

The ALPS Project

Open Source Software for
Quantum Lattice Models

Simon Trebst, ETH Zürich

The ALPS collaboration

<http://alps.comp-phys.org>



The ALPS collaboration

Algorithms and **L**ibraries for **P**hysics **S**imulations

ETH Zürich, Switzerland

- Fabien Alet
- Prakash Dayal
- Mathias Körner
- Guido Schmid
- Simon Trebst
- Matthias Troyer
- Philipp Werner
- Stefan Wessel

Universität Marburg, Germany

- Reinhard Noack
- Salvatore Manmana

Universität Würzburg, Germany

- Martin Jöstingmeier

University of Tokyo, Japan

- Synge Todo

Université de Toulouse, France

- Andreas Läuchli

RWTH Aachen, Germany

- Ulrich Schollwöck
- Ian McCulloch

Universität Bonn, Germany

- Axel Grzesik

TU Graz, Austria

- Franz Michel
- Hans-Gerd Evertz

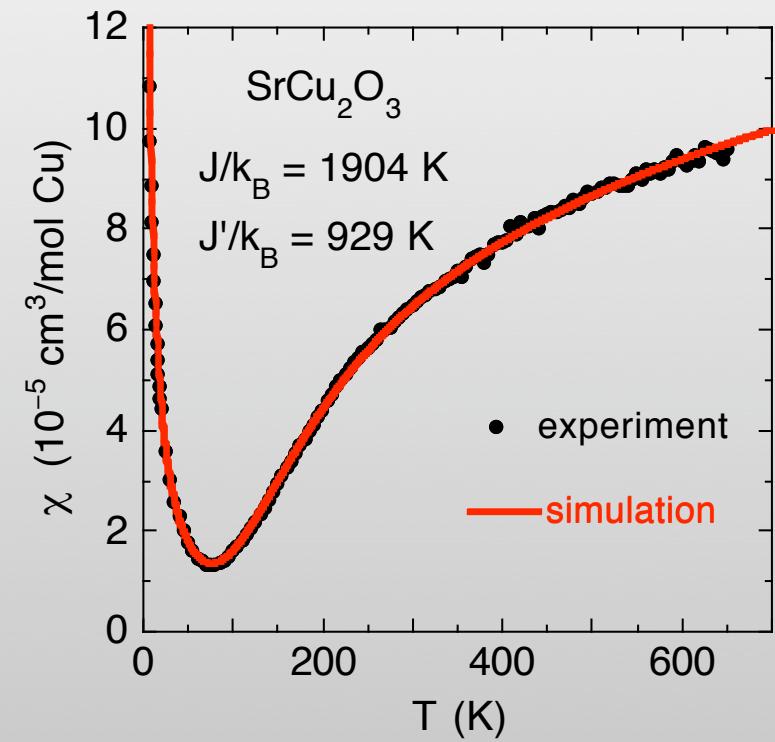
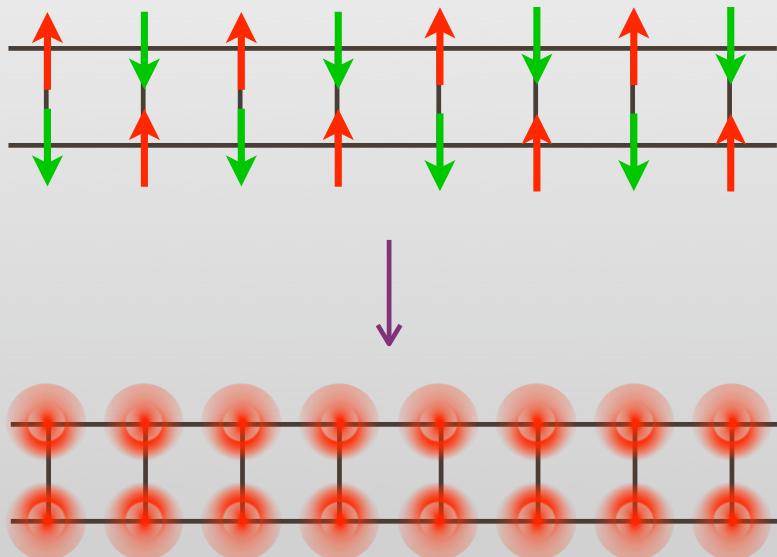
TU Braunschweig, Germany

- Andreas Honecker

The ALPS project

- open source libraries and simulation codes for strongly correlated quantum systems
 - Quantum Monte Carlo
 - exact diagonalization
 - DMRG
- Motivation
 - established algorithms
 - increased demand for reliable simulations from theorists and experimentalists

Strongly correlated systems

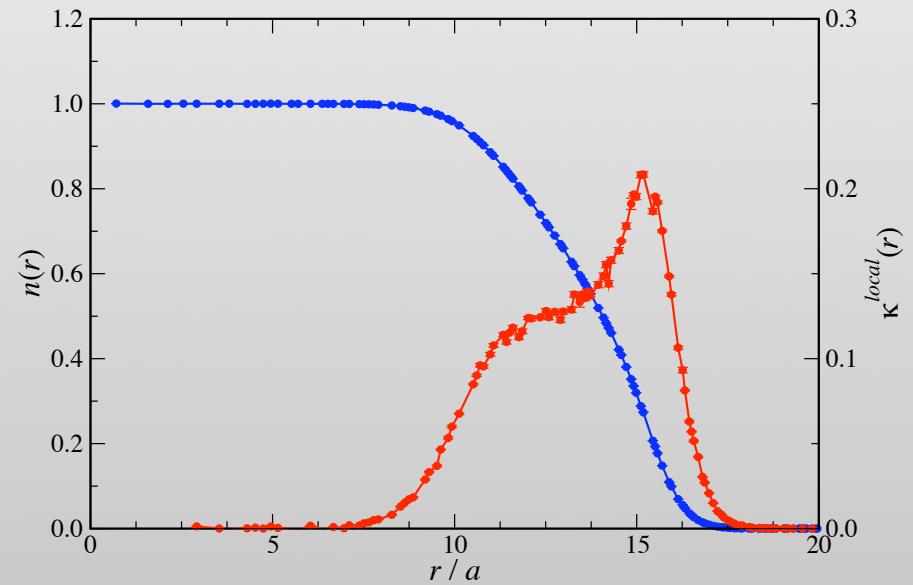
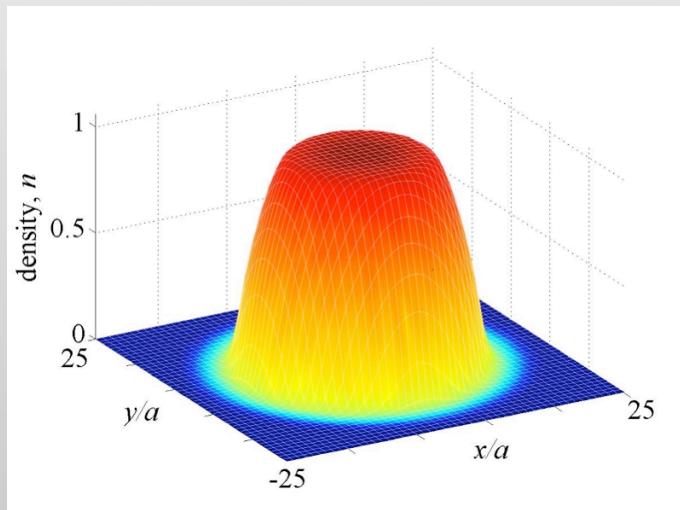


→ compare microscopic models to experiments

Strongly correlated systems

talk H28.004 by Stefan Wessel

- BEC of ultra-cold atoms in optical lattices



→ quickly test theoretical ideas

Simulation of quantum lattice models

- **The status quo**
 - individual codes
 - model-specific implementations
 - growing complexity of methods
- **ALPS**
 - community codes
 - generic implementations
 - common libraries simplify code development
 - common file formats

Modern technologies

- “Starting late” allows us to use
 - generic programming in C++
→ fast & flexible codes
 - standard C++ libraries
 - MPI and OpenMP for parallelization
 - XML / XSLT for file processing

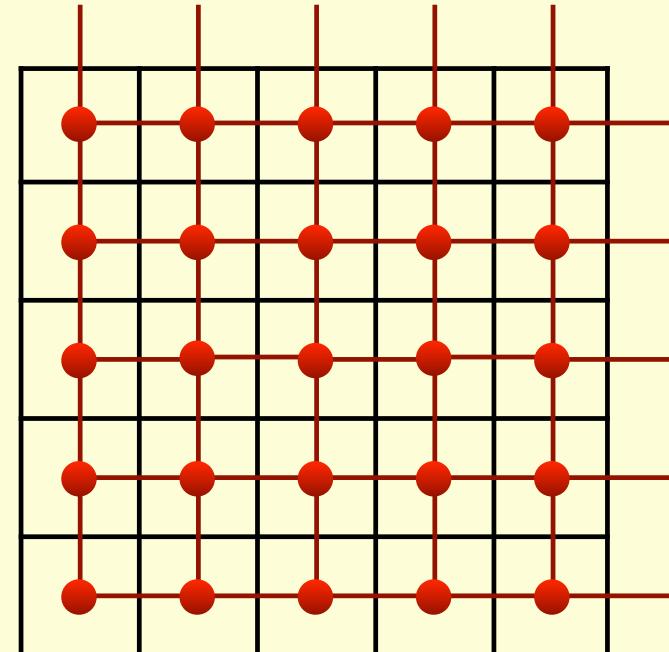
The ALPS sources

applications	QMC	ED	DMRG	...
domain specific libraries	lattice	model	observables	scheduler
numerics	random	ublas	IETL	
general C++	graph	serialization	XML/XSLT	
C / Fortran	BLAS	LAPACK	MPI	

The ALPS lattice library

A lattice

```
<LATICEGRAPH name = "square lattice">
  <FINITELATTICE>
    <LATICE dimension="2"/>
    <EXTENT dimension="1" size="L"/>
    <EXTENT dimension="2" size="L"/>
    <BOUNDARY type="periodic"/>
  </FINITELATTICE>
  <UNITCELL>
    <VERTEX/>
    <EDGE type="1">
      <SOURCE vertex="1" offset="0 0"/>
      <TARGET vertex="1" offset="0 1"/>
    </EDGE>
    <EDGE type="2">
      <SOURCE vertex="1" offset="0 0"/>
      <TARGET vertex="1" offset="1 0"/>
    </EDGE>
  </UNITCELL>
</LATICEGRAPH>
```



The ALPS model library

A model

$$H_{XXZ} = \frac{J_{xz}}{2} \sum_{\langle i,j \rangle} (S_i^+ S_j^- + S_i^- S_j^+) + J_z \sum_{\langle i,j \rangle} S_i^z S_j^z - h \sum_i S_i^z$$

```
<BASIS>
  <SITEBASIS name="spin">
    <PARAMETER name="S" default="1/2"/>
    <QUANTUMNUMBER name="Sz" min="-S" max="S"/>
  </SITEBASIS>
</BASIS>

<OPERATOR name="Splus" matrixelement="sqrt(S*(S+1)-Sz*(Sz+1))">
  <CHANGE quantumnumber="Sz" change="1"/>
</OPERATOR>
<OPERATOR name="Sminus" matrixelement="sqrt(S*(S+1)-Sz*(Sz-1))">
  <CHANGE quantumnumber="Sz" change="-1"/>
</OPERATOR>
<OPERATOR name="Sz" matrixelement="Sz"/>

<HAMILTONIAN name="spin">
  <BASIS ref="spin"/>
  <SITETERM> -h*Sz </SITETERM>
  <BONDTERM source="i" target="j">
    Jxy/2*(Splus(i)*Sminus(j)+Sminus(i)*Splus(j))+ Jz*Sz(i)*Sz(j)
  </BONDTERM>
</HAMILTONIAN>
```

Simulations with ALPS

Lattice

```
<LATICEGRAPH name = "square lattice">
  <FINITELATTICE>
    <LATICE dimension="2"/>
    <EXTENT dimension="1" size="L"/>
    <EXTENT dimension="2" size="L"/>
    <BOUNDARY type="periodic"/>
  </FINITELATTICE>
  <UNITCELL>
    ...
  </UNITCELL>
</LATICEGRAPH>
```

Model

```
<BASIS>
  <SITEBASIS name="spin">
    <PARAMETER name="S" default="1/2"/>
    <QUANTUMNUMBER name="Sz" min="-S" max="S"/>
  </SITEBASIS>
</BASIS>

<HAMILTONIAN name="spin">
  <BASIS ref="spin"/>
  <SITETERM> -h*Sz </SITETERM>
  <ONDTERM source="i" target="j">
    Jxy/2*(Splus(i)*Sminus(j)+Sminus(i)*Splus(j))
    + Jz*Sz(i)*Sz(j)
  </ONDTERM>
</HAMILTONIAN>
```

Parameters

```
LATTICE = "square lattice"
L = 100

MODEL = "spin"
Jxy = 1
Jz = 1
h = 0

{ T = 0.1 }
{ T = 0.2 }
{ T = 0.5 }
{ T = 1.0 }
```

quantum system

Quantum Monte Carlo

Exact diagonalization

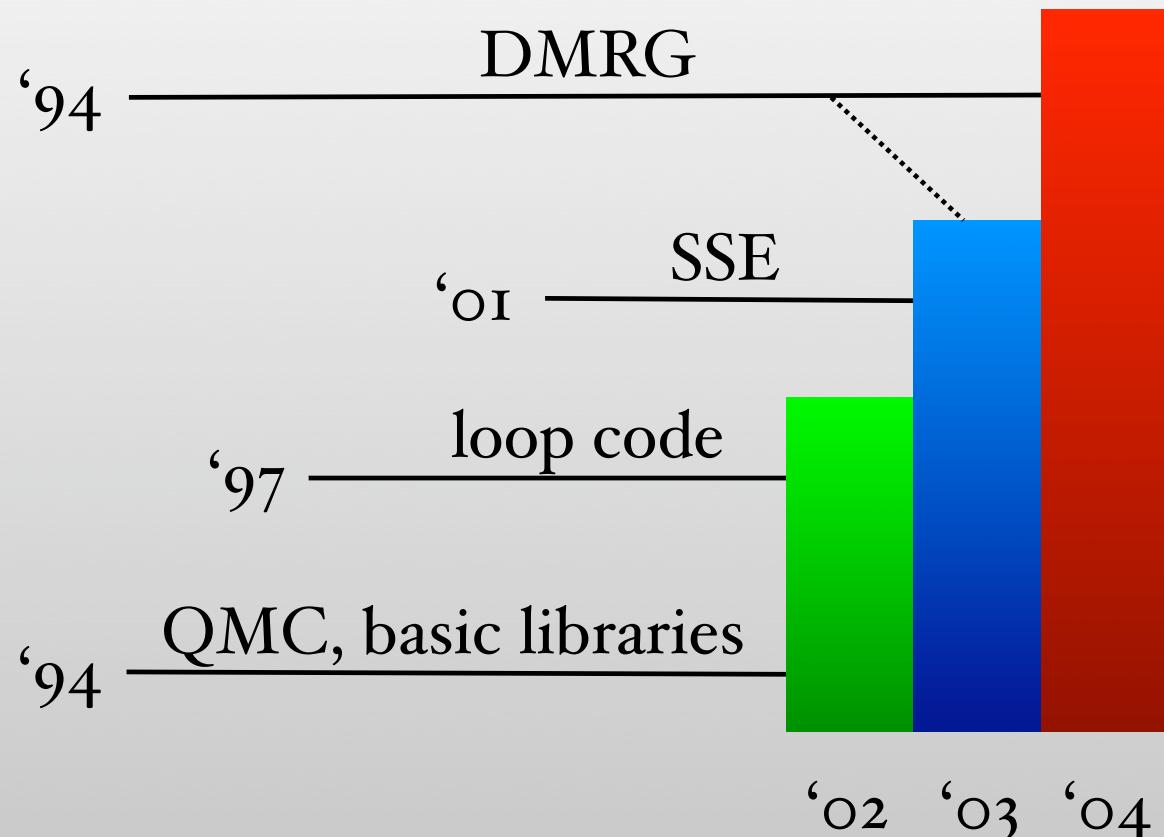
DMRG

Results

Open source development of ALPS

- Collaborative development
- Why open source codes?
- License discussion
- Funding situation

The History



the dark age

ALPS

Current applications

- **Quantum Monte Carlo**
 - stochastic series expansions (SSE), F. Alet, M. Troyer
 - loop code for spin systems, S. Todo
 - continuous time worm code, S. Trebst, M. Troyer
- **Exact diagonalization**
 - full and sparse, A. Honecker, A. Läuchli, M. Troyer
- **DMRG**
 - single particle, S. Manmana, R. Noack, U. Schollwöck
 - interacting particles, I. McCulloch

Collaborative development

Austria, France, Germany, Japan, Netherlands and Switzerland

- Some 15 developers, mostly PhDs.
- technical infrastructure
 - source code control system (CVS)
 - mailing lists
 - web page
- semi-annual workshops

Developer workshops

- Developers meet at semi-annual workshops
 - review of finished projects
 - discussion of current developments
 - road-map for new developments in the next 6 months
- Workshop are essential to coordinate and drive the development.



Open source codes

- **Why open source?**
 - community profits from improvements by individual developers
 - e.g. adding disorder to models
 - simplify code development
 - build on lattice, model, ... libraries
 - non-experts can use sophisticated algorithms
 - reproducibility of published scientific results

Open source codes

open sources \longleftrightarrow **scientific return**

- **Why should a researcher publish his codes?**
 - start new collaborations / networks
 - name recognition
 - citations
 - peer pressure to contribute
- **The “cite me” - license**

The “cite me” - license

- **Applications codes**
 - free for non-commercial applications
 - based on GNU public license
 - **citation requirements**
- *Modifications / improvements* of codes
 - should be integrated into ALPS
 - not obligatory to publish

The “cite me” - license

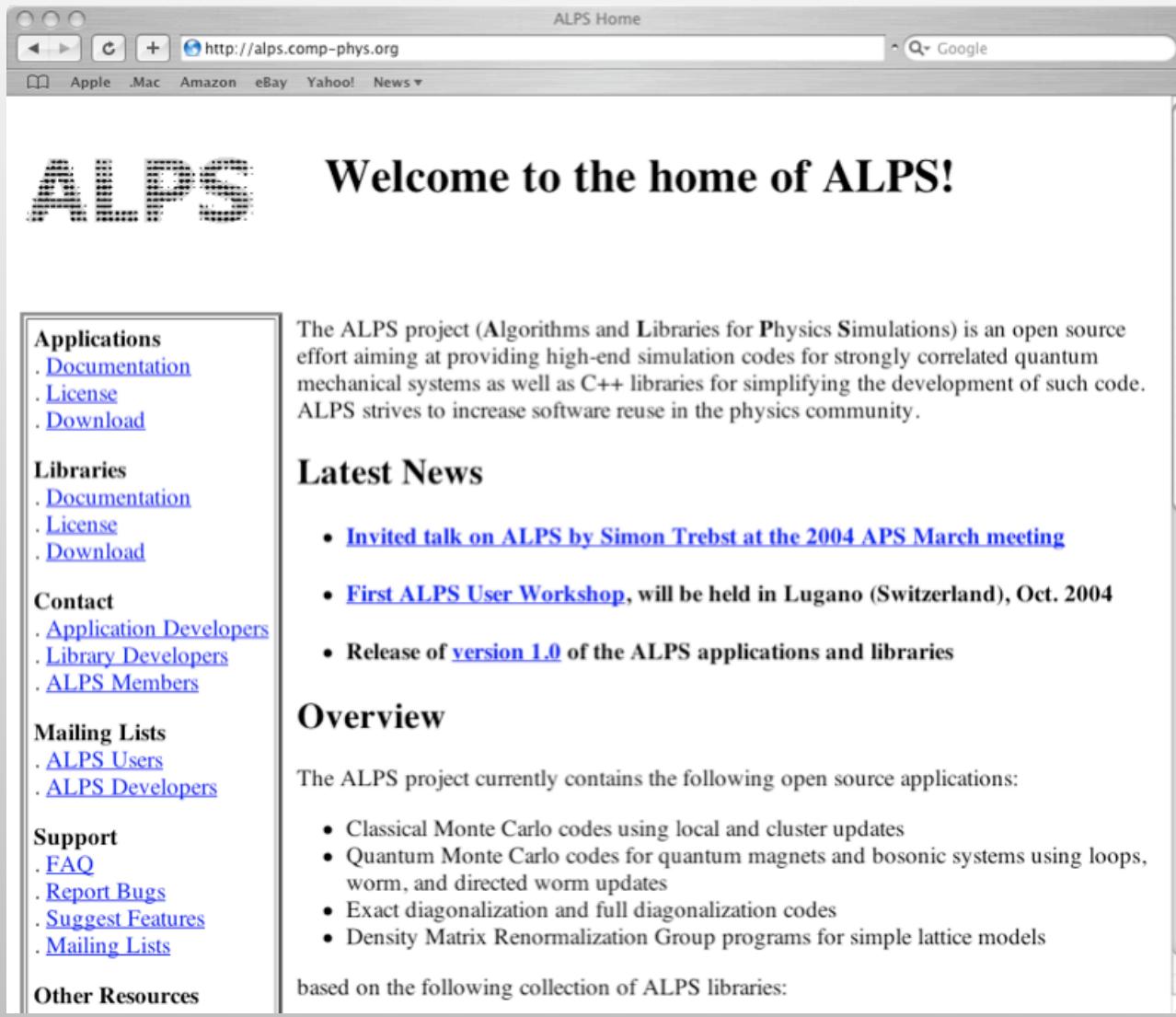
- **Applications codes**
 - free for non-commercial applications
 - based on GNU public license
 - **citation requirements**
- **Library codes**
 - less restrictive
 - partially available under a free license

User workshops

- **Oak Ridge '03**
 - part of workshop on Wang-Landau sampling
- **Lugano '04**
 - general lectures on numerical methods
 - hands-on tutorials of applications
 - direct feedback from users
 - workshops/tutorials set deadlines for developments

<http://alps.comp-phys.org>

Become a user yourself!



The screenshot shows a web browser window with the URL <http://alps.comp-phys.org> in the address bar. The page itself is titled "ALPS Home". On the left side, there is a sidebar with links for "Applications", "Libraries", "Contact", "Mailing Lists", "Support", and "Other Resources". The main content area features a large title "Welcome to the home of ALPS!" and a paragraph describing the project's purpose: "The ALPS project (Algorithms and Libraries for Physics Simulations) is an open source effort aiming at providing high-end simulation codes for strongly correlated quantum mechanical systems as well as C++ libraries for simplifying the development of such code. ALPS strives to increase software reuse in the physics community." Below this, there is a section titled "Latest News" with three bullet points: "Invited talk on ALPS by Simon Trebst at the 2004 APS March meeting", "First ALPS User Workshop, will be held in Lugano (Switzerland), Oct. 2004", and "Release of version 1.0 of the ALPS applications and libraries". At the bottom, there is a section titled "Overview" with a list of bullet points about the project's components and a note about the collection of ALPS libraries.

ALPS

Welcome to the home of ALPS!

Applications

- [. Documentation](#)
- [. License](#)
- [. Download](#)

Libraries

- [. Documentation](#)
- [. License](#)
- [. Download](#)

Contact

- [. Application Developers](#)
- [. Library Developers](#)
- [. ALPS Members](#)

Mailing Lists

- [. ALPS Users](#)
- [. ALPS Developers](#)

Support

- [. FAQ](#)
- [. Report Bugs](#)
- [. Suggest Features](#)
- [. Mailing Lists](#)

Other Resources

The ALPS project currently contains the following open source applications:

- Classical Monte Carlo codes using local and cluster updates
- Quantum Monte Carlo codes for quantum magnets and bosonic systems using loops, worm, and directed worm updates
- Exact diagonalization and full diagonalization codes
- Density Matrix Renormalization Group programs for simple lattice models

based on the following collection of ALPS libraries:

Funding situation

- Software development classified as **infrastructure project**
 - no funding by research grants
 - combination with research projects
 - no professional code developers

Funding situation

- **Funding sources for infrastructure**
 - computer centers,
Swiss National Supercomputer Center
ORNL
 - companies?
 - think big: group grant
can fund a software developer
 - open source development helpful



Experiences

- Advantages of open source development 
- improved code design
- more flexibility
- code review by fellow developers
- new collaborations, joint research projects 
- new contacts to experimental groups 
- more work 

ALPS workshop '04

Lugano, September 26 - October 1

Invited speakers

F. Alet	R. Noack
G. Brown	T. Schulthess
H.-G. Evertz	S. Todo
C. Kollath	S. Trebst
A. Läuchli	Shan-Ho Tsai
I. McCulloch	

Hands-on tutorials

Classical Monte Carlo
Spin Dynamics
Quantum Monte Carlo
Exact diagonalization
DMRG

