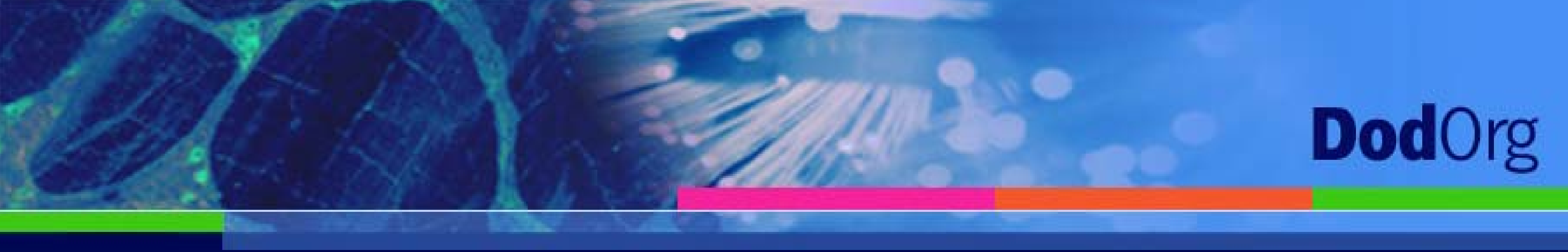


Digital On-Demand Computing Organism for Real-time Systems
DodOrg

Jürgen Becker, Uwe Brinkschulte, Jörg Henkel, Wolfgang Karl, Heinz Wörn

SPP OC Kolloquium

DFG SPP 1183 “Organic Computing”

Lübeck, September 13th and 14th, 2007

Overview

- ▶ Project Motivation: DodOrg Application Scenario
- ▶ Results of the 1st Project Phase:
  - Organic Robot Control
  - Organic Middleware
  - Organic Monitoring
  - Ultra Low Power Processing
  - Organic Processing Cells
- ▶ Future Work (Plasticity, Dynamics, and Stability)
  - Organic Robot Control
  - Organic Middleware
  - Organic Monitoring
  - Distributed Low Power Management
  - Organic Processing Cells
- ▶ Conclusions

Robot Control System: Dynamic Scenario


Defective fan


Klaus


Voltage peak


Maintenance

Classic Scenario:

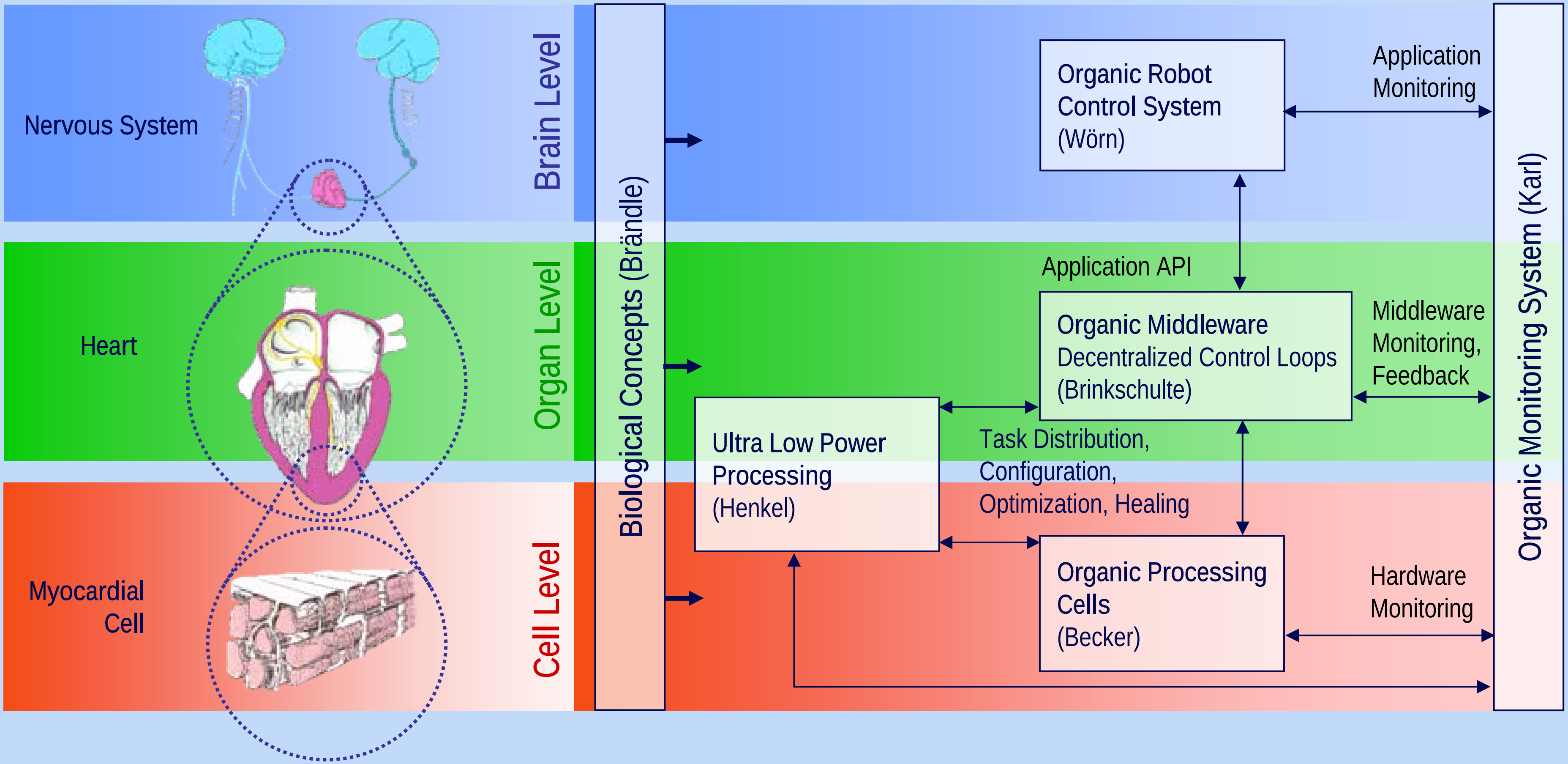
- ►Only those scenarios can be handled:
  - that had been considered in advance
  - where the cause can be detected
  - where the corresponding reaction had been explicitly programmed
- ►Lack of adaptation leads to insufficient reactions (e.g. shutdown …)

DodOrg Scenario:

- ►System reaction based on indications (higher level of abstraction)
  - e.g. CRC/bit error rate, network bottleneck, change of robot model
- ►Proper reaction possible even if:
  - Scenario was not considered in advance
  - Cause was not detected
  - Reaction was not explicitly programmed
- ►Flexible response to changed environmental situation

The Layer Model

DodOrg



Nervous System

Heart

Myocardial Cell

Brain Level

Organ Level

Cell Level

Biological Concepts (Brändle)

Organic Robot Control System (Wörn)

Application Monitoring

Application API

Organic Middleware Decentralized Control Loops (Brinkschulte)

Middleware Monitoring, Feedback

Ultra Low Power Processing (Henkel)

Task Distribution, Configuration, Optimization, Healing

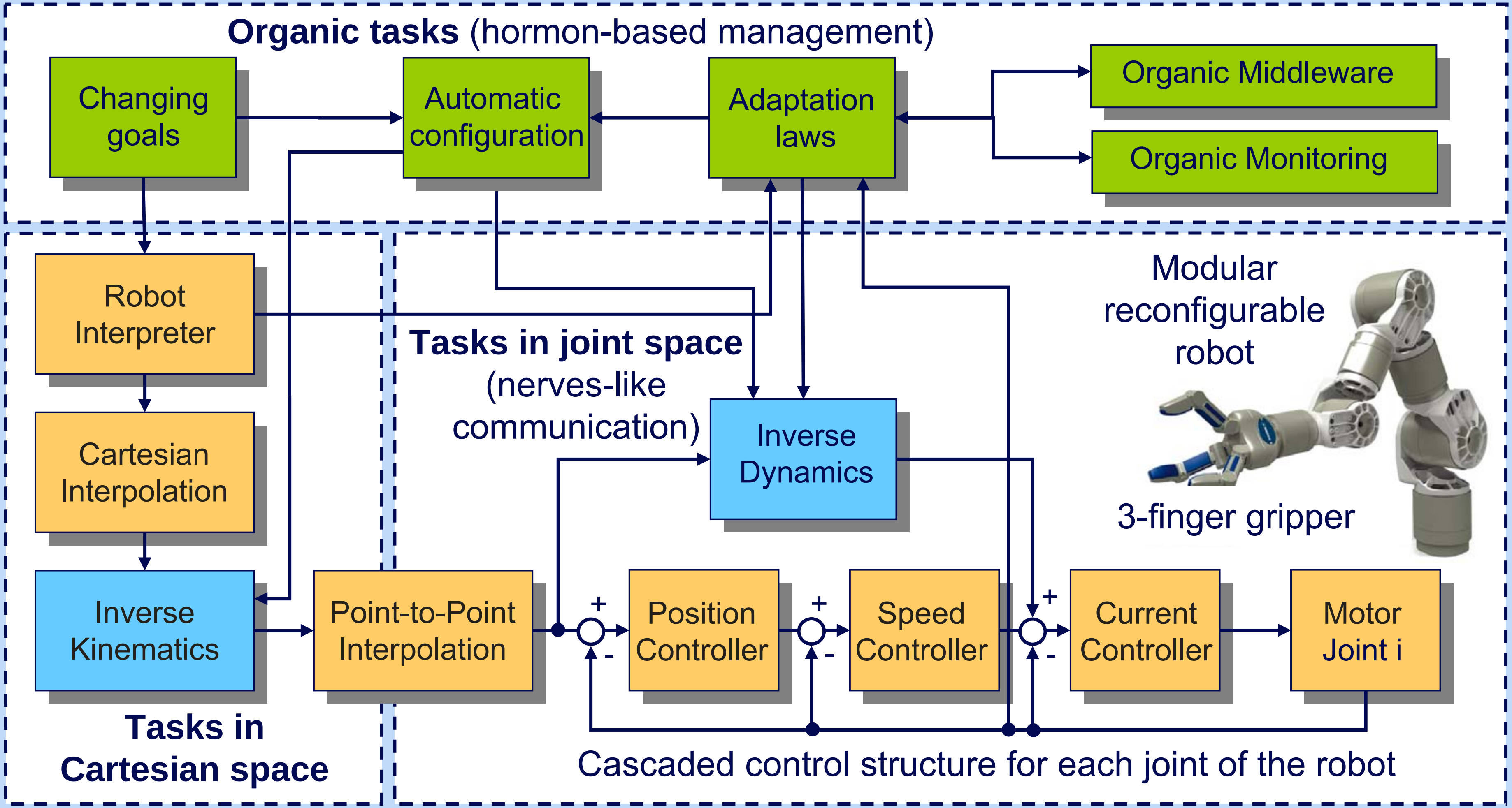
Organic Processing Cells (Becker)

Hardware Monitoring

Organic Monitoring System (Karl)

Organic Robot Control: Architecture
(Prof. Wörn)

Current Phase of
DodOrg



Organic tasks (hormon-based management)

Changing goals

Automatic configuration

Adaptation laws

Organic Middleware

Organic Monitoring

Robot Interpreter

Cartesian Interpolation

Inverse Kinematics

Tasks in Cartesian space

Tasks in joint space (nerves-like communication)

Inverse Dynamics

Modular reconfigurable robot

3-finger gripper

Point-to-Point Interpolation

Position Controller

Speed Controller

Current Controller

Motor Joint i

Cascaded control structure for each joint of the robot

Organic Robot Control: Results of first project phase (Prof. Wörn)

- ► Development of Organic Robot Control Architecture
  - ▪ Fast and efficient self-configuration and self-adaptation to ensure real-time constraints and efficiency
  - ▪ Development of large variety of different tasks e. g. robot-specific tasks and adaptation tasks
- ► Development of Functionalities for Self-Organization and Self-Configuration
  - ▪ Plug-and-play behavior for different kinematics
  - ▪ Fully automatic generation of kinematics and dynamics of the manipulator
  - ▪ Development of easy-to-use configuration system
- ► Simulation and Validation of the Results
  - ▪ Fulfillment of hard real-time requirements
  - ▪ Very fast computation of the inverse kinematics model for an articulated robot (in less than 0.005 ms)

Hexapod

Scara robot with spherical wrist

Articulated robot on a linear unit

Cartesian robot

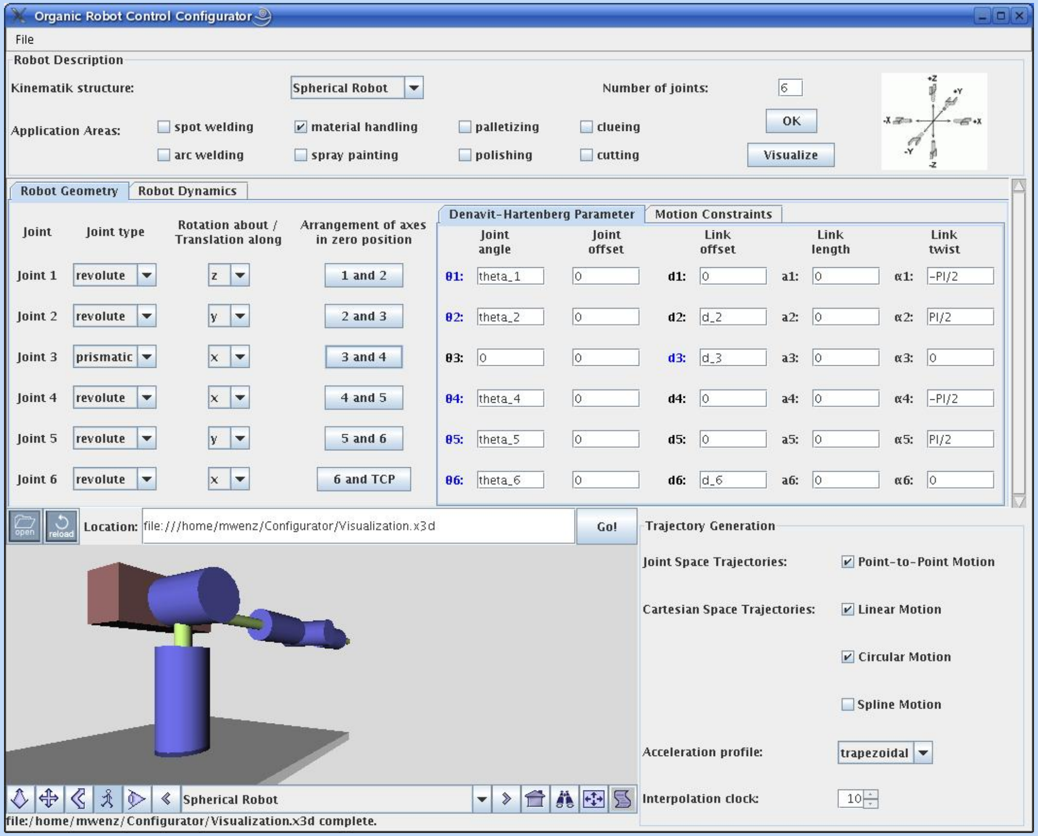
Spherical robot

Articulated robot

Cylindrical robot

Organic Robot Control: Configurator
(Prof. Wörn)

- Development of a configuration system and a graphical user interface in order to configure the robot control on the fly (self-configuration)
  - The user describes the mechanical structure of a particular robot and the configurator automatically generates the motion control system
- The configurator opens up numerous selection and combination possibilities:
  - Number and type of joints
  - Arrangement of joints and constraints concerning their movement
  - Geometric dimensions, arm lengths, workspace
  - Dynamics data of each link: mass, location of mass center, inertia tensors
  - Interpolation clock, acceleration profile, interpolation algorithms that should be supported, e.g. ptp, linear, circular, spline, …



Robot Description

Kinematik structure: Spherical Robot

Number of joints: 6

Application Areas: ☐ spot welding ☑ material handling ☐ palletizing ☐ clueing ☐ arc welding ☐ spray painting ☐ polishing ☐ cutting

Joint	Joint type	Rotation about / Translation along	Arrangement of axes in zero position
Joint 1	revolute	z	1 and 2
Joint 2	revolute	y	2 and 3
Joint 3	prismatic	x	3 and 4
Joint 4	revolute	x	4 and 5
Joint 5	revolute	y	5 and 6
Joint 6	revolute	x	6 and TCP

	Joint angle	Joint offset		Link offset		Link length		Link twist
θ1:	theta_1	0	d1:	0	a1:	0	α1:	-PI/2
θ2:	theta_2	0	d2:	d_2	a2:	0	α2:	PI/2
θ3:	0	0	d3:	d_3	a3:	0	α3:	0
θ4:	theta_4	0	d4:	0	a4:	0	α4:	-PI/2
θ5:	theta_5	0	d5:	0	a5:	0	α5:	PI/2
θ6:	theta_6	0	d6:	d_6	a6:	0	α6:	0

Location: file:///home/mwenz/Configurator/Visualization.x3d

Spherical Robot

file:/home/mwenz/Configurator/Visualization.x3d complete.

Trajectory Generation

Joint Space Trajectories: ☑ Point-to-Point Motion

Cartesian Space Trajectories: ☑ Linear Motion ☑ Circular Motion ☐ Spline Motion

Acceleration profile: trapezoidal

Interpolation clock: 10

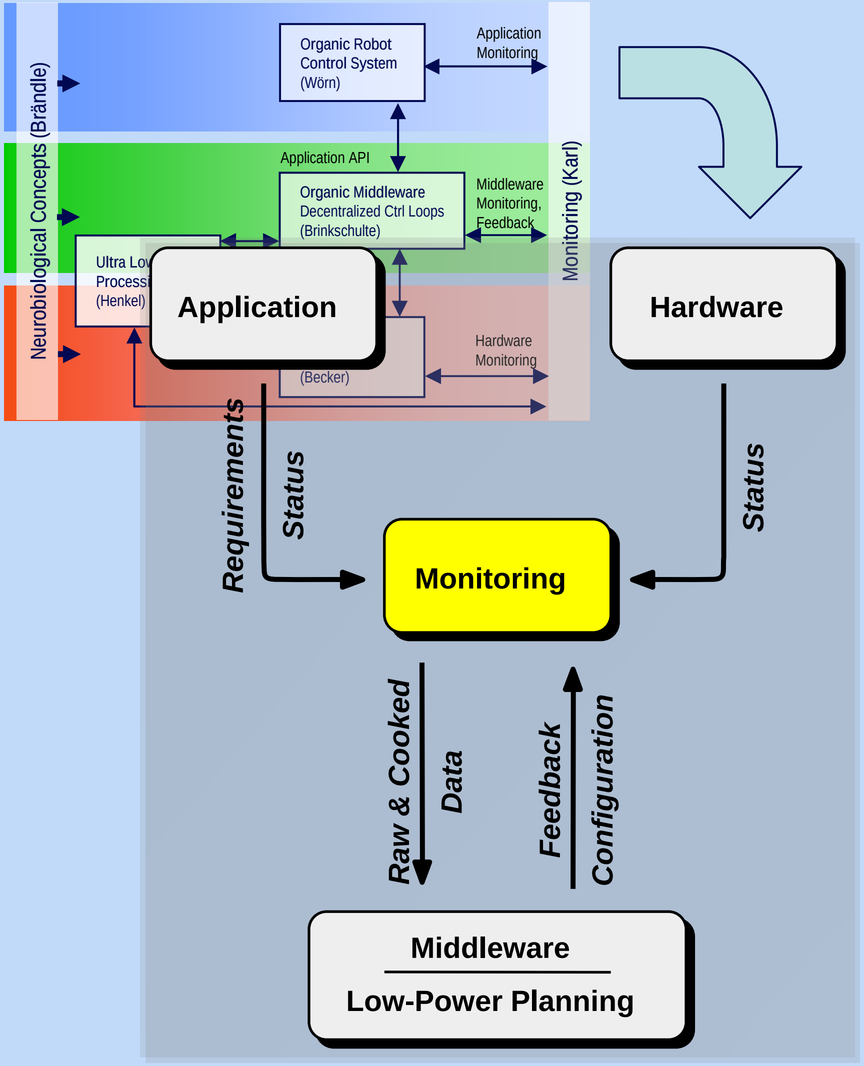
Organic Monitoring: Overview
(Prof. Karl)

►Aim

- Enable and Support Self-X Capabilities
- Focus on increased Self-Awareness

►Requirements

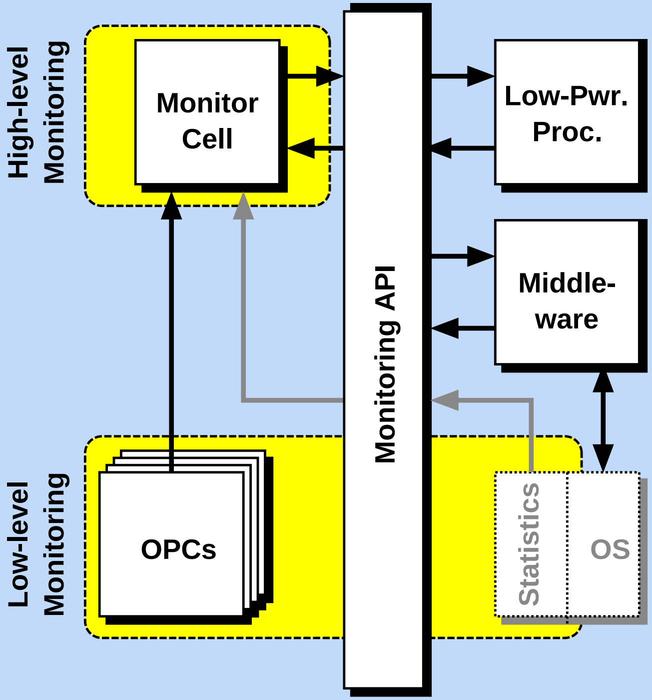
- Sustained System Monitoring
- Real-time Analysis and Evaluation
  - Correlation of (many) Events
  - Identification of Problems/Causes
- Semantic Data Compression
  - Processing at the Source of Data
  - Generation of Meta-Data
- Adaptivity (Reconfiguration)
- Interfacing



Organic Monitoring: Architectural Considerations (Prof. Karl)

Monitoring consists of

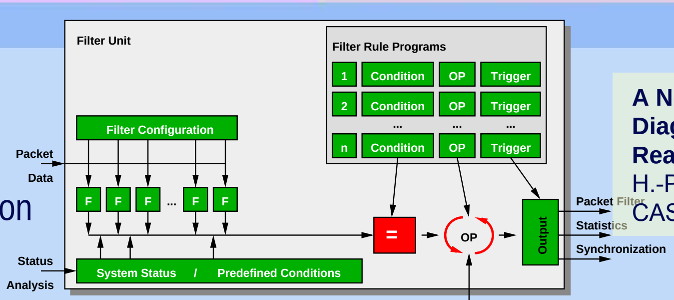
- ►Low-Level Monitoring
  - HW-Level: Fixed, but parametrizeable Monitoring Hardware in every Cell
  - SW-Level: System monitoring and data aggregation (comparable to /proc filesystem)
- ►High-Level Monitoring
  - Processing of Low-Level Monitoring information according to given rules
  - Correlation of various events into distilled information required by Middleware/Low-Power
  - Task of one or more Monitoring Cells
- ►Separation of Interface & Functionality
  - Monitor Capsule (Interface)
    - Standardized Query API
  - Monitoring Module (Functionality)
    - Domain-specific & Dynamically Reconfigurable
    - Extract, Process, & Store Data at Source
- ►Interface API
  - Provides uniform Interface to Monitoring Subsystem
  - Simple, extensible Communication Interface
  - Collection of Monitoring Resources
  - Management & Processing of Monitoring Rules
  - Generation of Events (Messengers), if required



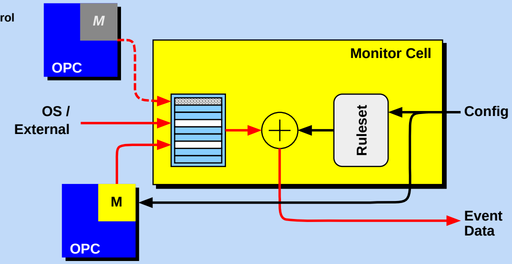
Organic Monitoring: Design Space Exploration (Prof. Karl)

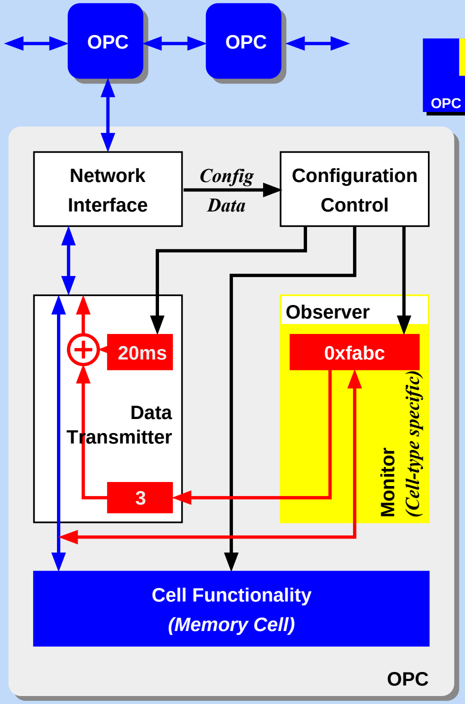
▶ Prototype Design Space Exploration

- Hardware Prototype
  - Real-time Monitoring and Semantic Data Compression
  - Application Scenario: Real-time Ethernet Diagnosis
- Software Prototype
  - Initial SW Prototype (Hierarchy & Protocol Considerations)
  - Data-Locality Optimization (DLO) through Self-Optimization
  - Rules and Metrics for Feedback-directed Optimization
- Combined Prototype
  - Monitoring Hierarchy: HW-Monitor and SW-based Monitor Cell
  - Protocol Refinements
  - Hardware & Software Costs
  - Application Scenario: DLO
- Distributed DodOrg Monitoring
  - Cluster Approach: Workstations are Processing Cells, Dedicated Workstation is Monitoring Cell
  - Creation and Destruction of Low-Level Monitoring Events
  - Control through Monitoring Cell
  - Work in progress



A Network Agent for Diagnosis and Analysis of Real-time Ethernet Networks
H.-P. Löb, R. Buchty, W. Karl
CASES 2006 Proceedings





A Monitoring Infrastructure for the Digital on-demand Computing Organism
R. Buchty, W. Karl
IWSOS2006 Proceedings

Automatic Data Locality Optimization through Self-Optimization
R. Buchty, J. Tao, W. Karl
IWSOS2006 Proceedings

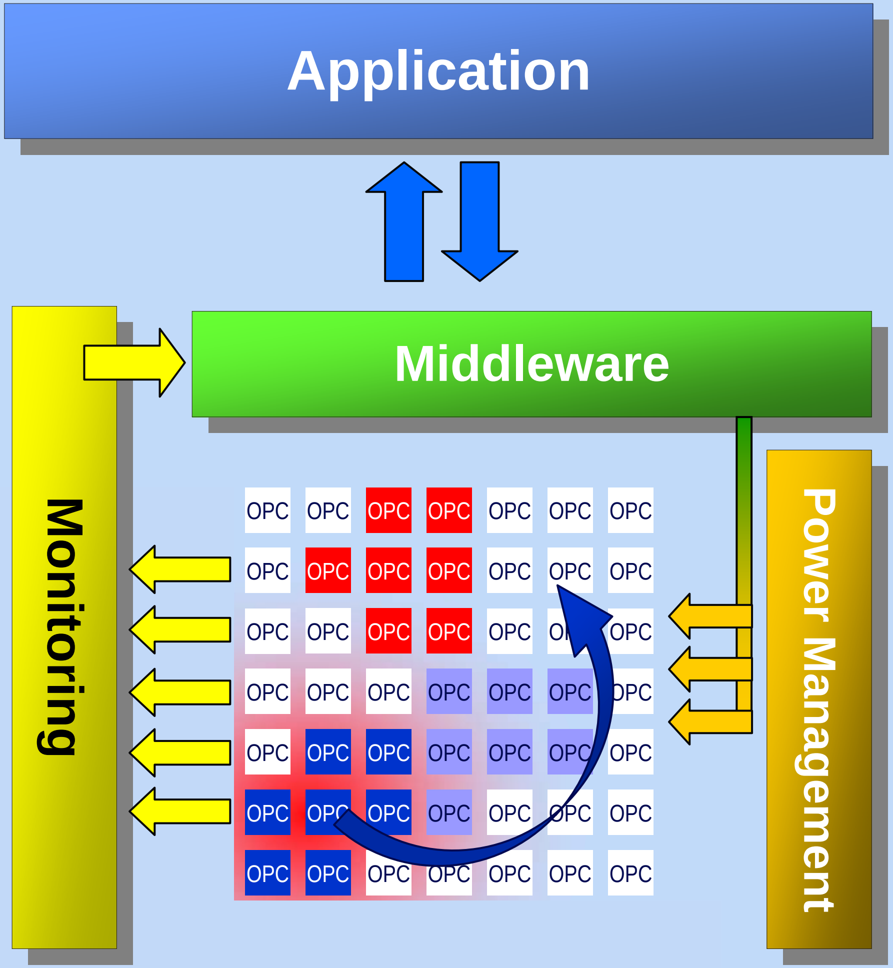
- Analysis of control theory
  - Demonstration of task load balancing by classical control loops

  Disadvantage:
  centralized structure → single point of failure

- Development of a basic organic middleware inspired by the hormone system

  ➔ Artificial Hormone System
  - Organic Processing Cells (OPC) work autonomously with a simple set of rules
  - Information is exchanged via hormones
  - Each OPC decides for itself if a hormone is relevant for itself

  ➔ By the interaction of many OPCs, the whole system is very powerful



Organic Middleware: Artificial Hormone System
(Prof. Brinkschulte)

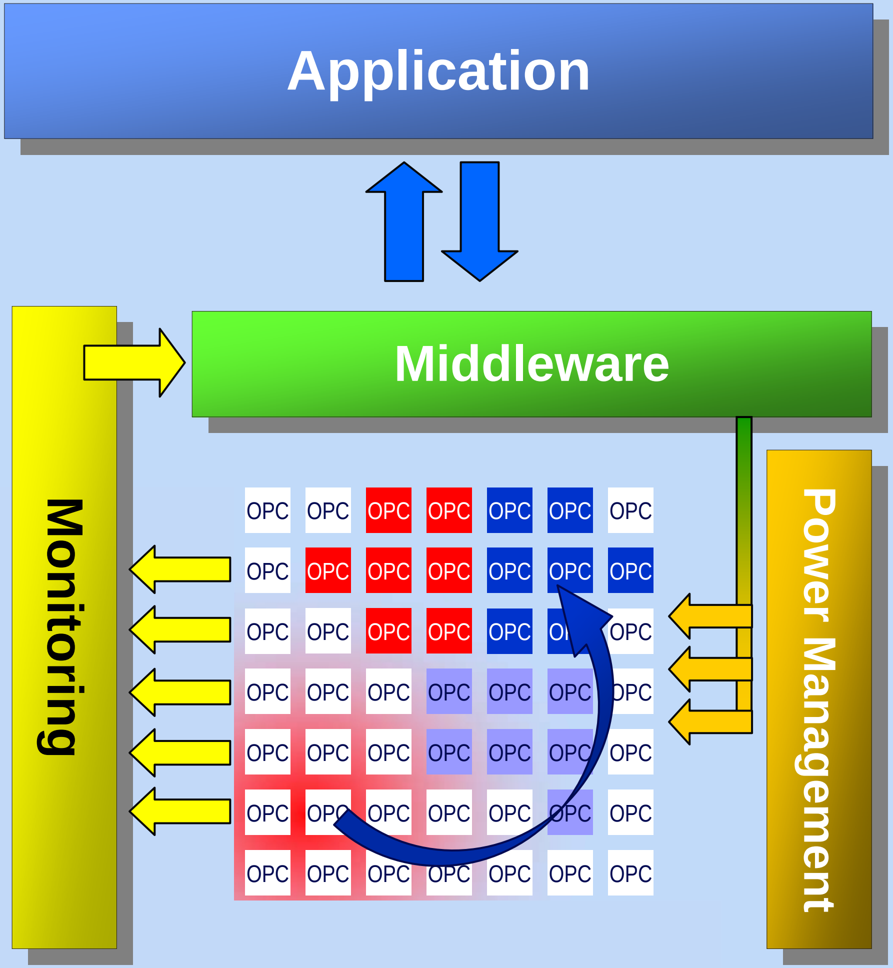
- Analysis of control theory
  - Demonstration of task load balancing by classical control loops

  Disadvantage:
  centralized structure → single point of failure

- Development of a basic organic middleware inspired by the hormone system

  ➔ Artificial Hormone System
  - Organic Processing Cells (OPC) work autonomously with a simple set of rules
  - Information is exchanged via hormones
  - Each OPC decides for itself if a hormone is relevant for itself

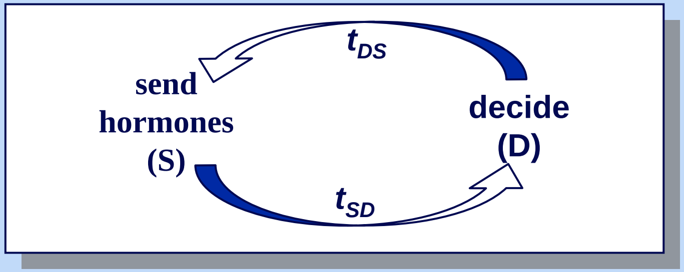
  ➔By the interaction of many OPCs, the whole system is very powerful



Organic Middleware: Hormone Cycle
(Prof. Brinkschulte)

- Theoretical analysis of the Artificial Hormone System
  - Each OPC follows a **simple hormone cycle**:
    - First, it sends out the hormones
    - Then, it waits to receive other hormones
    - Based on the information gathered, it autonomously decides whether and how to react
  - Definition of suitable **preconditions**:
    - To be able to get to mutual decisions, the waiting time has to be met
  - With these **preconditions**, it is possible to specify worst-case timing behaviors

- **Hormone cycle of a cell:**



- **Precondition for each hormone cycle:**

$t_{SD} \geq t_{DS} + 2\ t_C$
(t_C = communication time)

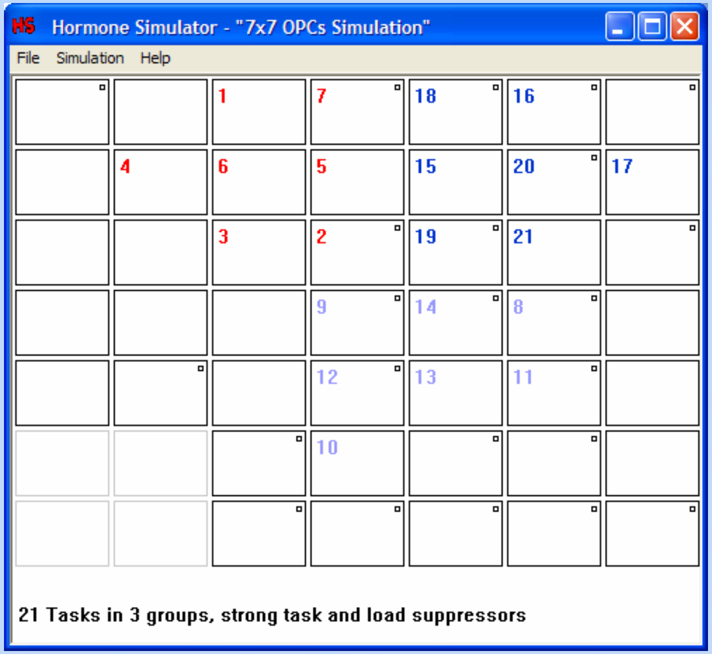
with t_{DS} as small as possible:
➔ $t_{DS} = 0$: $t_{SD} \geq 2\ t_C$

- **Worst-case time behavior for the task allocation:**

2m - 1 cycles
(with m = numbers of tasks)

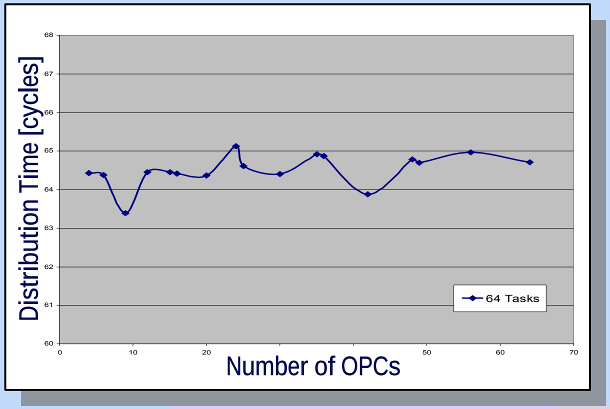
Organic Middleware: Hormone Simulator
(Prof. Brinkschulte)

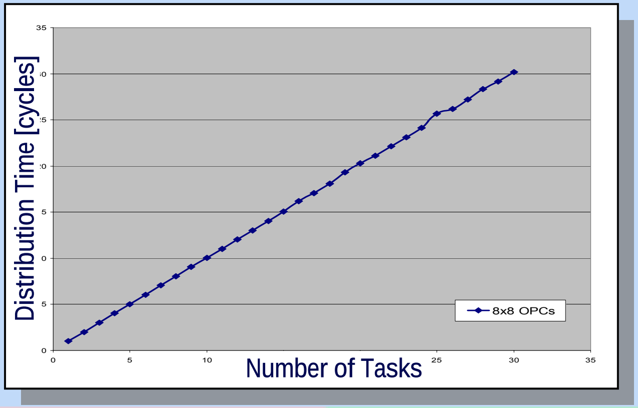
- ▶ Basic Prototype Middleware Implementation
  - Developed a simulator for task distribution as proof of concept
  - Mapping of tasks on virtual processing cells (asynchronous)
  - Occurrence of “organs” consisting of related tasks (same color)



21 Tasks in 3 groups, strong task and load suppressors

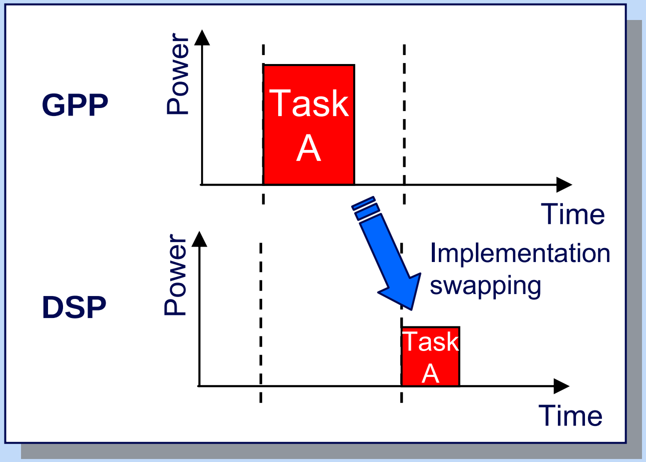
- ▶ Simulation and Testing
  - Simulation results confirm, that upper bounds apply



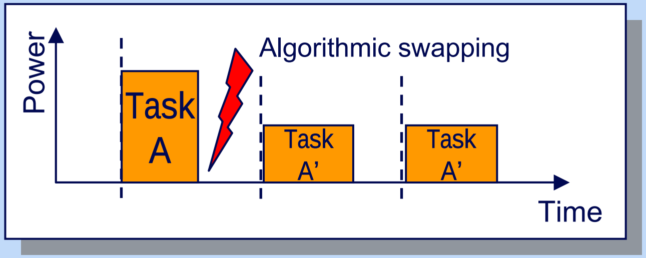


Ultra Low Power Processing: Exploiting Energy Savings Potential (Prof. Henkel)

- Tasks of the Ultra Low Power Processing
  - Reduction of power consumption while meeting given constraints (e.g. power budget, deadlines, etc.)
  - Reaction to changing constraints from within an organ
- Energy Savings Potential
  - Tasks consume energy depending on which OPC they are running on
  - Different algorithmic implementations of a task have different energy consumption
- Idea: seamless swap-on-the-fly
  - Mapping of tasks to OPCs (implementation swapping)
  - Choosing the algorithmic implementation (algorithmic swapping), e.g. matrix multiplication in sparse and normal matrices



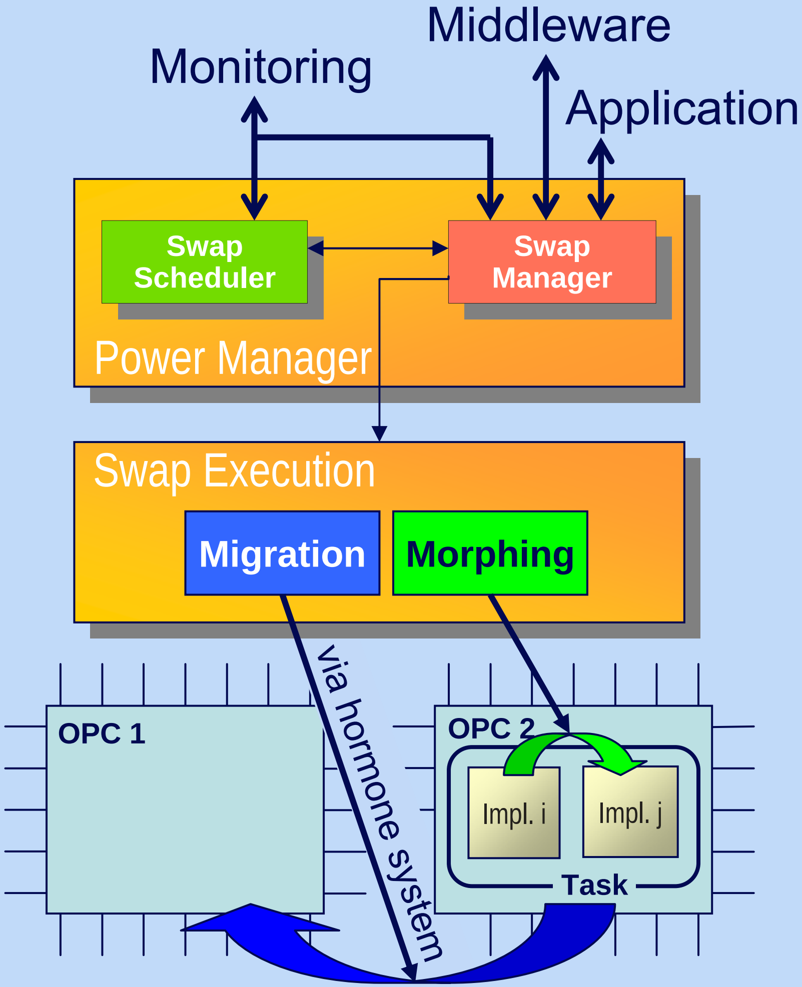
Energy consumption before and after **implementation** swapping (swap between micro-architectures or fabrics)



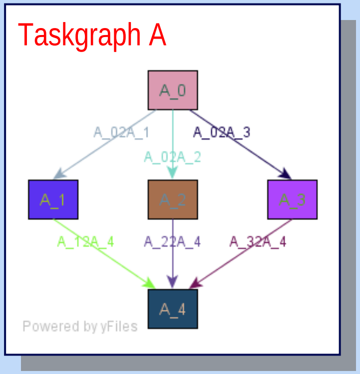
Energy consumption before and after **algorithm** swapping

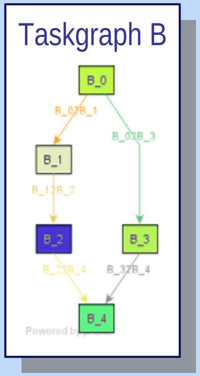
Ultra Low Power Processing: Swap-on-the-Fly Architecture (Prof. Henkel)

- ▶ Power Manager
  - Swap Scheduler
    - Computes good mapping of task to OPC within the organ
    - Here: Performance Effective Task Scheduling (PETS)
  - Swap Manager
    - Analyzes situation
    - Estimates whether or not a swap will pay off
    - Decides when a swap to new configuration is performed
- ▶ Swap Execution
  - Provides two services:
    task migration and task morphing
- ▶ OPC Level
  - Local real-time scheduler, e.g. EDF or RM
  - Dynamic voltage & frequency scaling, clock gating, …

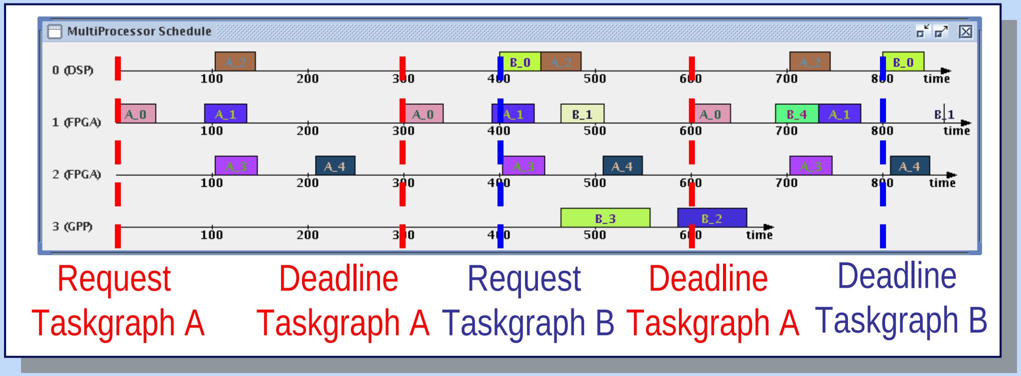


Ultra Low Power Processing: Mapping and Scheduling Results (Prof. Henkel)





Task graphs with request times/deadlines



Request Taskgraph A
Deadline Taskgraph A
Request Taskgraph B
Deadline Taskgraph A
Deadline Taskgraph B

Corresponding schedule on a heterogeneous MPSoC (by PETS)

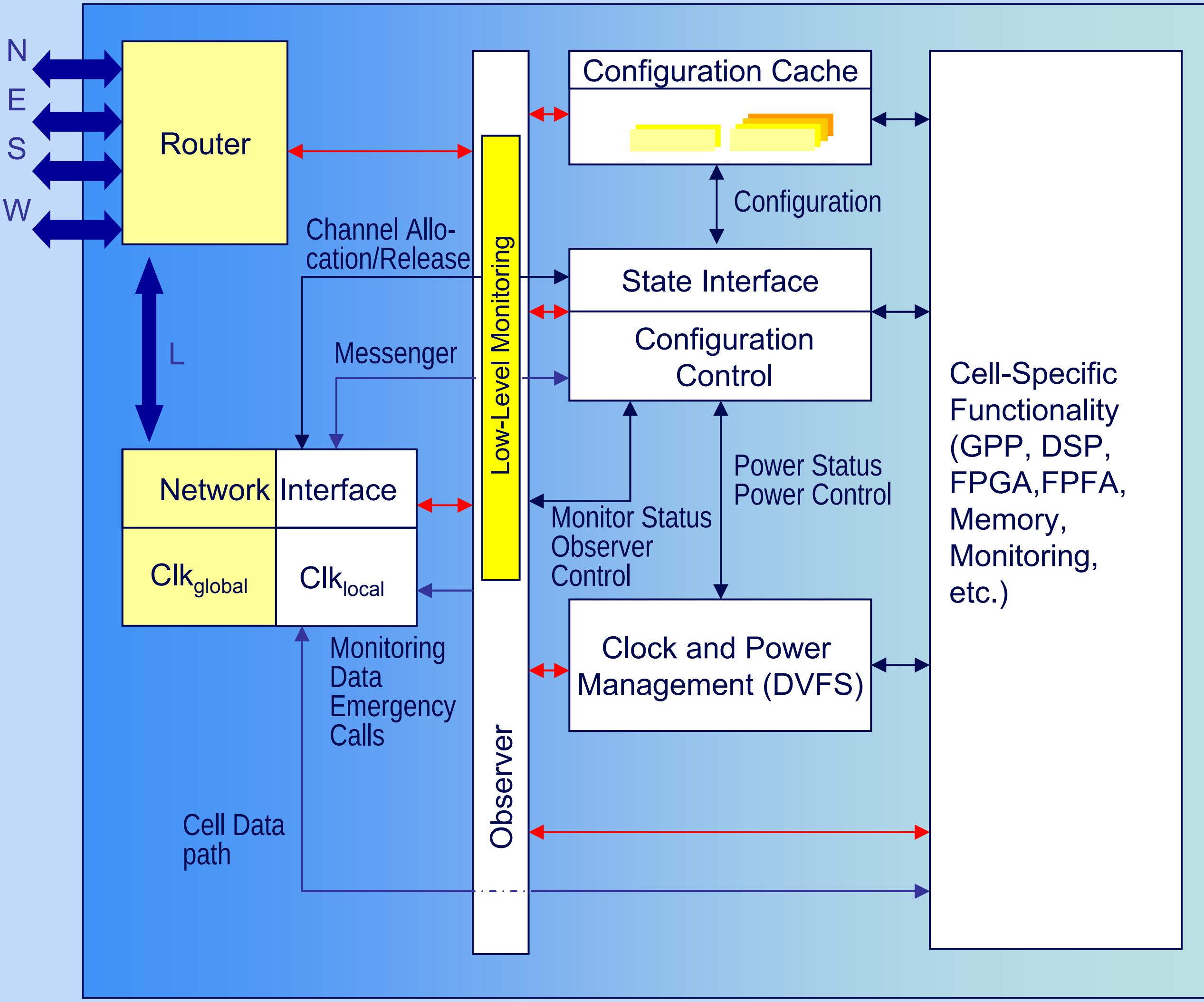
Test No.	Organic Processing Cells			
	I	II	III	IV
1	24 %	24 %	5 %	25 %
2	31 %	43 %	35 %	50 %
3	49 %	68 %	53 %	57 %
4	76 %	79 %	66 %	79 %
5	76 %	79 %	85 %	118 %

Utilization of OPCs after mapping with PETS

- ► Mapping by PETS in the Swap Scheduler
  - ▪ Considers utilization of OPCs
    - → More space for energy savings
    - → Incremental computation of new configurations
  - ▪ Task graph is divided into independent tasks
    - → Allows local scheduling by EDF or RM
- ► Scheduling by local entities
  - ▪ Guarantees are very conservative (max. utilization ≈ 7 %)
  - ▪ Without guarantees ≈ 65 % average utilization before violating deadlines

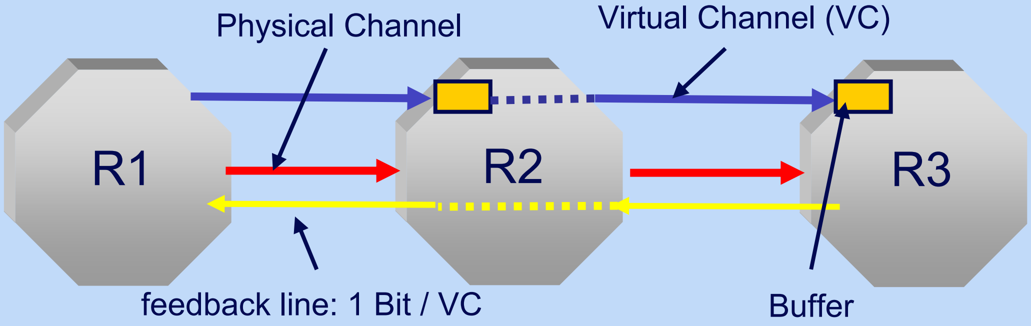
Organic Processing Cells: Cell Overview (Prof. Becker)

- Modularity
  - Same footprint for all cells
  - Common infrastructure
  - Cells can easily fill in for defective neighbors
  - Interface for higher-level functions (middleware, monitoring) stays the same
- Local intelligence
  - Power management
  - Basic monitoring facilities
  - Configuration management
  - Router
  - Built into each cell
- Cross-hierarchy Features
  - Monitoring
  - Low-Power Management
  - Hormone Broadcast



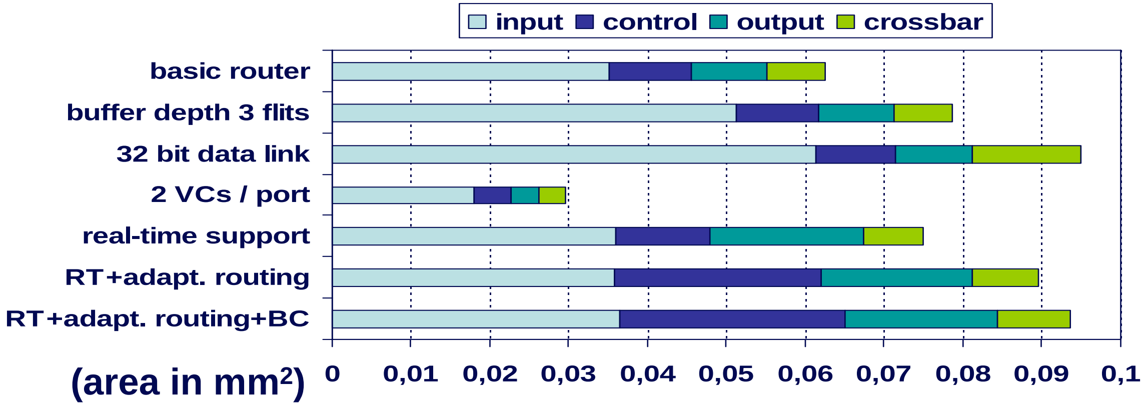
Organic Processing Cells: Router Architecture
(Prof. Becker)

- ▶ Highly modular design
- ▶ Highly customizable/scalable
- ▶ Completely decentral organization
  - Real-time connection setup/ release on demand
    - Reserve Virtual Channels
    - Feedback
  - Broadcast/Multicast service
    - Robust
    - Efficient resource utilization
  - Adaptive routing
    - Local decisions only
    - Shortest path if possible
- ▶ Efficient integration
  - All traffic shares common resources



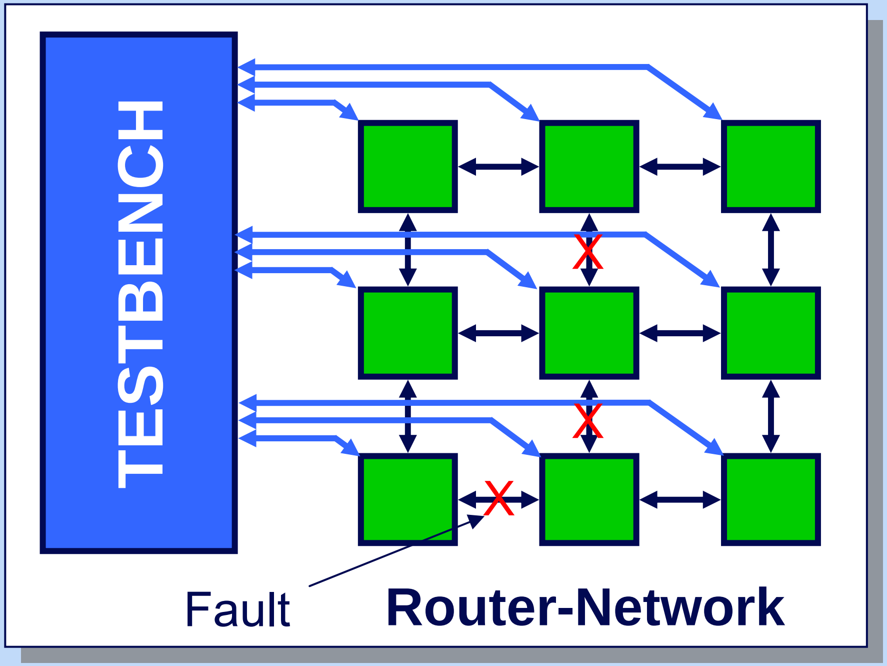
Standard Cell Synthesis (Basic Router)

- 0,13µ -Technology (TSMC130)
- 500 MHz
- 128 bit buffer per port (4VCs , 2 flits buffer depth)
- 16 bit data link → 40 Gbit/s
- xy-routing



Organic Processing Cells: Fault-Tolerant Routing - Verification and Simulation
(Prof. Becker)

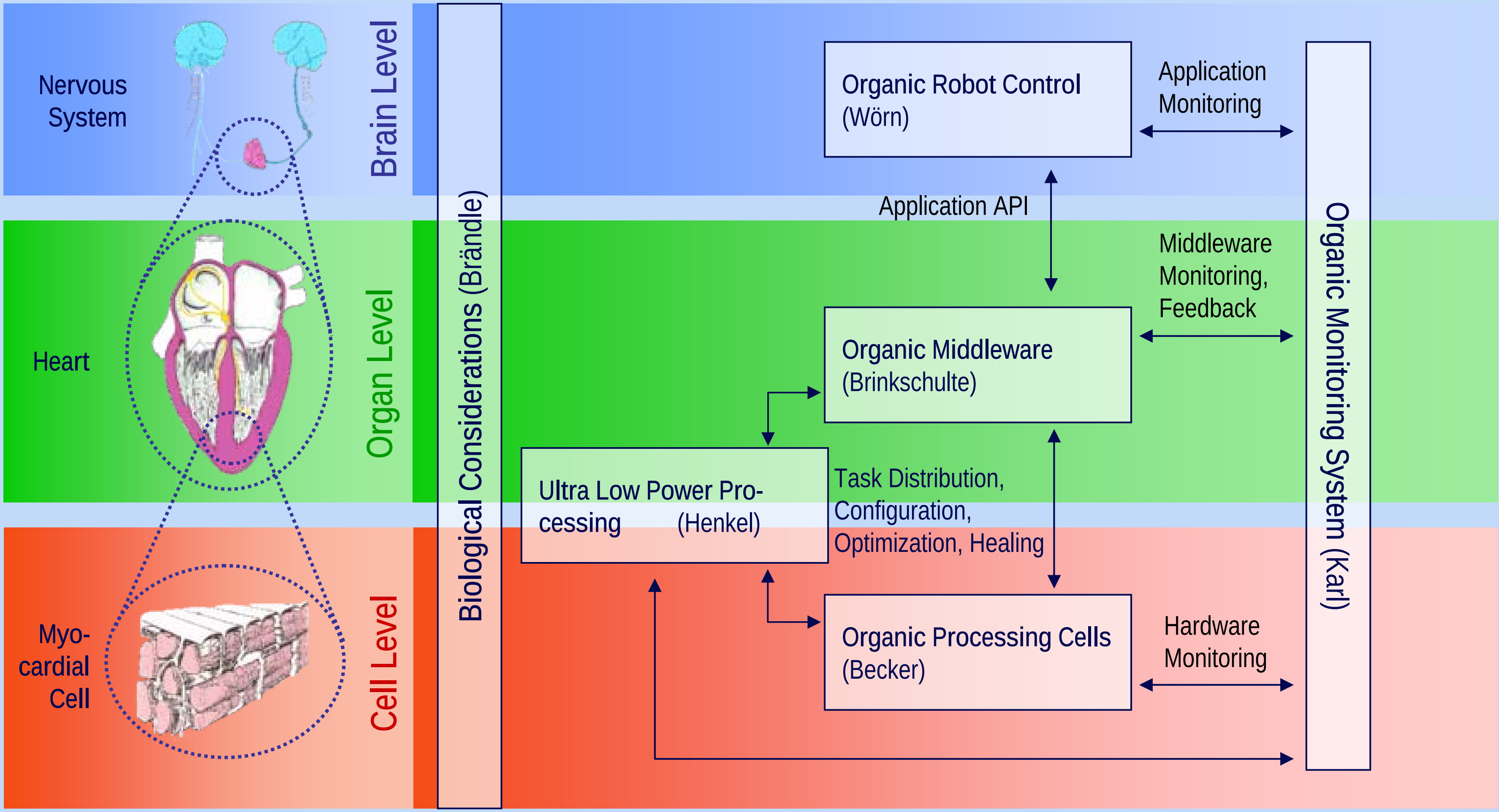
- ▶ Goal: avoid nodes with faulty links (permanent/transient)
  - Neighbor nodes send status
  - Node determines own status
  - Circumvention of defective links
- ▶ Automated Testbench
  - High-level stimuli definition
  - Packet-format
  - Interleaving
  - Insertion of link failures (dynamic)
  - Verification
    - Correct routing
    - Comparison of data sent / data received



- ▶ Ongoing work
  - Recover from interrupted transmission
    - Permanent
    - Transient
  - Avoidance of fault-induced deadlock

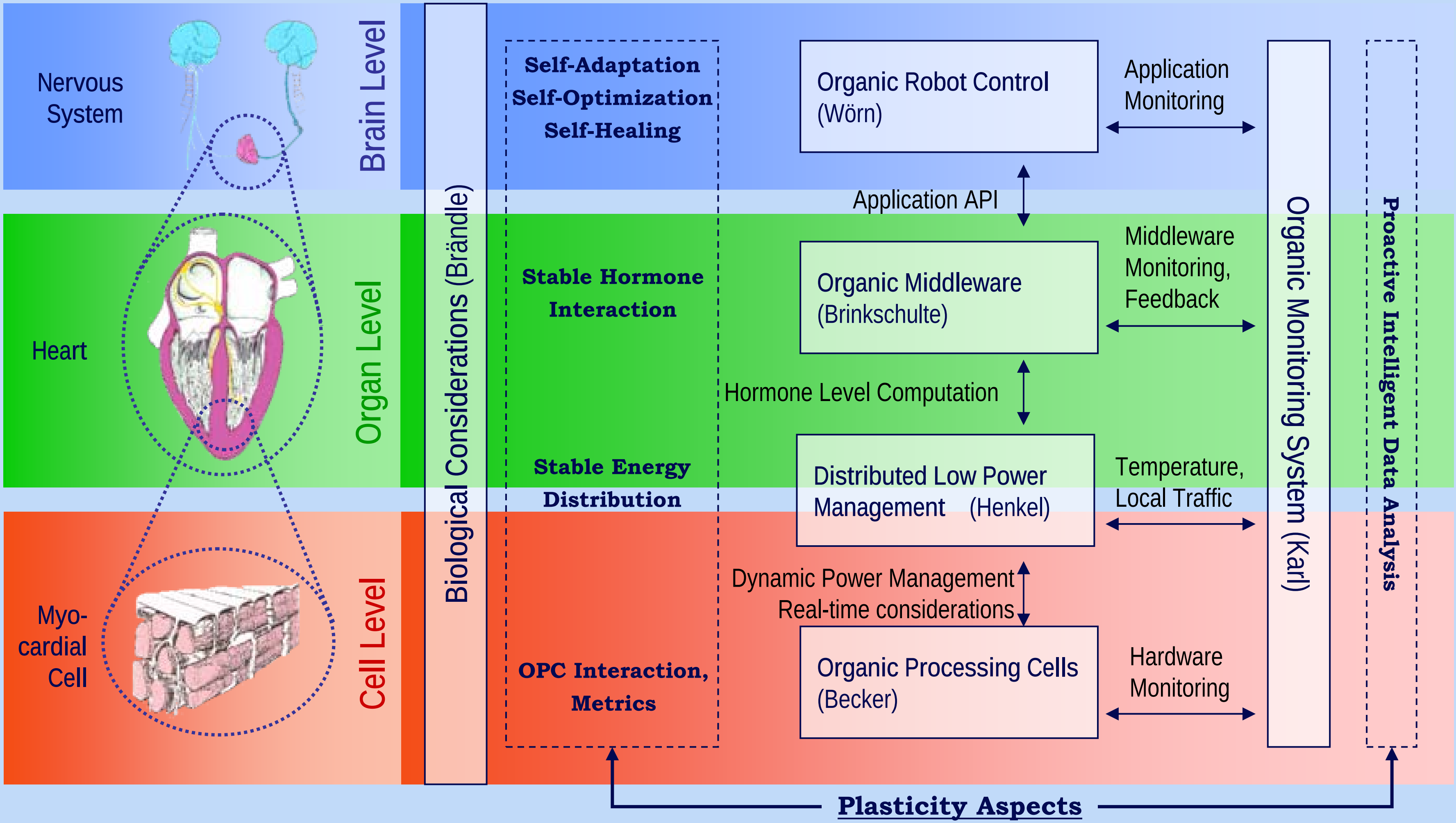
Refined Layer Model

DodOrg



Refined Layer Model

DodOrg



Nervous System

Brain Level

Heart

Organ Level

Myo-cardial Cell

Cell Level

Biological Considerations (Brändle)

Self-Adaptation
Self-Optimization
Self-Healing

Stable Hormone Interaction

Stable Energy Distribution

OPC Interaction, Metrics

Organic Robot Control (Wörn)

Application Monitoring

Application API

Organic Middleware (Brinkschulte)

Middleware Monitoring, Feedback

Hormone Level Computation

Distributed Low Power Management (Henkel)

Temperature, Local Traffic

Dynamic Power Management
Real-time considerations

Organic Processing Cells (Becker)

Hardware Monitoring

Organic Monitoring System (Karl)

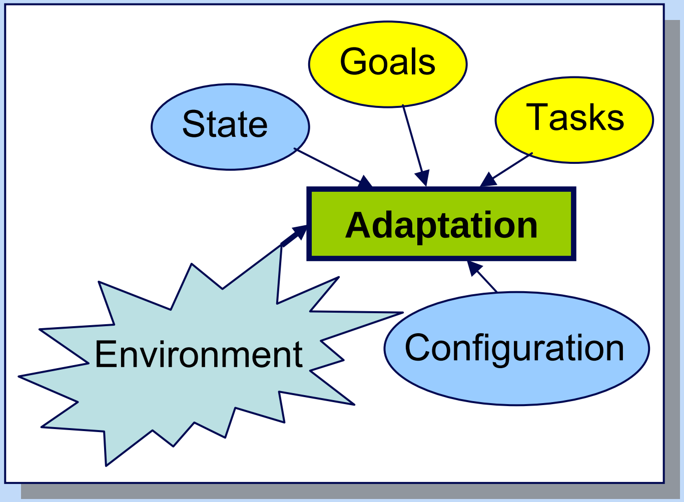
Proactive Intelligent Data Analysis

Plasticity Aspects

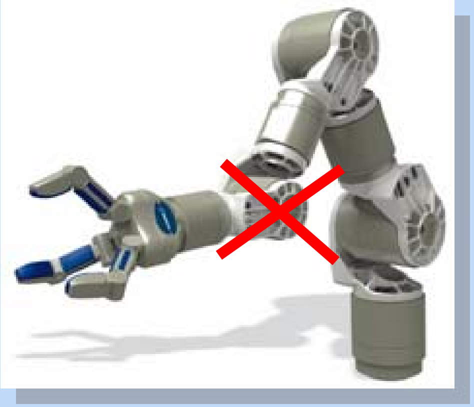
Focus lies on Self-Adaptation, Self-Healing, and Self-Optimization

►Self-Adaptation of the Organic Robot Control System to varying Goals, Processes, and Tasks

- Goals: safety, time, power consumption, accuracy, …
- Processes: robot-robot cooperation, human steering, …
- Tasks: welding, paint spraying, deburring, cutting, handling, assembly, polishing, …
- Algorithms changing at runtime to adapt the robot to both changes in the environment and changing goals
- Research approach:
    - Encapsulation of each technology task in one or several components
    - Knowledge about itself in each component
    - Smaller tasks learn to combine themselves to bigger tasks
    - Components connect themselves autonomously knowledge-based and using learning methods
    - Adaptation achieves stability and low power consumption
- Adaptation improves the **dynamics** and is a part of **plasticity** implementation for the ORCS



Organic Robot Control: Self-Healing and Self-Optimization (Prof. Wörn)

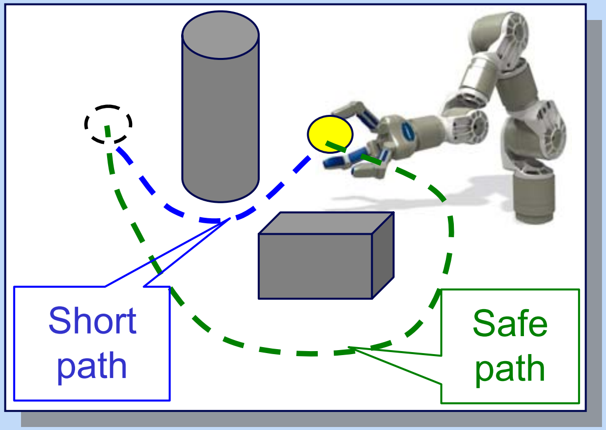


►Self-Healing Strategies for the Organic Robot Control System

- Component failures, the states, the load, etc. of the robot sensing by ORCS and Organic Monitoring System
- Automatic reconfiguration of the ORCS after detecting a joint or sensor failure to keep up operation
- Fast computing newly kinematics and dynamics models to compensate joint faults
- Adaptation algorithms improve new configuration additionally

►Self-Optimization of Robot Motions

- Algorithms to improve efficiency of the robot motion with respect to goal achievement, performance, or cost
- Finding the most suitable robot configuration respecting goals and physical constraints imposed by environment and manipulator design (including joint limits / faults)

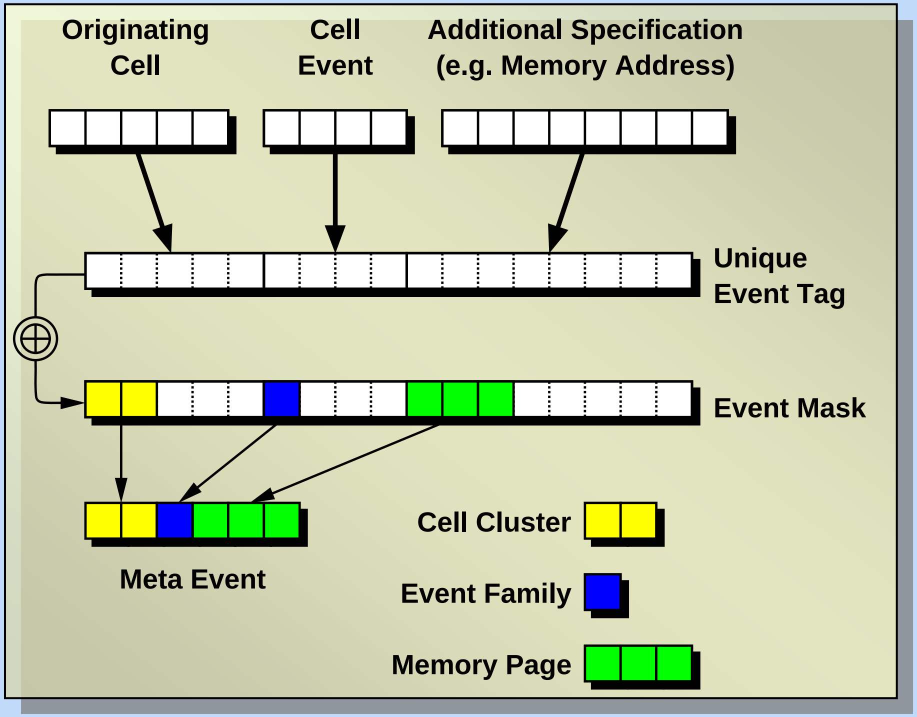


- Special algorithms to find, to refine, and to optimize paths through complex and dynamic environments respecting current goals and configuration in cooperation with Organic Monitoring System – it is also part of improvement of **dynamics** and **plasticity** implementation for the ORCS

Organic Monitoring: Events and Event Spaces (Prof. Karl)

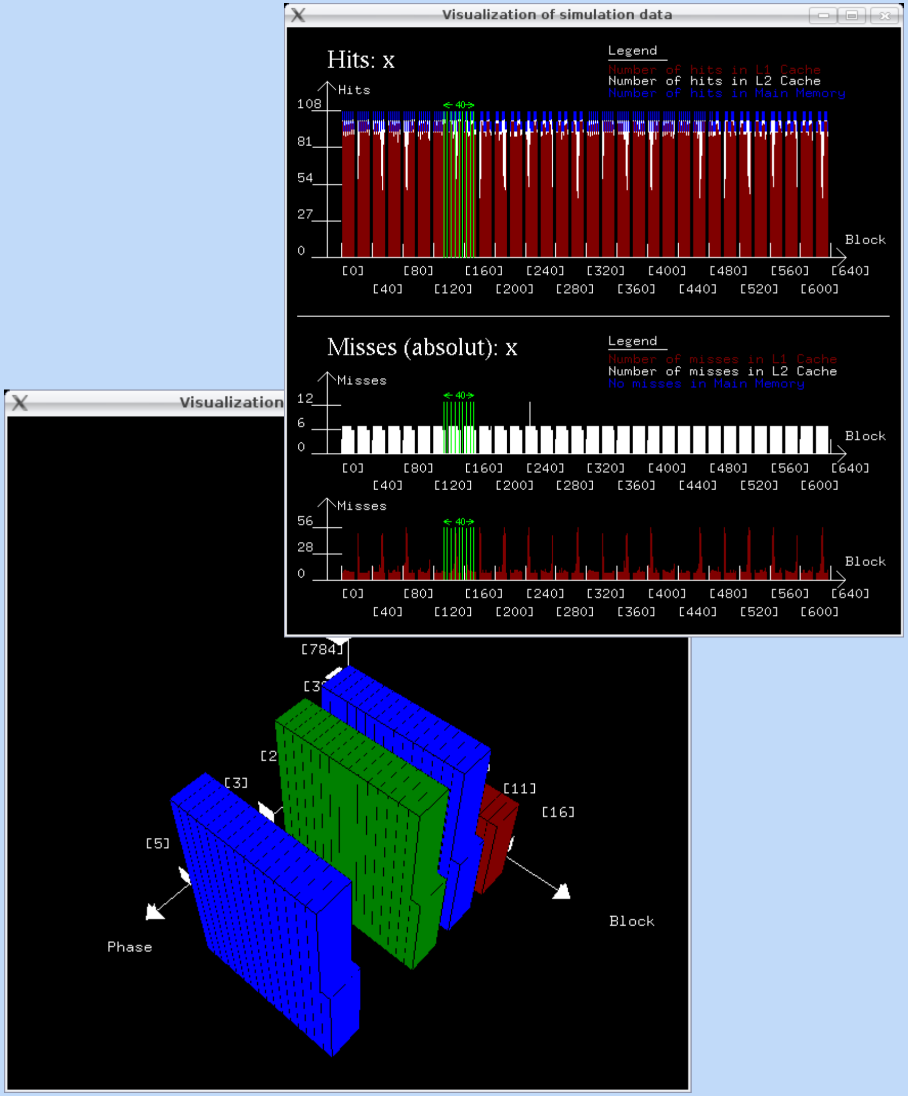
►Events and Event Spaces

- Conventional Monitoring and Evaluation relies on well-defined individual events and rules
- Hard to maintain in dynamic environments
    - Changing Event Types
    - Changing Event Quality
    - Adaption of Rule-set required
- Event Spaces required
    - Consider Event Spaces, not individual Event
    - Enables Classification and Scaling
- Concept of "Event Resolution"
    - Scale Resolution as required
    - Use entire Event or only Partial Information
- Matches Hormone Messenger Concept
    - Receiver decides if and how to react
- Interfaces well with Correlation Algorithms



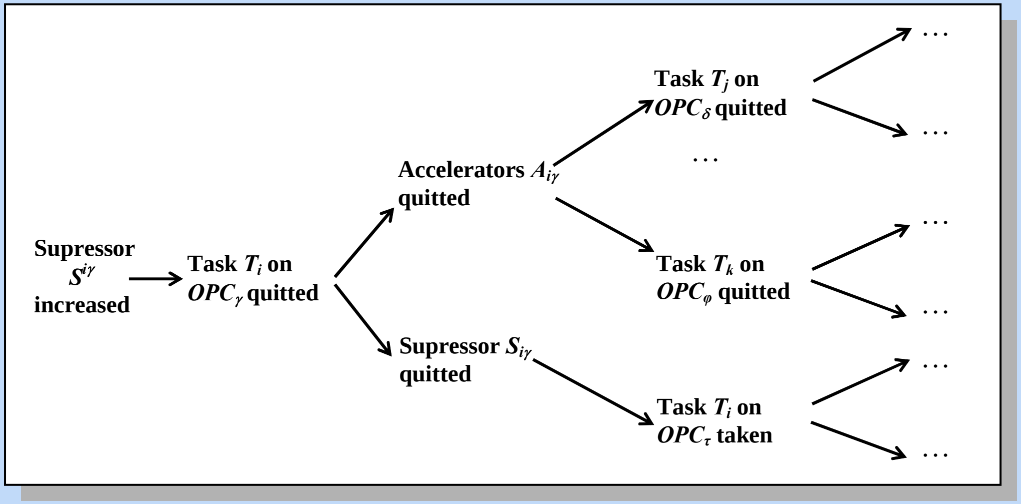
Organic Monitoring: Intelligent Data Analysis and Proactivity (Prof. Karl)

- Proactivity through Intelligent Data Analysis
  - Rule-sets hard to maintain in dynamically changing systems
    - Availability of Event Types & Quality
  - Rule-sets rely on Profiling
    - Determine Application Behavior (Phases, Hot-Spots) and define according Rule-set(s)
    - Not suitable for transient or Data-driven Events
    - Profiling not possible for dynamically changing Systems
  - Intelligent Data Analysis Techniques required
    - Improved Self-Awareness through Auto-Correlation and –Evaluation of Events
    - Introduce Proactivity: actively avoid potential conflicts by timely adaptation



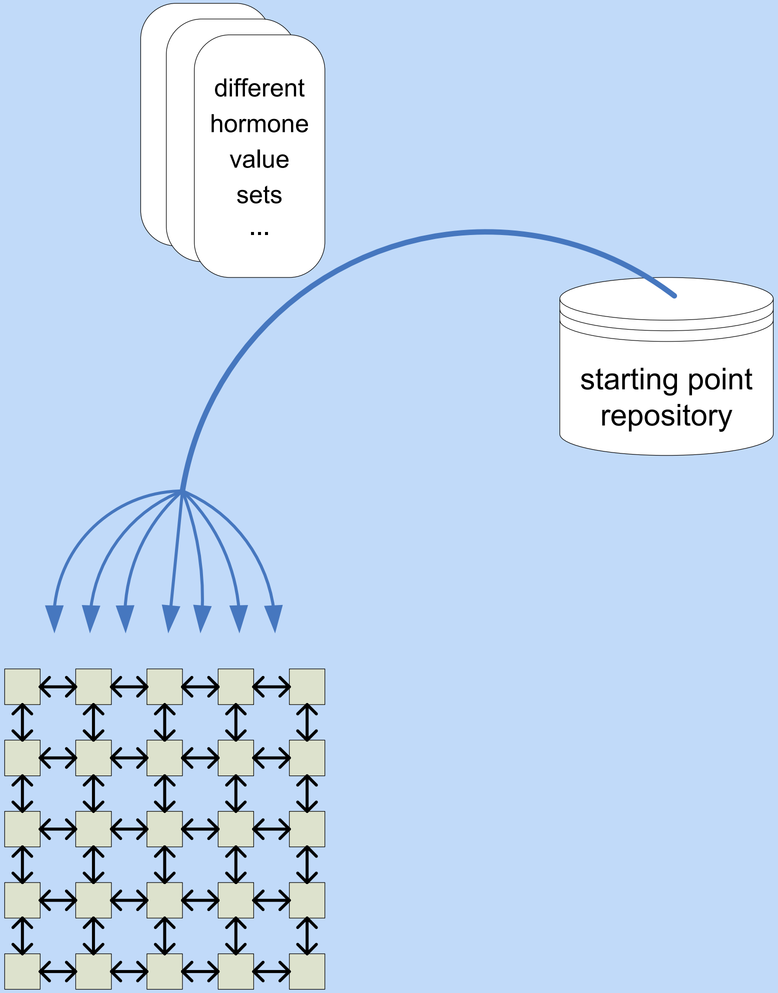
- Evaluation and Refinement of the Hormone Concept regarding:
  - Self-Optimization
  - Self-Adaptation
  - Self-Healing
- Investigation and Research of Dynamic Processes:
  - System reactions to changes
  - Possibilities to influence the reactions
  - Conditions for stable system behavior
  - Duration for system stabilization
- Examining Stability and Robustness of the Artificial Hormone System

	Plasticity		
	Self-Optimization	Self-Adaptation	Self-Healing
System operational	now and in near future	now	no longer
Available resources	stable	changing	degrading
Time of change	can be freely decided by the system	soon	now

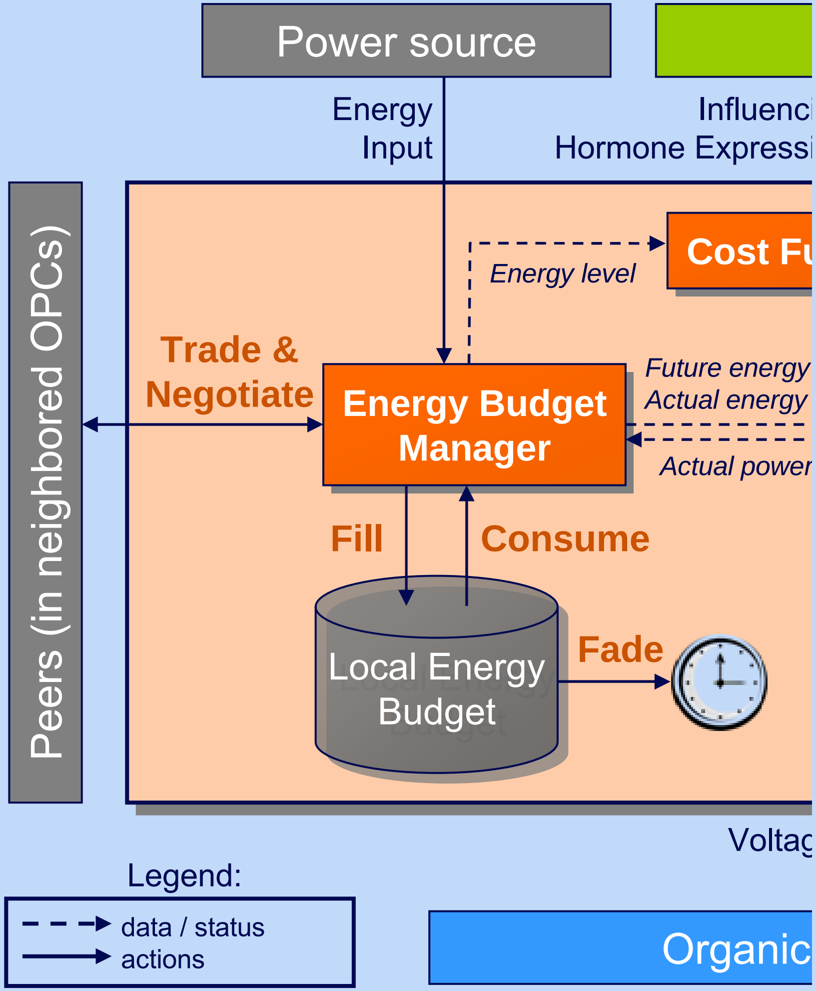


Next Phase of DodOrg

- Examination and Evaluation of different Self-Optimization Scenarios
  - Different Starting Points, e.g. from Scratch versus saved configurations from previous tests
  - Different methods like Case-based Reasoning or Rule-Induction Algorithms
- Quality Analysis of the Artificial Hormone System
  - Quantitative measures for the Quality of the Task Assignment
  - Merge different aspects:
    - Cell Suitability vs. Task Assignment
    - Communication Distance
    - Organ Formation



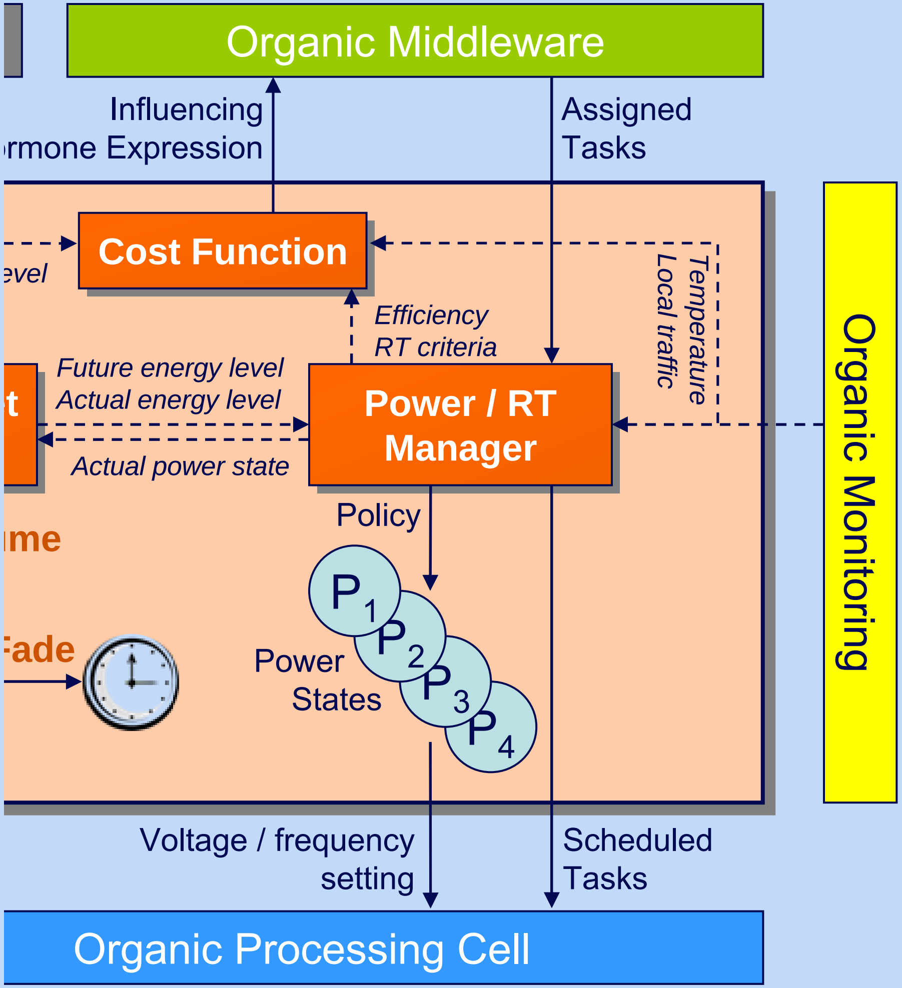
Distributed Low Power Management: Managing Local Energy-Distribution (Prof. Henkel)



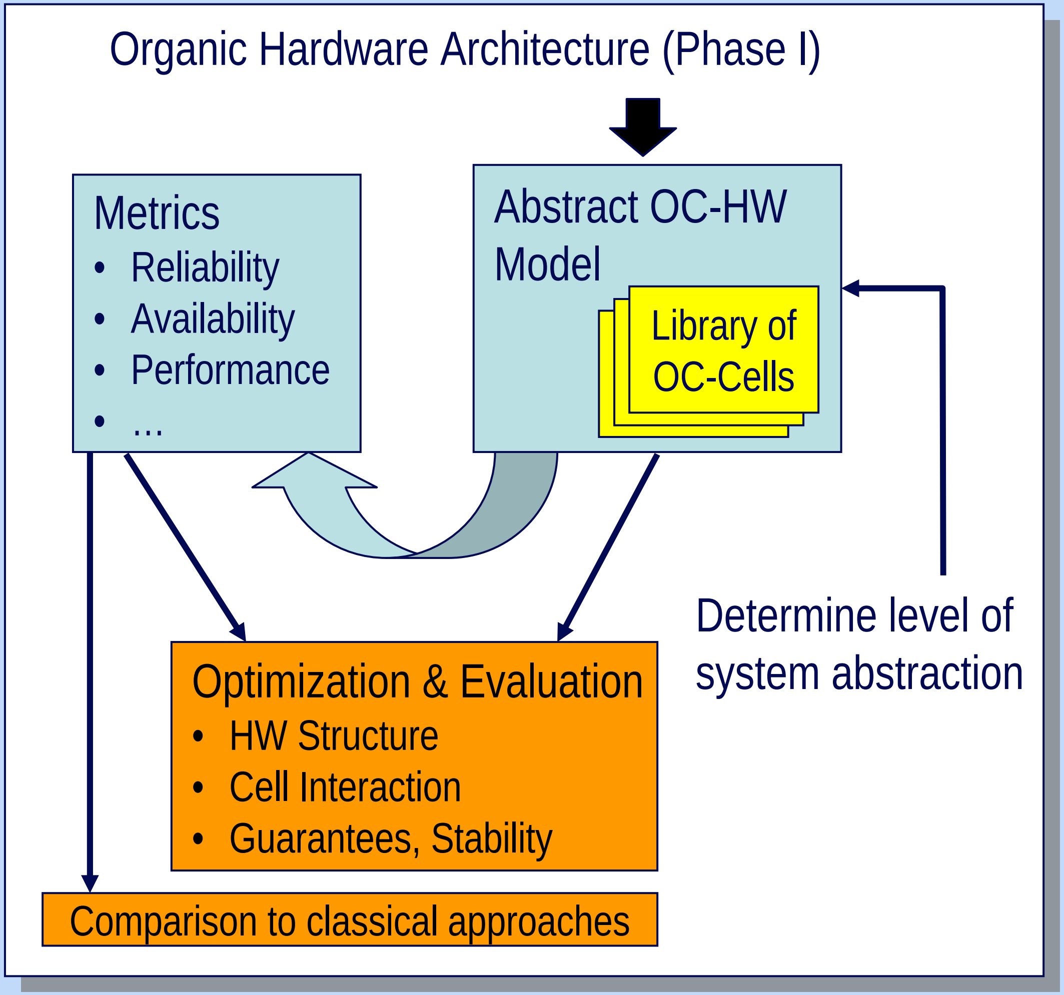
- Architecture
  - Global power source
    - Assigns energy budgets to OPCs
    - Depending on e.g. state of charge
  - Energy Budget Manager
    - Agent controlling local energy budget
    - Trades energy budget with peers
  - Local energy budget
    - Defines OPC energy consumption
- Research Goals
  - Convergent system behavior
  - Avoidance of local thermal hot-spots
  - Low energy consumption

Distributed Low Power Management: Interactions, Dynamics and Stability (Prof. Henkel)

- Architecture
  - Cost function
    - Used for computation of local eager values
  - Power / RT manager
    - Power states
    - Real-time management
    - Policy influenced by Energy Budget Manager
- Research goals
  - Finding an apt cost function
  - Dynamics / stability issues
  - Finding suitable power management policies



- ►Foundation laid by
  - DNA configuration control
  - Fault-tolerant/adaptive routing
  - Automated test system
  - Hardware prototype
- ►Challenges
  - Dynamics of cell interaction
  - Interference with Middleware/ Low-Power Management
- ►Research Goals
  - Metrics
  - Optimization
  - Plasticity



Organic Hardware Architecture (Phase I)

Metrics
- Reliability
- Availability
- Performance
- …

Abstract OC-HW Model

Library of OC-Cells

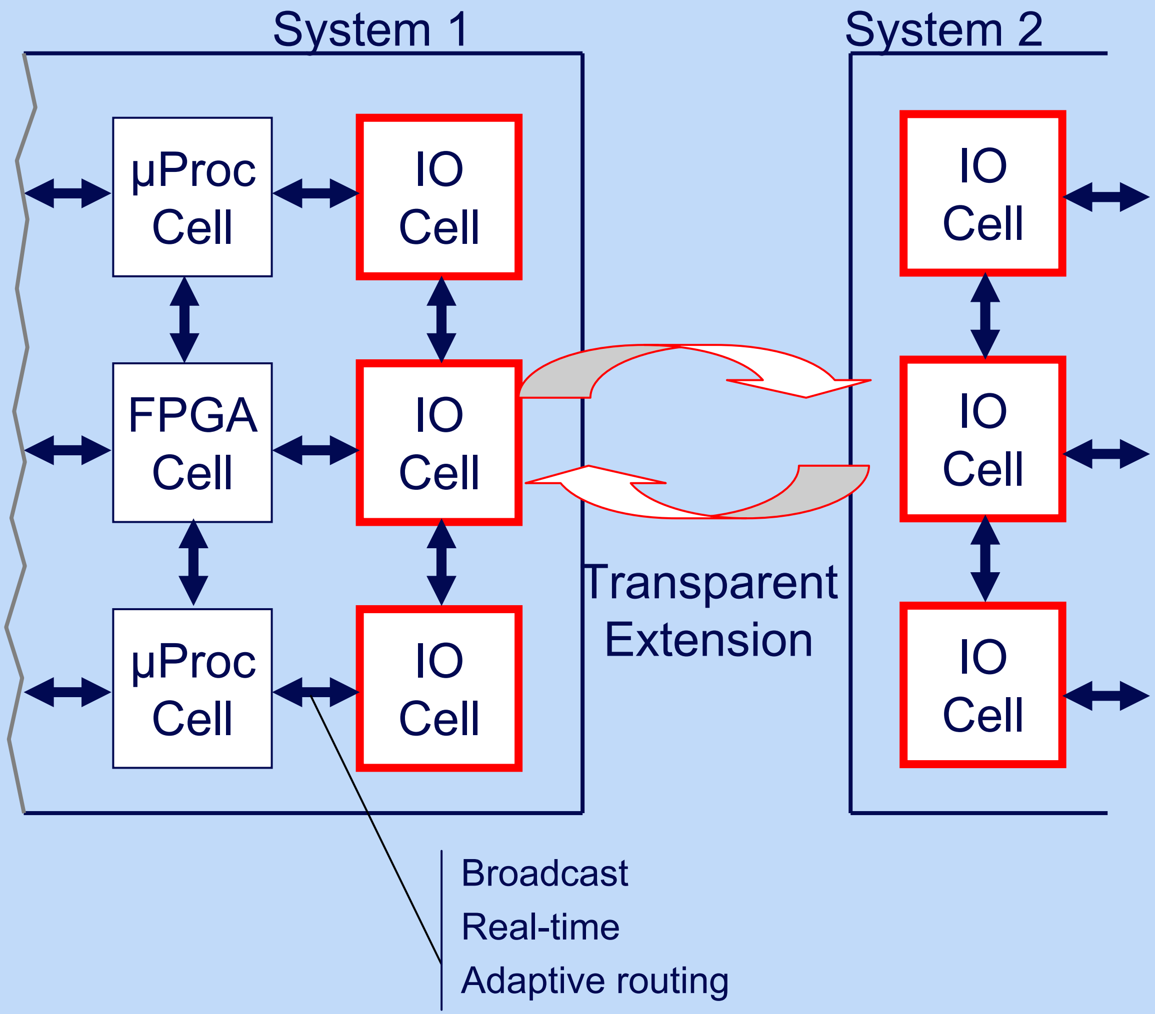
Determine level of system abstraction

Optimization & Evaluation
- HW Structure
- Cell Interaction
- Guarantees, Stability

Comparison to classical approaches

Organic Processing Cells: Transparent System Extension
(Prof. Becker)

- ► Challenges
  - Generalization of generic hardware model
  - Transfer of network services
  - Removing/adding individual OPCs
- ► Research Goals
  - Communication mechanisms of IO-cells
  - Adaptation of address space through use of DNA Configuration Management
  - Achieving true modularity



Current Phase of DodOrg

- ►Current Phase of the DodOrg Project:
  - Organic Robot Control
    - GUI-based generalized self-configuring control system for motion control of different robot types
    - Kinematic model automatically generated from a description of the robot’s mechanical structure
  - Monitoring Infrastructure
    - Interface definition and design space exploration
    - Software and hardware prototype
  - Middleware
    - Exploration of basic principles -- upper bound for self-configuration found
    - Hormone simulator
  - Ultra Low Power Processing
    - Categorized the basic principles for swapping-on-the-fly and conducted a hardware case study
    - Task Mapping / Scheduling Simulator
  - Organic Processing Cells
    - Exploration of the cells communication and configuration infrastructure
    - FPGA- Router Prototype

Conclusions (continued)

- ►Next Phase of the DodOrg Project
  - Organic Robot Control
    - Self-Adaptation, Self-Organization, and Self-Healing
    - Increased Fault-Tolerance and Self-Repairing
  - Organic Monitoring
    - Events and Event Spaces
    - Proactivity through Intelligent Data Analysis
  - Organic Middleware
    - Plasticity, Dynamics, and Stability
    - Refinement and Analysis of the Hormone Concept
  - Distributed Low Power Processing
    - Distributed Energy Brokering
    - Cost Functions & Convergence
  - Organic Processing Cells
    - Real-time Guarantees
    - Scalability

Organic Robot Control: Own Publications resulting from SPP1183 (Prof. Wörn)

DodOrg

- M. Wenz and H. Wörn. Automatic Configuration of the Dynamic Model for Common Industrial Robots. In C. Hochberger and R. Liskowsky, editors, Proceedings of the GI Jahrestagung, GI– Edition Lecture Notes in Informatics (LNI), pages 137–144, Dresden, Germany, October 2006.
- M. Wenz and H. Wörn. Entwicklung einer selbstkonfigurierenden und selbstorganisierenden Bewegungssteuerung für Industrieroboter. In G. Brandenburg, W. Schumacher, R. D. Schraft, and K. Bender, editors, Tagungsband SPS/IPC/Drives 2006, pages 619–627, Nürnberg, Germany, November 2006.
- M. Wenz and H. Wörn. Rule–based Generation of Motion Control Software for General Serial–Link Robot Manipulators. In Proceedings of the 8th International Workshop on Computer Science and Information Technologies (CSIT), pages 16–21, Karlsruhe, Germany, September 2006.
- M. Wenz and H. Wörn. Automatische Konfiguration der Bewegungssteuerung von seriellen Kinematiken. In J. Gausemeier, F. Rammig, W. Schäfer, A. Trächtler, and J. Wallaschek, editors, Entwurf mechatronischer Systeme, volume 210 of HNI-Verlagsschriftenreihe, pages 57–68, Paderborn, Germany, March 2007.
- M. Wenz and H. Wörn. Computing Symbolic–Algebraic Models of Industrial Manipulator Arms in a Fully Automatic Way. In K. Schilling, editor, Proceedings of the 13th IASTED International Conference on Robotics and Applications (RA 2007), pages 334–339, Würzburg, Germany, August 2007. ACTA Press.
- M. Wenz and H. Wörn. Solving the Inverse Kinematics Problem Symbolically by Means of Knowledge–Based and Linear Algebra–Based Methods. In Proceedings of the 12th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2007), Patras, Greece, September 2007. accepted.

Organic Monitoring: Own Publications Resulting from SPP1183 (Prof. Karl)

- Hans-Peter Löb, Rainer Buchty, and Wolfgang Karl: A Network Agent for Diagnosis and Analysis of Real-time Ethernet Networks. In Seongsoo Hong and Wayne Wolf, editors, International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES 2006), pages 65-73. ACM Press, New York, October 2006. ISBN 1-59593-543-6
- Rainer Buchty and Wolfgang Karl. A Monitoring Infrastructure for the Digital on-demand Computing Organism (DodOrg). In Hermann de Meer and James P. G. Sterbenz, editors, Self-OrganisingSystems: First International Workshop (IWSOS2006), LNCS4124, page 258. Springer Verlag, Berlin-Heidelberg, September 2006. ISBN 3-540-37658-5
- Rainer Buchty, Jie Tao, and Wolfgang Karl: Automatic Data Locality Optimization through Self-Optimization. In Hermann de Meer and James P. G. Sterbenz, editors, Self-Organising Systems: First International Workshop (IWSOS2006), LNCS4124, pages 187-201. Springer Verlag, Berlin-Heidelberg, September 2006. ISBN 3-540-37658-5
- Rainer Buchty, Wolfgang Karl: A Monitoring Infrastructure for the Digital on-demand Computing Organism (DodOrg), ACACES 2006 Proceedings, pp. 41-44, Academia Press, Ghent, ISBN 90-382-0981-9, July 2006

Organic Middleware: Own Publications Resulting from SPP1183 (Prof. Brinkschulte)

- Uwe Brinkschulte, Alexander von Renteln. Reliablity of an Artificial Hormone System with. In Parallel and Distributed Computing and Systems, Cambridge, Massachusetts, USA, November 19 -21, 2007.
- Uwe Brinkschulte, Mathias Pacher, and Alexander von Renteln. An Artificial Hormone System for Self-Organizing Real-Time Task Allocation in Organic Middleware. In Organic Computing. Springer, 2007. to be published in 2007.
- Uwe Brinkschulte, Mathias Pacher, and Alexander von Renteln. Towards an Artificial Hormone System for Self-Organizing Real-Time Task Allocation. In 5th IFIP Workshop on Software Technologies for Future Embedded & Ubiquitous Systems (SEUS 2007), 2007.
- Uwe Brinkschulte, Mathias Pacher, and Alexander von Renteln. Selbst-organisierende Middleware für eingebettete Echtzeitsysteme. In doIT Forschungstag 2006 (dPunkt Verlag), Mannheim, Germany, 13. Juli 2006.

Organic Processing Cells: Own Publications Resulting from SPP1183 (Prof. Becker)

DodOrg

- Michael Hübner, Christian Schuck, Matthias Kühnle, and Jürgen Becker. New 2-Dimensional Partial Dynamic Reconfiguration Techniques for Real-Time Adaptive Microelectronic Circuits. In Jürgen Becker, Andreas Herkersdorf, Amar Mukherjee, and Asim Smailagic, editors, IEEE Computer Society Annual Symposium on VLSI (ISVLSI 2006. IEEE Computer Society, March 2006. ISBN 0-7695-2533-4.
- Katharina Paulsson, Michael Hübner, S. Bayar, and Jürgen Becker. Exploitation of Run-Time Partial Reconfiguration for Dynamic Power Management in Xilinx Spartan III-based Systems. In ReCoSoC'07 Proceedings, June 2007.
- Christian Schuck, Stefan Lamparth, and Jürgen Becker. artNoC - A Novel Multifunctional Router Architecture for Organic Computing. In 17th International Conference On Field Programmable Logic and Applications, August 2007. accepted.

Thank you
for your attention !