

HFM-ESF Trieste '07



# Introduction to ALPS

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

Andreas Läuchli  
*IRRMA, Lausanne*  
<[laeuchli@comp-phys.org](mailto:laeuchli@comp-phys.org)>

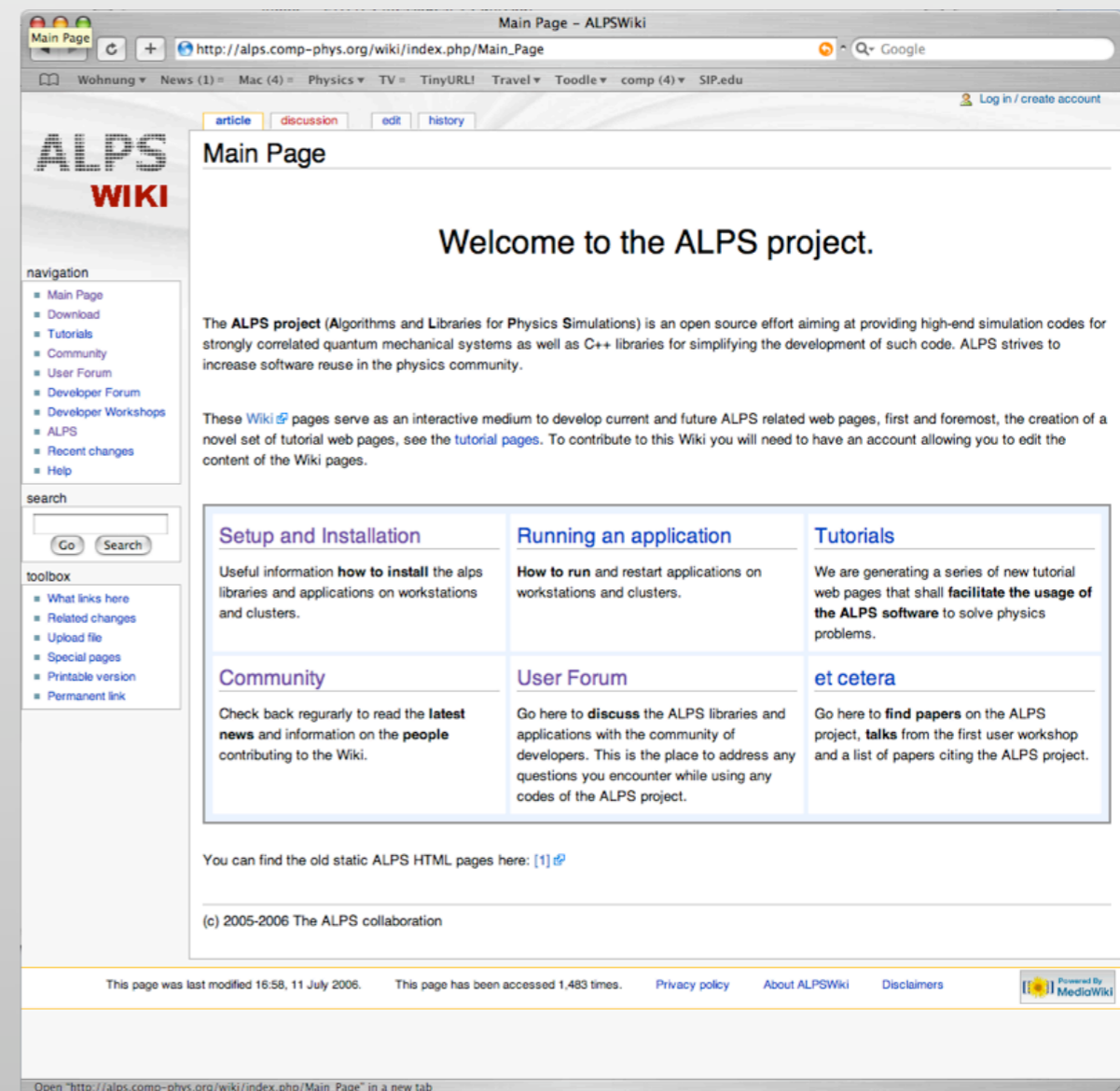
for the ALPS Collaboration

based on talks by M. Troyer & S. Todo

# The ALPS project

## Algorithms and Libraries for Physics Simulations

- **open source** data formats, libraries and simulation codes for quantum lattice models
- download codes from website **<http://alps.comp-phys.org>**



The screenshot shows the main page of the ALPS Wiki. The browser address bar displays [http://alps.comp-phys.org/wiki/index.php/Main\\_Page](http://alps.comp-phys.org/wiki/index.php/Main_Page). The page features a navigation menu on the left with links to Main Page, Download, Tutorials, Community, User Forum, Developer Forum, Developer Workshops, ALPS, Recent changes, and Help. A search box is also present. The main content area includes a welcome message, a description of the ALPS project, and a grid of links to various sections: Setup and Installation, Running an application, Tutorials, Community, User Forum, and et cetera. The footer contains copyright information and a footer bar with links to Privacy policy, About ALPSWiki, and Disclaimers.

Main Page - ALPSWiki

http://alps.comp-phys.org/wiki/index.php/Main\_Page

Wohnung News (1) Mac (4) Physics TV TinyURL Travel Toodle comp (4) SIP.edu Log in / create account

ALPS WIKI

Main Page

Welcome to the ALPS project.


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.

These [Wiki](#) pages serve as an interactive medium to develop current and future ALPS related web pages, first and foremost, the creation of a novel set of tutorial web pages, see the [tutorial pages](#). To contribute to this Wiki you will need to have an account allowing you to edit the content of the Wiki pages.

<b>Setup and Installation</b> Useful information <b>how to install</b> the alps libraries and applications on workstations and clusters.	<b>Running an application</b> <b>How to run</b> and restart applications on workstations and clusters.	<b>Tutorials</b> We are generating a series of new tutorial web pages that shall <b>facilitate the usage of the ALPS software</b> to solve physics problems.
<b>Community</b> Check back regularly to read the <b>latest news</b> and information on the <b>people</b> contributing to the Wiki.	<b>User Forum</b> Go here to <b>discuss</b> the ALPS libraries and applications with the community of developers. This is the place to address any questions you encounter while using any codes of the ALPS project.	<b>et cetera</b> Go here to <b>find papers</b> on the ALPS project, <b>talks</b> from the first user workshop and a list of papers citing the ALPS project.

You can find the old static ALPS HTML pages here: [\[1\]](#)

(c) 2005-2006 The ALPS collaboration

This page was last modified 16:58, 11 July 2006. This page has been accessed 1,483 times. [Privacy policy](#) [About ALPSWiki](#) [Disclaimers](#) 

Open "http://alps.comp-phys.org/wiki/index.php/Main\_Page" in a new tab

# The ALPS collaboration

## **ETH Zürich, Switzerland**

- Philippe Corboz
- Emanuel Gull
- Munehisa Matsumoto
- Lode Pollet
- Matthias Troyer

## **IRRMA / EPF Lausanne, Switzerland**

- Andreas Läuchli
- Salvatore Manmana

## **RWTH Aachen, Germany**

- Ulrich Schollwöck
- Ian McCulloch

## **Universität Marburg, Germany**

- Reinhard Noack

## **Universität Göttingen, Germany**

- Andreas Honecker
- Thomas Pruschke

## **Universität Stuttgart, Germany**

- Stefan Wessel

## **Université de Toulouse, France**

- Fabien Alet

## **TU Graz, Austria**

- Franz Michel

## **UC Santa Barbara, USA**

- Adrian Feiguin
- Simon Trebst

## **Columbia University, USA**

- Philipp Werner

## **Honk Kong University, China**

- Siegfried Gürtler

## **University of Tokyo, Japan**

- Ryo Igarashi
- Synge Todo

# Outline

- Overview of the ALPS project
- ALPS libraries and applications
- ALPS and XML
- Tutorial(s)

# The ALPS project

# The ALPS project

ALPS = Algorithms and Libraries for Physics Simulations



# The ALPS project

ALPS = **A**lgorithms and **L**ibraries for **P**hysics **S**imulations

- International collaboration for developing **open-source softwares** for simulation of **quantum lattice models**, such as quantum spin systems, electron systems, etc

# The ALPS project

ALPS = **A**lgorithms and **L**ibraries for **P**hysics **S**imulations

- International collaboration for developing **open-source softwares** for simulation of **quantum lattice models**, such as quantum spin systems, electron systems, etc
- **ALPS Libraries** = collection of **generic C++ libraries**



# The ALPS project

ALPS = **A**lgorithms and **L**ibraries for **P**hysics **S**imulations

- International collaboration for developing **open-source softwares** for simulation of **quantum lattice models**, such as quantum spin systems, electron systems, etc
- **ALPS Libraries** = collection of **generic C++ libraries**
- **ALPS Applications** = collection of application packages using **modern algorithms** such as QMC, DMRG, ED, etc

The status quo

# The status quo

- No “community codes” available
  - Individual code for each project
  - Model specific implementations

# The status quo

- No “community codes” available
  - Individual code for each project
  - Model specific implementations
- Growing complexity of methods
  - Weeks to months of software development

# The status quo

- No “community codes” available
  - Individual code for each project
  - Model specific implementations
- Growing complexity of methods
  - Weeks to months of software development
- Inputs/Outputs in non-portable formats

Target audience



# Target audience

- **Experimental physicists**
  - Use “canned codes” to **model materials**
  - **Determine microscopic parameters** by fitting experimental data to simulations

# Target audience

- **Experimental physicists**
  - Use “canned codes” to **model materials**
  - **Determine microscopic parameters** by fitting experimental data to simulations
- **Theoretical physicists**
  - **Quick check of theoretical ideas** using many modern algorithms
    - Monte Carlo, Diagonalization, DMRG, ...
    - Also useful for debugging
  - Libraries **simplify and accelerate code development**

# Current applications

- **Classical Monte Carlo**

- local and cluster updates for classical spin systems, *M. Troyer*

- **Quantum Monte Carlo**

- stochastic series expansions (SSE), *F. Alet, L. Pollet, M. Troyer*
- loop code for spin systems, *S. Todo*
- continuous time worm code, *S. Trebst, M. Troyer*
- extended ensemble simulations, *S. Wessel, N. Stoop*

- **Exact diagonalization**

- full and sparse, *A. Honecker, A. Läuchli, M. Troyer*

- **DMRG**

- single particle, *S. Manmana, R. Noack, U. Schollwöck*
- interacting particles, *A. Feiguin*

# ALPS - Platforms

- Unix Flavors:  
Linux (Fedora,...) , AIX, HP-UX, IRIX, ...
- Windows:  
Cygwin, CoLinux
- MacOS X
- High performance machines:  
Cray XT-3, IBM BlueGene/L

# The “cite me” license

- “ALPS Application License” for applications
  - free for non-commercial use
  - based on GNU public license
  - citation requirements
- “ALPS Library License” for library codes
  - less restrictive
  - partially available under a (BSD-like) free license as “ALPS Light Libraries”
- Read “LICENSE.txt” for more detail
- Most recent ALPS paper:

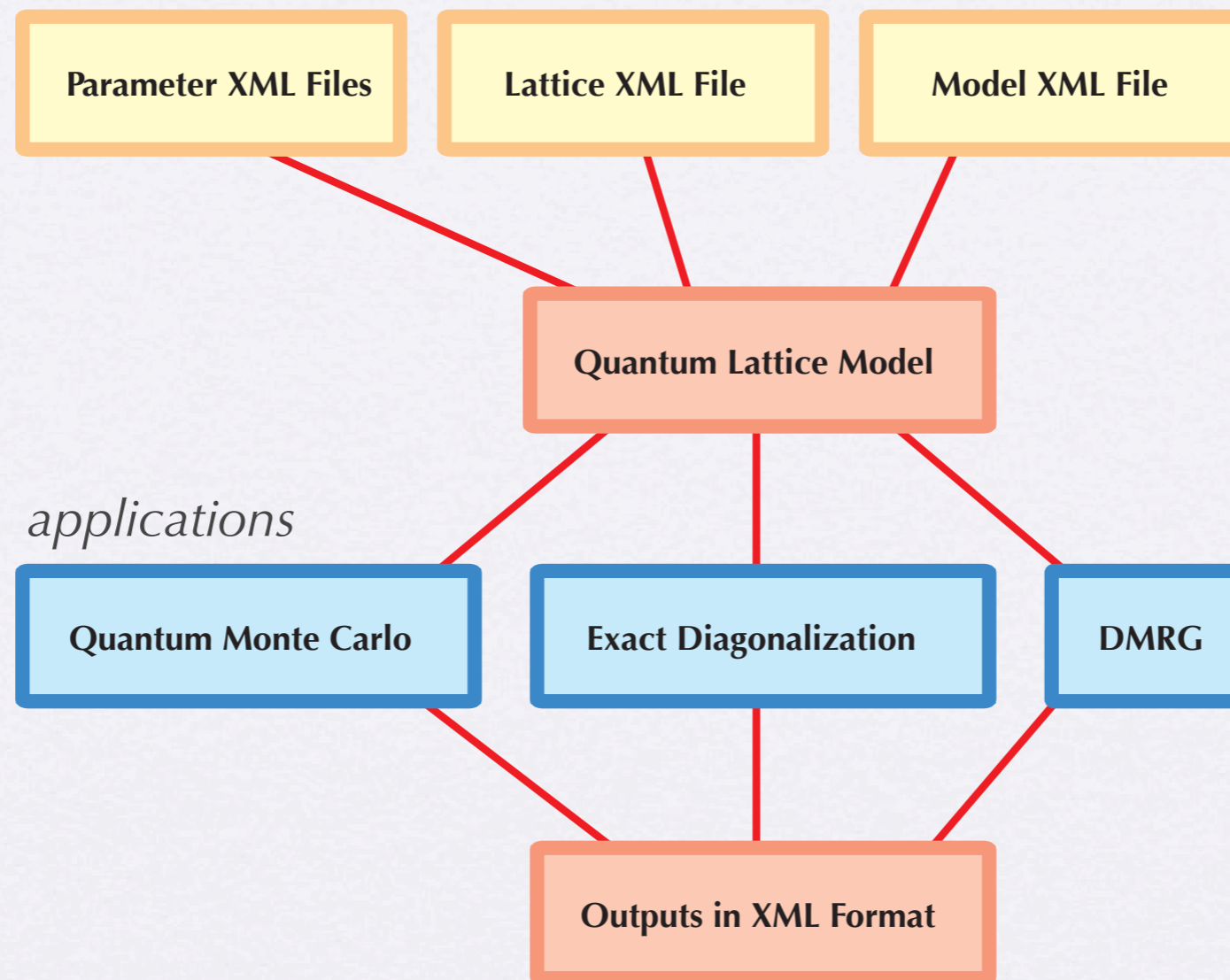
A.F. Albuquerque, et al, (for the ALPS collaboration),  
**The ALPS project release 1.3: open source software for strongly correlated systems,**  
J. Magn. Mag. Mat. **310**, 1187 (2007) ([JMMM](#)).

# XML and ALPS

- XML basics
- Input/Output of simulations in XML format
- Lattice/Model XML
- Running simulations in ALPS



# Simulations with ALPS



# XML basics

- XML = eXtensible Markup Language
- A flexible way to create **well-formed** and **self-describing** data
- We use “**tags**” to describe the **structure** of documents
- Example: HTML (XHTML)

```
<html>  
  <h1 align="center">Header</h1>  
  <p>A paragraph... And below it an image</p>  
    
</html>
```

*opening tag* points to `<html>`

*an attribute* points to `align="center"`

*contents* points to `A paragraph... And below it an image`

*empty element (no contents)* points to ``

*closing tag* points to `</html>`

# ALPS tool: parameter2xml

parameter sets in a compact form

```
LATTICE_LIBRARY="lattices.xml";  
LATTICE="chain lattidce";  
MODEL_LIBRARY="models.xml";  
MODEL="spin";  
THERMALIZATION=10000;  
SWEEPS=1000000;  
L=128;  
{ T=0.2; }  
{ T=0.4; }  
{ T=0.6; }  
{ T=0.8; }  
{ T=1.0; }
```

# ALPS tool: parameter2xml

parameter sets in a compact form

```
LATTICE_LIBRARY="lattices.xml";  
LATTICE="chain lattidce";  
MODEL_LIBRARY="models.xml";  
MODEL="spin";  
THERMALIZATION=10000;  
SWEEPS=1000000;  
L=128;           common parameters  
{ T=0.2; }  
{ T=0.4; }  
{ T=0.6; }  
{ T=0.8; }  
{ T=1.0; }
```

# ALPS tool: parameter2xml

parameter sets in a compact form

```
LATTICE_LIBRARY="lattices.xml";  
LATTICE="chain lattidce";  
MODEL_LIBRARY="models.xml";  
MODEL="spin";  
THERMALIZATION=10000;  
SWEEPS=1000000;  
L=128; common parameters
```

```
{ T=0.2; }  
{ T=0.4; }  
{ T=0.6; }  
{ T=0.8; }  
{ T=1.0; } task-specific parameters
```

# ALPS tool: parameter2xml

parameter sets in a compact form

```
LATTICE_LIBRARY="lattices.xml";  
LATTICE="chain lattidce";  
MODEL_LIBRARY="models.xml";  
MODEL="spin";  
THERMALIZATION=10000;  
SWEEPS=1000000;  
L=128;
```

*common parameters*

```
{ T=0.2; }  
{ T=0.4; }  
{ T=0.6; }  
{ T=0.8; }  
{ T=1.0; }
```

*task-specific parameters*

parameter2xml



myjob.in.xml

myjob.task1.in.xml

myjob.task2.in.xml

myjob.task3.in.xml

myjob.task4.in.xml

myjob.task5.in.xml



# Lattice XML (1/3)

definition of basis vectors

```
<LATTICES>
  <LATTICE name="square lattice" dimension="2">
    <PARAMETER name="a" default="1"/>
    <BASIS>
      <VECTOR>a 0</VECTOR>
      <VECTOR>0 a</VECTOR>
    </BASIS>
    <RECIPROCALBASIS>
      <VECTOR>2*pi/a 0</VECTOR>
      <VECTOR>0 2*pi/a</VECTOR>
    </RECIPROCALBASIS>
  </LATTICE>
  ...
```

# Lattice XML (2/3)

definition of the unit cell

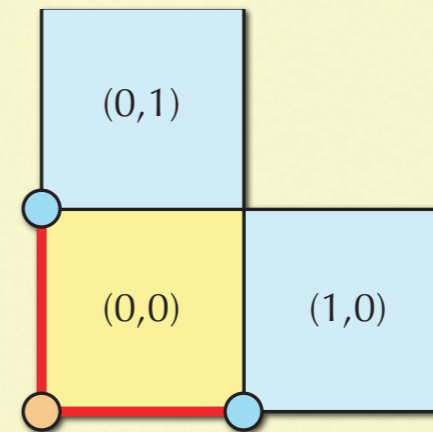
```
<LATTICES>
  <LATTICE name="square lattice" dimension="2">
    ...
  </LATTICE>
  <UNITCELL name="simple2d" dimension="2">
    <VERTEX/>
    <EDGE>
      <SOURCE vertex="1" offset="0 0"/>
      <TARGET vertex="1" offset="0 1"/>
    </EDGE>
    <EDGE>
      <SOURCE vertex="1" offset="0 0"/>
      <TARGET vertex="1" offset="1 0"/>
    </EDGE>
  </UNITCELL>
  ...

```

# Lattice XML (2/3)

definition of the unit cell

```
<LATTICES>
  <LATTICE name="square lattice" dimension="2">
    ...
  </LATTICE>
  <UNITCELL name="simple2d" dimension="2">
    <VERTEX/>
    <EDGE>
      <SOURCE vertex="1" offset="0 0"/>
      <TARGET vertex="1" offset="0 1"/>
    </EDGE>
    <EDGE>
      <SOURCE vertex="1" offset="0 0"/>
      <TARGET vertex="1" offset="1 0"/>
    </EDGE>
  </UNITCELL>
  ...
</LATTICES>
```



# Lattice XML (3/3)

specifying lattice size and boundary conditions

```
<LATTICES>
  <LATTICE name="square lattice" dimension="2">
    ...
  </LATTICE>
  <UNITCELL name="simple2d" dimension="2">
    ...
  </UNITCELL>
  <LATTICEGRAPH name="square lattice">
    <FINITELATTICE>
      <LATTICE ref="square lattice"/>
      <EXTENT dimension="1" size="L"/>
      <EXTENT dimension="2" size="L"/>
      <BOUNDARY type="periodic"/>
    </FINITELATTICE>
    <UNITCELL ref="simple2d"/>
  </LATTICEGRAPH>
</LATTICES>
```

# Model XML (1/2)

definition of local basis and operators

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

```



# Model XML (2/2)

definition of model Hamiltonian

```
<MODELS>
  <BASIS name="spin">
    ...
  </BASIS>
  <OPERATOR name="Sz" matrixelement="Sz"/> ...
  <HAMILTONIAN name="spin">
    <PARAMETER name="Jz" default="1"/>
    <PARAMETER name="Jxy" default="1"/>
    <PARAMETER name="h" default="0"/>
    <BASIS ref="spin"/>
    <SITETERM>-h*Sz</SITETERM>
    <BONDTERM source="i" target="j">
      Jz*Sz(i)*Sz(j) +
      Jxy/2*(Splus(i)*Sminus(j)+Sminus(i)*Splus(j))
    </BONDTERM>
  </HAMILTONIAN>
</MODEL>
```



# Model XML (2/2)

definition of model Hamiltonian

```
<MODELS>
  <BASIS name="spin" ...
    ...
  </BASIS>
  <OPERATOR name="Sz" matrixelement="Sz"/> ...
  <HAMILTONIAN name="spin">
    <PARAMETER name="Jz" default="1"/>
    <PARAMETER name="Jxy" default="1"/>
    <PARAMETER name="h" default="0"/>
    <BASIS ref="spin"/>
    <SITETERM>-h*Sz</SITETERM>
    <BONDTERM source="i" target="j">
      Jz*Sz(i)*Sz(j) +
      Jxy/2*(Splus(i)*Sminus(j)+Sminus(i)*Splus(j))
    </BONDTERM>
  </HAMILTONIAN>
</MODEL>
```

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

- More examples for lattice and model XMLs can be found at

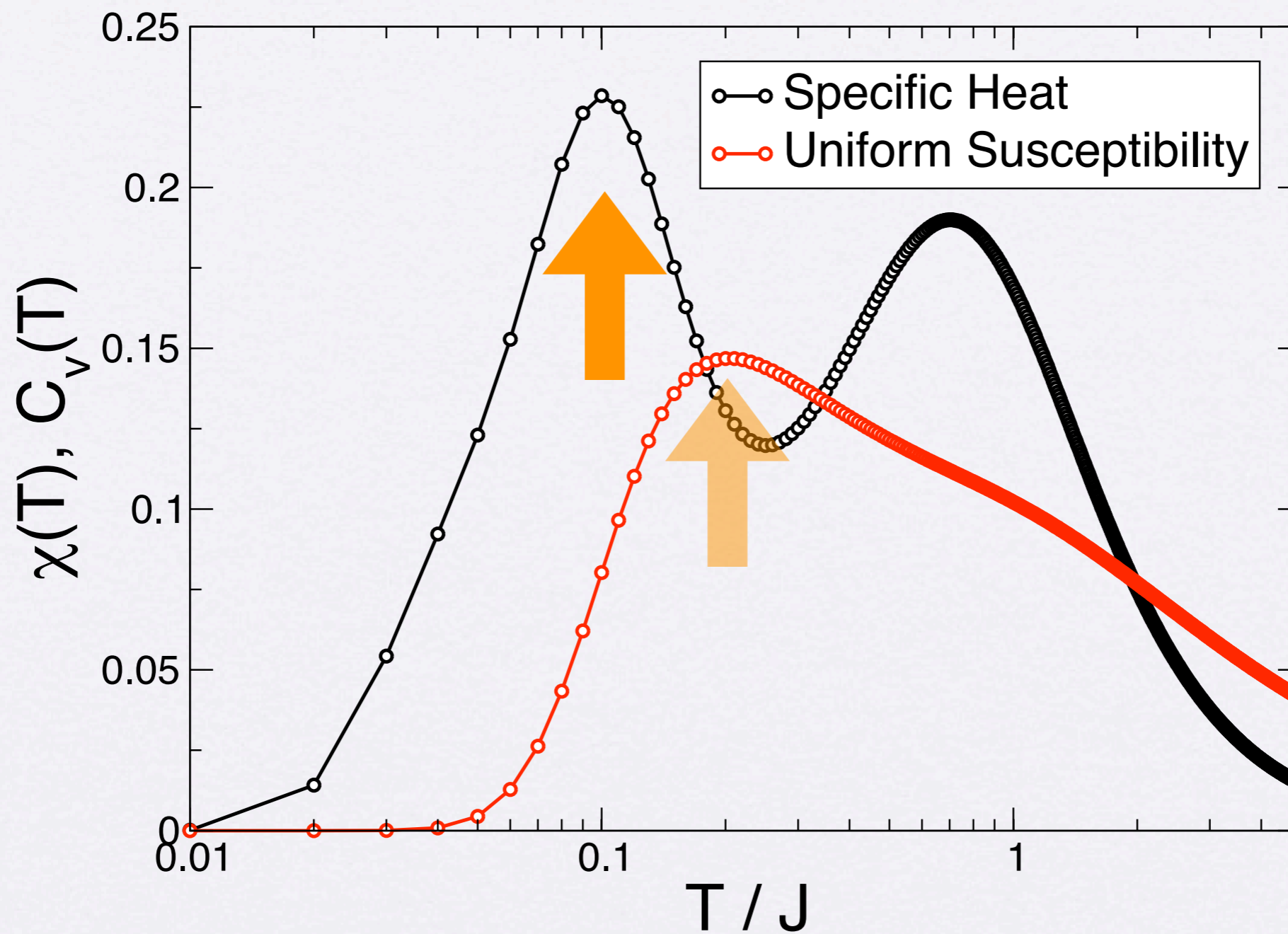
[\\$HOME/ALPS/lib/xml/lattices.xml](#)

chain, ladder,  
square, triangular, hexagonal,  
simple cubic, bcc, fcc, etc

[\\$HOME/ALPS/lib/xml/models.xml](#)

spin, hardcore boson,  
boson Hubbard, fermion Hubbard,  
Kondo lattice models, etc

# Thermodynamics of a 12 sites $S=1/2$ Kagome system



Dive into the mysteries of the  $S=1/2$  Kagome system ...

- Find the instructions for the tutorial here:  
<http://irrmawww.epfl.ch/scs/HFM-Tutorial.html>
- Further information on ALPS  
<http://alps.comp-phys.org/>

*Enjoy!*