

SICS

Swedish Institute of Computer Science

What is SICS?

- **A non-profit research institute**
- **Founded in 1985 by the government and a number of swedish companies**
- **Legally a foundation**
- **Research staff: 85 (30 PhDs)**
- **Turn-over 1998: 71 MSEK**

SICS mission

SICS mission is to contribute to the competitive strength of Swedish industry by conducting advanced and focused research in strategic areas of computer science, and actively promoting industrial use of new research ideas and results in industry and society at large. SICS works in a close collaboration with industry and the national and international research community.

Main sponsors

- **NUTEK, Swedish National Board for Industrial and Technical Development**
- **Telia**
- **L M Ericsson**
- **CelsiusTech Systems**
- **FMV, Defence Materiel Administration**
- **SJ, Swedish State Railways**
- **IBM**
- **Hewlett-Packard**
- **Sun Microsystems**

SICS funding

Basic research program funded by the main sponsors (30-35% of turn-over)

The rest (70-65%) consists of contracted projects:

- European Union projects
- Real World Computing program (Japanese Govt.)
- NUTEK and other national R&D-funding programs
- Industry funded projects

SICS memberships

- IRIS – Industrial Research Institutes in Sweden
- ERCIM – European Research Consortium for Informatics and Mathematics
- RWCP – Japanese Real World Computing Program
- W3C – National office of the World Wide Web consortium

SICS Research Partners (SRP)

A framework for collaboration with small and medium sized high-tech enterprises

- Acacia
- Dynarc
- G.R.A.F.
- Industrilogik L4i
- Medical Link
- Pharmasoft
- Pipebeach
- Prover Technology
- Scandface IT
- Tacton Systems

SICS results

- ~70 refereed publications in international journals and conferences per year
- 2–4 research degrees (PhD and licentiate) per year
- 1–3 persons move to academia for tenured positions (professor or lecturer) per year
- on average 10 people per year move to industry
- transfer of SICS projects and patents to industry, including spin-off companies
- distribution of software, examples: DIVE, SICStus Prolog, SimICS, Mozart

SICS research labs

CNA	Computer and Network Architectures
ISL	Intelligent Systems
NNRC	Neural Networks and Real-Time Computing
FDT	Formal Design Techniques
ICE	Interactive Collaborative Environments
HUMLE	Human-Machine Interaction and Language Engineering

ISL – Intelligent Systems Laboratory

- **Agent-Based Systems:** Automation of our participation in different forms of electronic markets
- **Programming Technology for Networked and Mobile Computing:** the OZ programming language and the distributed software platform Mozart
- **Decision Support for Planning and Scheduling**
- **Combinatorial Problem Solving and Prolog Technology:** finite domain constraint solving algorithms and propagation techniques, maintaining and developing the SICStus and Quintus Prolog systems

CNA – Computer and Network Architectures

We are carrying out research to enhance the computer and network platform for future distributed applications

- **Protocols and Networks:** design and implementation of protocols, network & OS support for audio & video
- **Mobile Computing and Communication:** network, OS and protocol architectures for mobility, ad-hoc networks
- **Scalable Distributed Applications:** network and OS support for large scale applications, serverless architectures, use of multicast, communication security
- **Computer Architecture Simulation:** timing analysis and dependability analysis using system level simulator

NNRC – Neural Networks and Real-time Computation laboratory

- **Stochastic Pattern Computing:** engineering theory of the brain that can serve as a foundation for sound engineering practise
- **Programming Intelligent Real-time Autonomous Interacting Agents:** design and control of robots with many degrees of freedom, such as snake robots
- **Learning and Adaptive Systems:** automatic extraction of interesting dependencies and relations from large volumes of data
- **Data security:** encryption algorithms, protocols and formal analysis

FDT – Formal Design Techniques

- Formal logic and mathematics to model and reason about computer systems
- Using state transition systems, process algebras and temporal logic
- Validation and verification of Erlang programs
- Java security using proof-carrying code

ICE – Interactive Collaborative Environments

Development of future computational environments that will support and promote collaborative environments.

- *Platforms for Collaborative Environments*: open platforms that will ease implementing collaborative applications
- *Applied Collaborative Environments*: electronic landscapes, distributed meeting rooms using VR techniques, distributed engineering, interactive television
- *Lucid Augmented Collaborative Experience*: development of new ways of experiencing future collaborative environments including new interaction techniques

HUMLE – Human-Machine Interaction and Language Engineering Laboratory

- *Information and Language Engineering*: spoken or written language, algorithms for maintaining, retrieving and extracting data from large sets of documents
- *Open and Adaptive Service Infrastructures*: open sets of users and services, means for users to access the services, user roles: end user, information broker, developer
- *Individuals in Space*: information overload, individual differences, social navigation