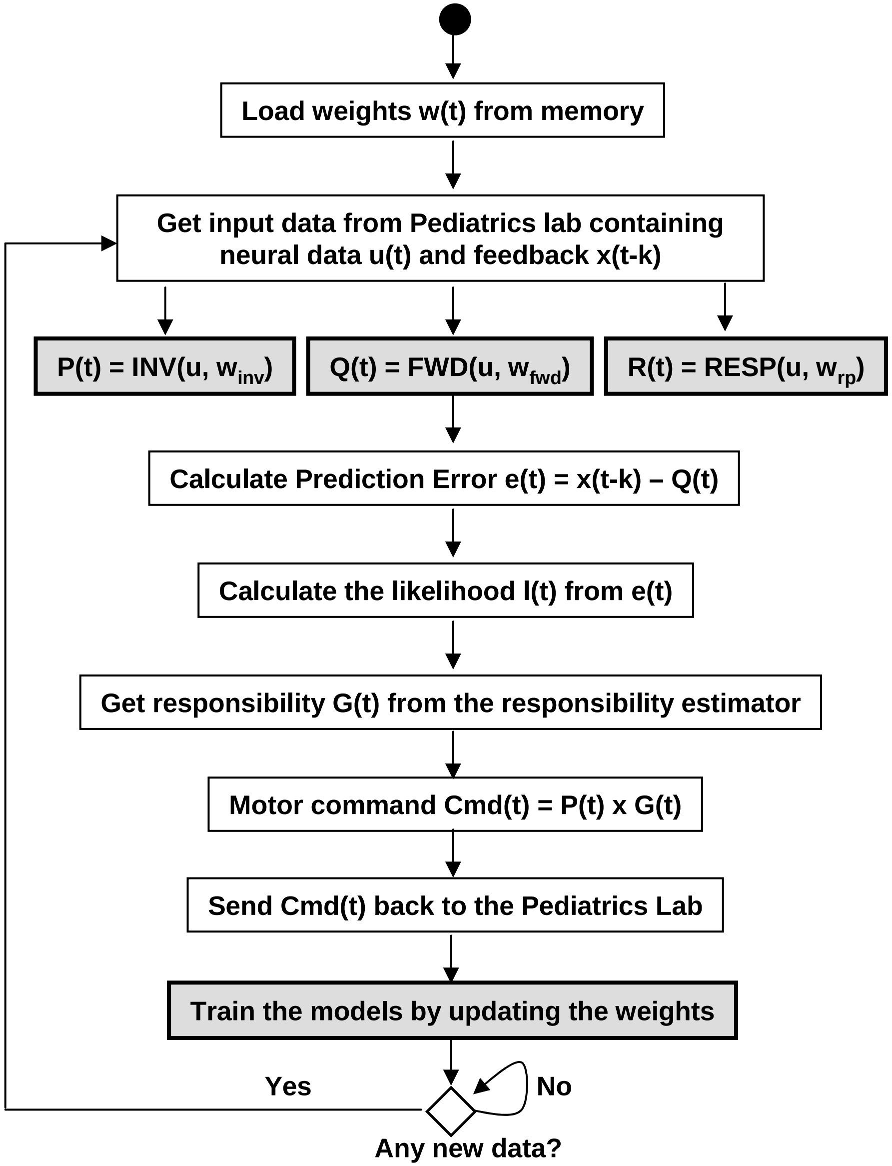
Basic computation structure





- Online – real-time (hard and soft deadlines)
- Offline – recreation of experiments from data in storage

Preliminary computational estimates



Load weights w(t) from memory

Get input data from Pediatrics lab containing neural data u(t) and feedback x(t-k)

$P(t) = INV(u, w_{inv})$

$Q(t) = FWD(u, w_{fwd})$

$R(t) = RESP(u, w_{rp})$

Calculate Prediction Error e(t) = x(t-k) – Q(t)

Calculate the likelihood l(t) from e(t)

Get responsibility G(t) from the responsibility estimator

Motor command Cmd(t) = P(t) x G(t)

Send Cmd(t) back to the Pediatrics Lab

Train the models by updating the weights

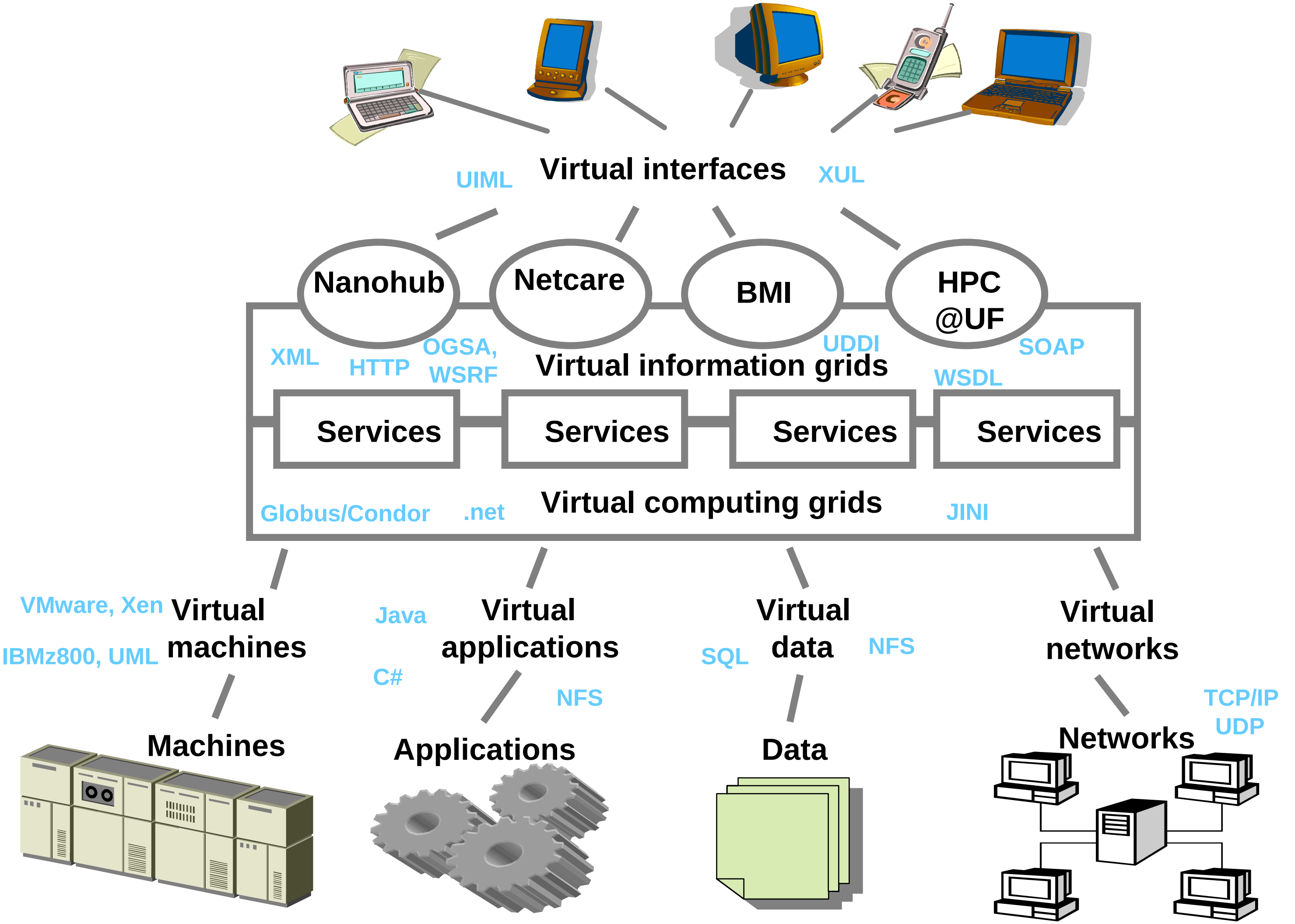
Complexity	**O(n)**	**$O(n^2)$**	**$O(n^3)$**
Avg. iteration time	277 µs	119 ms	115 s

Communication = 8 ~10 ms

Intel Pentium III 1.13GHz 512KB Cache, 1GB Memory, Fedora 4

32 channels from electrodes, 3 neurons per channel, 10 sliding taps, n = 960

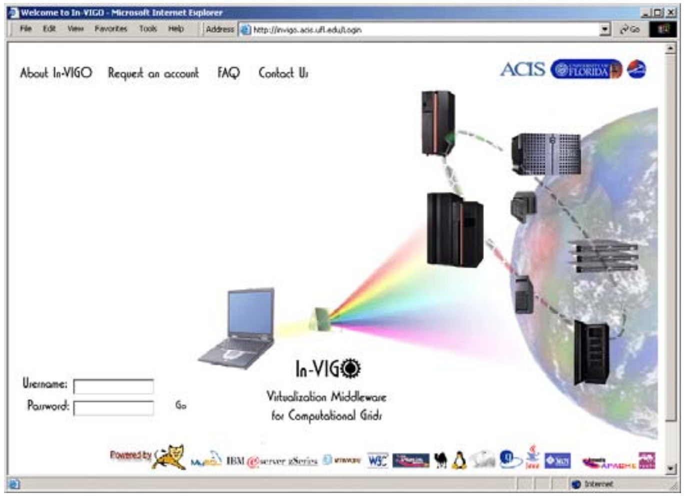
The In-VIGO approach

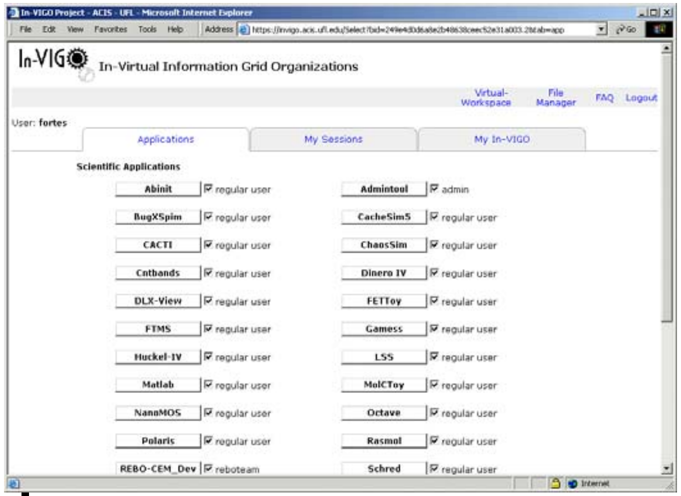


- ✓ local control, decentralized management
- ✓ open general-purpose standards
- ✓ non-trivial QoS

BMI portal

- Reservation of resources
  - For online experiments
- Access to data sets
  - For replay and analysis of experiments
- Specification of models
  - For use in either offline or online experiments
- Access to computational tools
  - For analysis, simulation, visualization ….





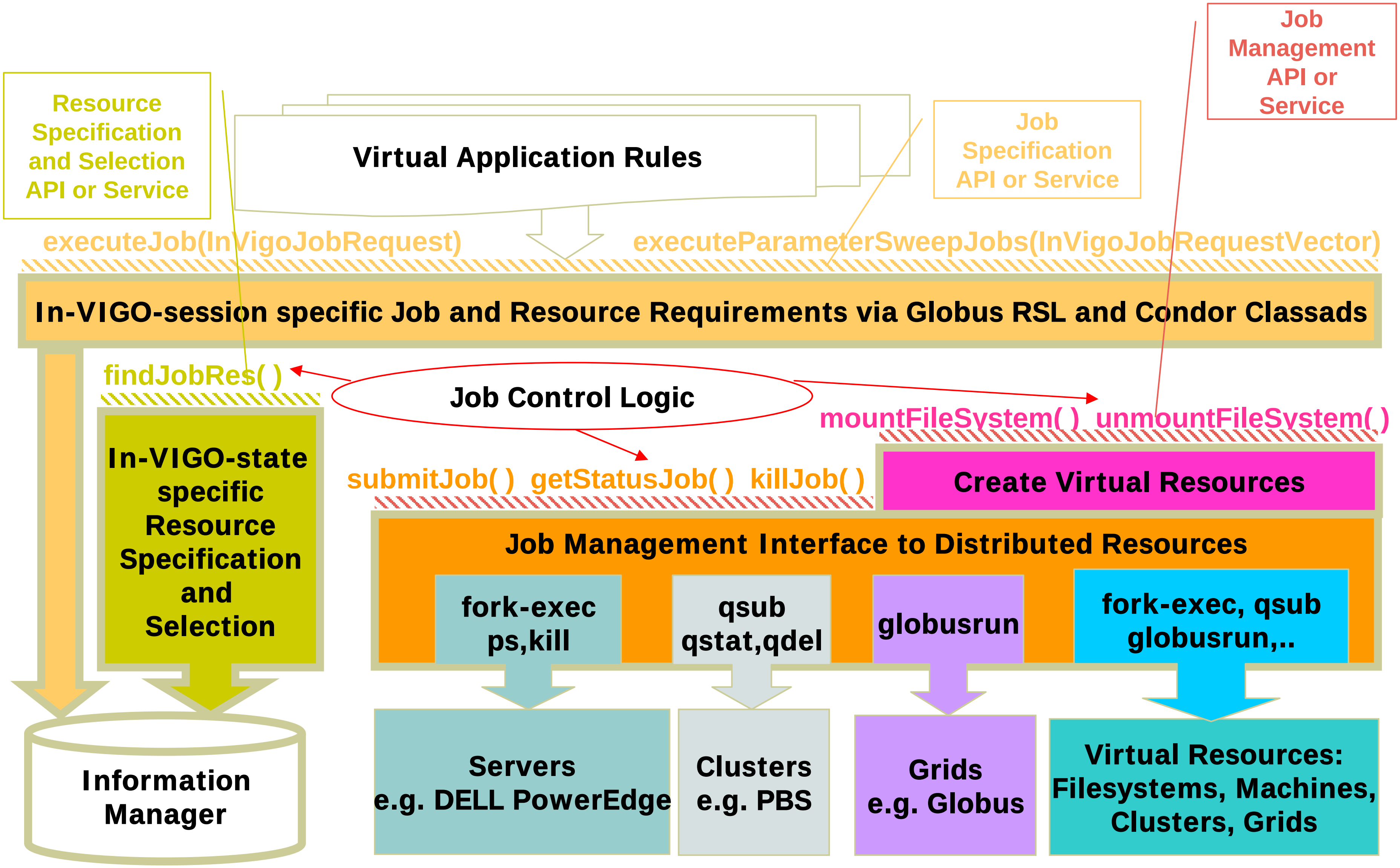
In-Virtual Information Grid Organizations

User: fortes

Scientific Applications

Abinit	☑ regular user	Admintool	☑ admin
BugXSpim	☑ regular user	CacheSim5	☑ regular user
CACTI	☑ regular user	ChaosSim	☑ regular user
Cntbands	☑ regular user	Dinero IV	☑ regular user
DLX-View	☑ regular user	FETToy	☑ regular user
FTMS	☑ regular user	Gamess	☑ regular user
Huckel-IV	☑ regular user	LSS	☑ regular user
Matlab	☑ regular user	MolCToy	☑ regular user
NanoMOS	☑ regular user	Octave	☑ regular user
Polaris	☑ regular user	Rasmol	☑ regular user
REBO-CEM_Dev	☑ reboteam	Schred	☑ regular user

Resource management system



Resource Specification and Selection API or Service

Virtual Application Rules

Job Specification API or Service

Job Management API or Service

executeJob(InVigoJobRequest)

executeParameterSweepJobs(InVigoJobRequestVector)

In-VIGO-session specific Job and Resource Requirements via Globus RSL and Condor Classads

findJobRes( )

Job Control Logic

mountFileSystem( ) unmountFileSystem( )

In-VIGO-state specific Resource Specification and Selection

submitJob( ) getStatusJob( ) killJob( )

Create Virtual Resources

Job Management Interface to Distributed Resources

fork-exec ps,kill

qsub qstat,qdel

globusrun

fork-exec, qsub globusrun,..

Information Manager

Servers e.g. DELL PowerEdge

Clusters e.g. PBS

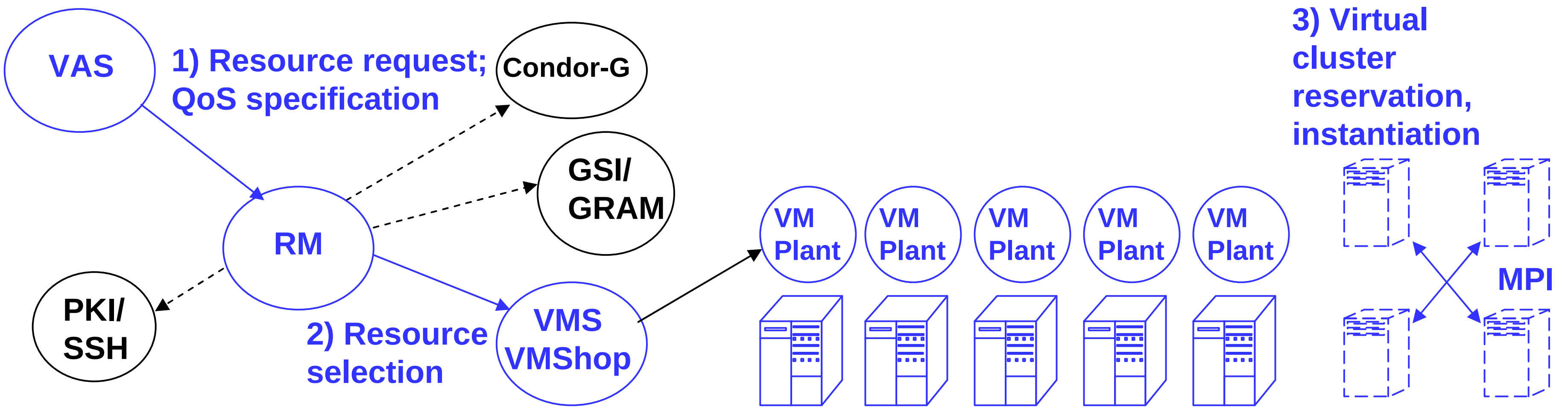
Grids e.g. Globus

Virtual Resources: Filesystems, Machines, Clusters, Grids

Slide provided by S. Adabala

Requirements for Grid-based DDBMIs

1. resource discovery based on quality of service specifications and scheduling based on virtual machine reservations,
2. dynamic steering of applications to computing resources based on run-time feedback from application inputs.



Closing remarks

- Multidisciplinary collaboration
- Research goals
  - novel BMI system using motor control theory.
  - understanding of human brain dynamics.
  - definition of movemes in motion to provide structure for our model.
  - middleware to run closed-loop, real-time BMI experiments via grid computing
- Preliminary results
  - movemes framework for movement specification and decomposition
  - analysis of spatial neural activity
  - characterization of communication/computation delays

Acknowledgements

- Graduate students
  - Shalom Darmanjian
  - Jack DiGiovanna
  - Prapaporn Rattanatamrong
  - Ming Zhao
- BJ Fregly
  - collaboration and support with the musculoskeletal modeling.
- Wendy Murray (Stanford University)
  - for use of the skeletal model.
- National Science Foundation
  - NSF Grant No. CNS-0540304 under the DDDAS program

Publications

- J. DiGiovanna, J. C. Sanchez, BJ Fregly, and J. C. Principe, "Arm motion reconstruction via clustering in joint angle space" presented at IEEE Intl. Joint Conf. Neural Networks, Vancouver, BC, 2006

- J. DiGiovanna, J. C. Sanchez, and J. C. Principe, "Improved linear BMI systems via population averaging" submitted to IEEE EMBS conference, New York, NY, 2006