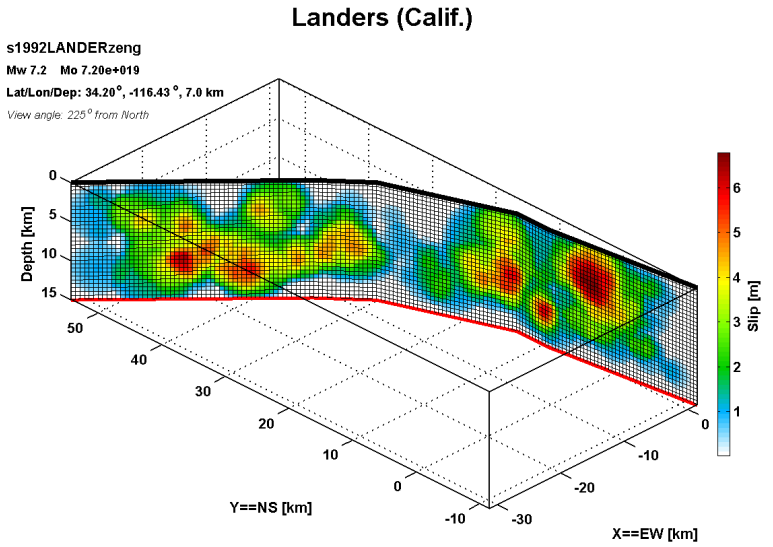
An Existing Testing Center: CSEP

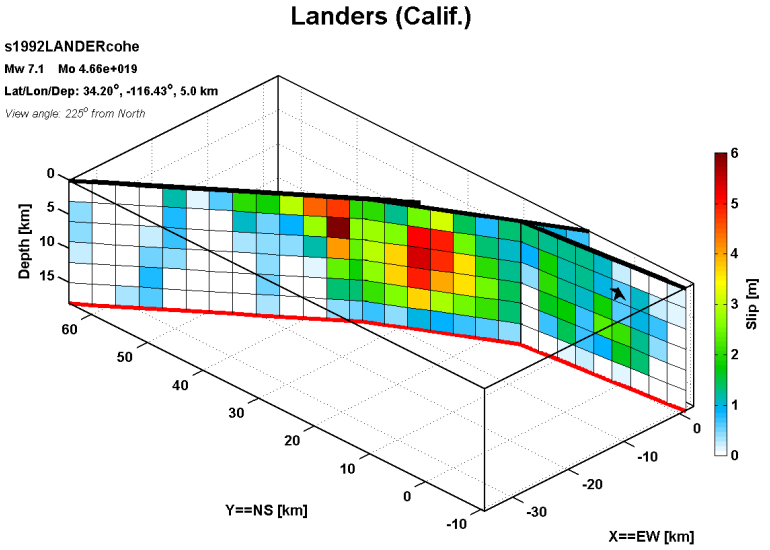
D. Schorlemmer (SCEC/USC)

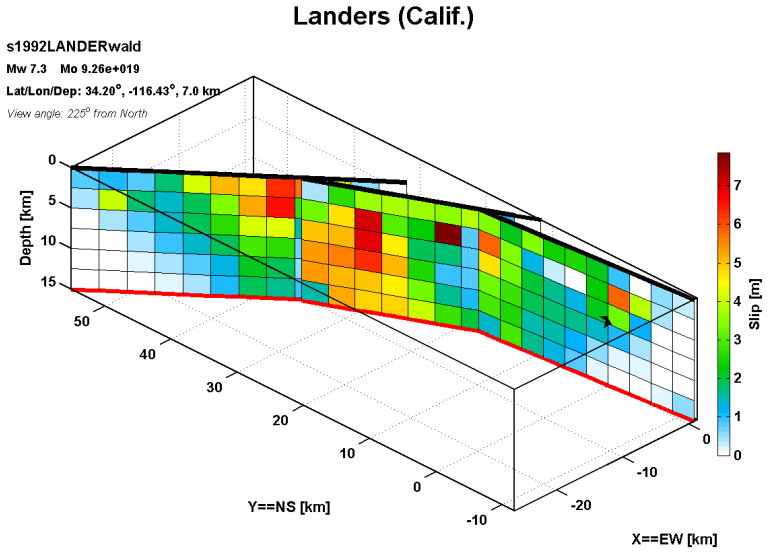
Problems in Assessing Predictions

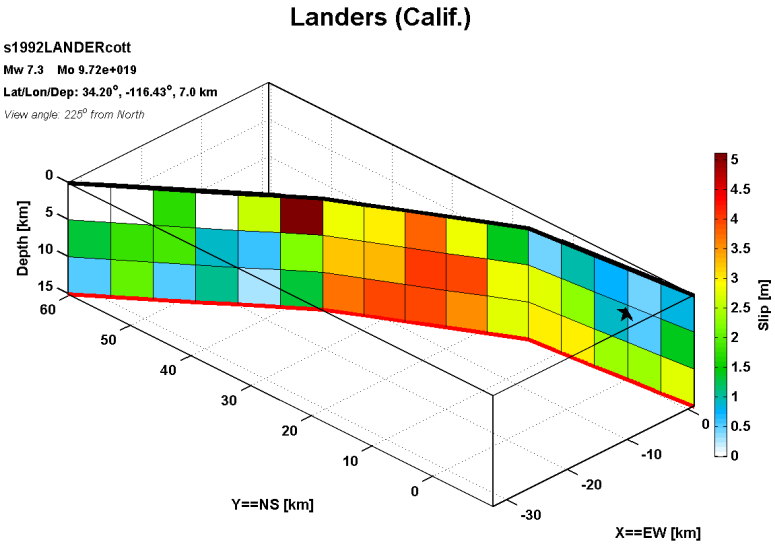
- Scientific publications provide insufficient information for independent evaluation
- Active researchers are constantly tweaking their procedures, which become moving targets
- Standards are lacking for testing predictions against reference forecasts
- Data to evaluate prediction experiments are often improperly specified
- Infrastructure for conducting and evaluating long-term prediction experiments does not exist

Same Problem









Strategy Change

Possible "problem solvers" (precursors):

- Seismic quiescence
- Foreshocks
- Radon emissions
- EM signals
- ...

None of the considered precursors can be used to predict earthquakes

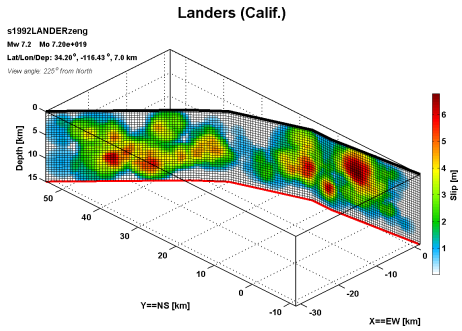
Let us first solve the 0-order problem!

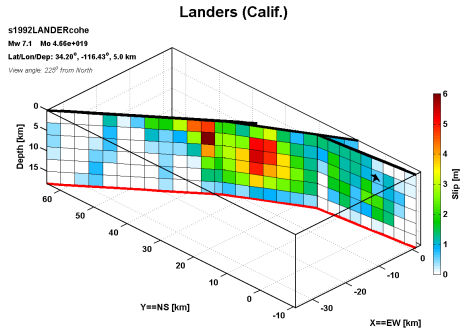
Strategy Change

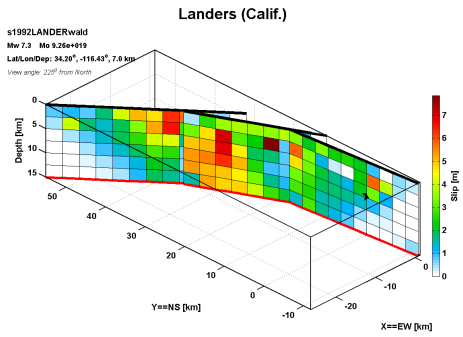
Develop generic forecast models, e.g., smoothed sesmicity

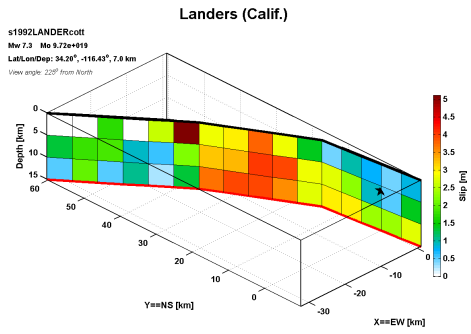
Test such models against future observations

Increase number of parameters and test for probability gain



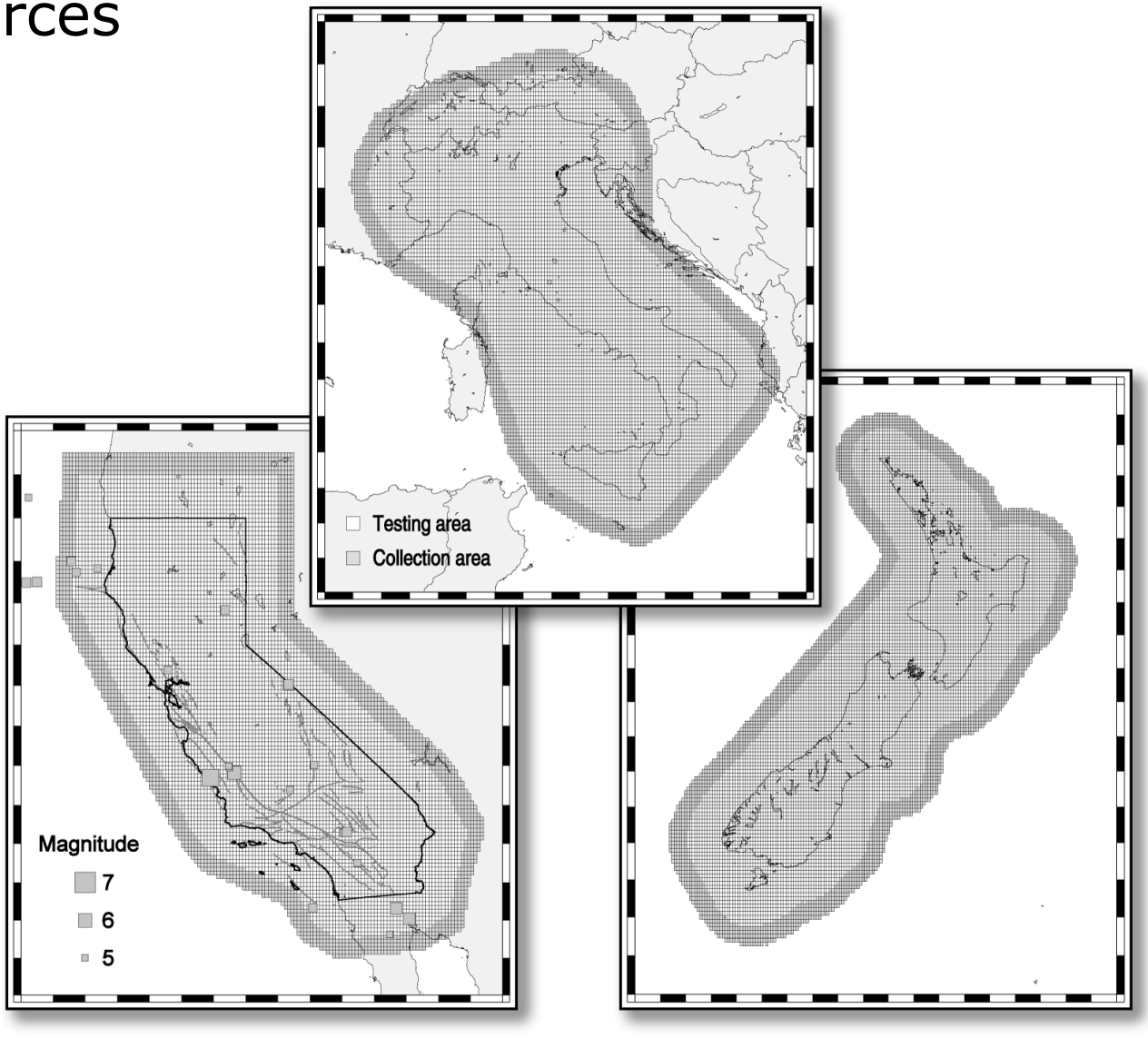






Four Essential CSEP Components

- Regional natural laboratories with adequate, authorized data sources



Four Essential CSEP Components

- Regional natural laboratories with adequate, authorized data sources
- Community standards for registration and evaluation of scientific prediction experiments


Earthquake Likelihood Model Testing

D. Schorlemmer,[1,2] M. C. Gerstenberger,[3] S. Wiemer,[1] D. D. Jackson,[4] and D. A. Rhoades[5]

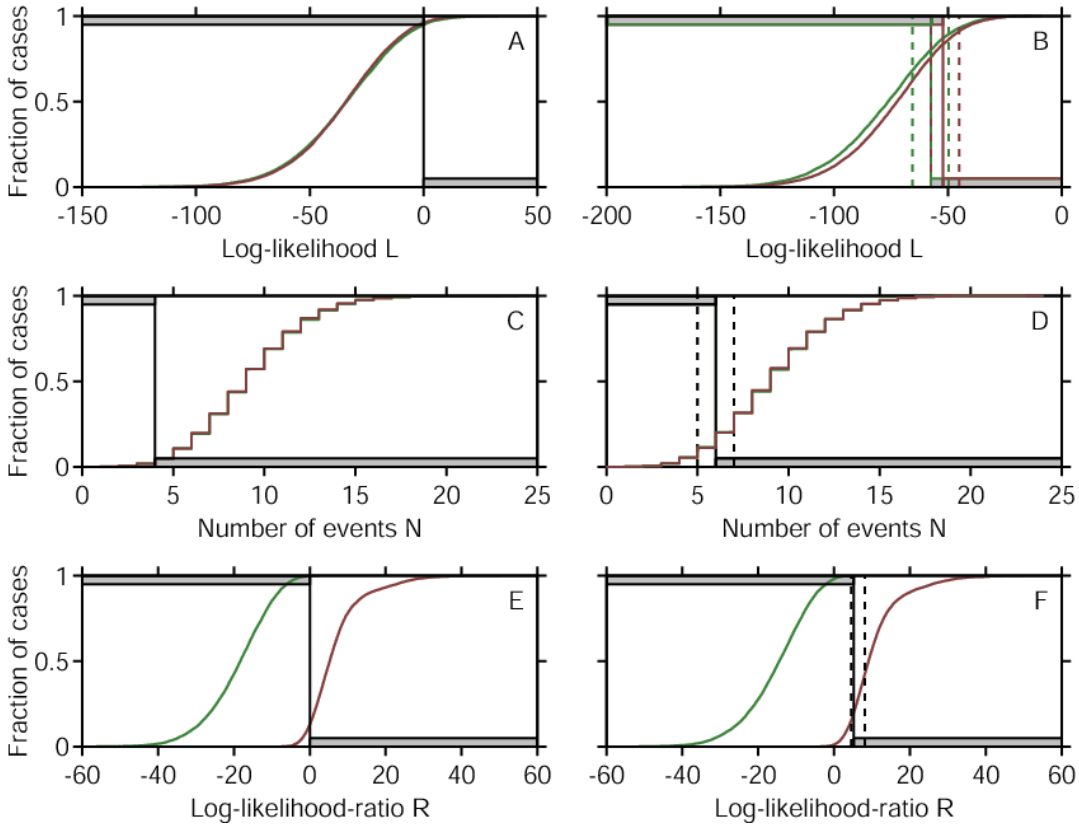
INTRODUCTION


RELM Testing Center

D. Schorlemmer

M. C. Gerstenberger

INTRODUCTION



Four Essential CSEP Components

- Regional natural laboratories with adequate, authorized data sources
- Community standards for registration and evaluation of scientific prediction experiments
- Testing facilities with validated procedures for conducting and evaluating prospective prediction experiments

Four Essential CSEP Components

- Regional natural laboratories with adequate, authorized data sources
- Community standards for registration and evaluation of scientific prediction experiments
- Testing facilities with validated procedures for conducting and evaluating prospective prediction experiments
- Communication protocols for conveying scientific results
  - the scientific community, including professional societies
  - government agencies responsible for risk management
  - the general public and other end-users

Four Goals of Testing Centers

1. Transparency
2. Controlled environment
3. Comparability
4. Reproducibility

RELM Testing Center

D. Schorlemmer

M. C. Gerstenberger

INTRODUCTION

Transparency

- All data cataloged
- All codes version-controlled
- Open-source codes (fully downloadable)
- Every change is documented

Controlled Environment

- Model codes cannot be changed (no moving targets)
- All data streams are authorized and fully specified
- All data streams come from independent providers and are not "interpreted" by the Testing Center (e.g., which fault ruptured during an earthquake)
- All models get identical input data
- All models are evaluated using the same testing codes

Comparability

- Models are tested against observation (consistency tests)
- Models are tested against each other (comparative testing)
- Avoids testing against overly simple null hypotheses

Reproducibility

- Any previous result can be recomputed:
  - All data used is cataloged
  - All codes are version-controlled
  - All system configurations are stored
  - All library packages (e.g., Python) are in a dedicated repository
- Tests can be rerun with alternative tests

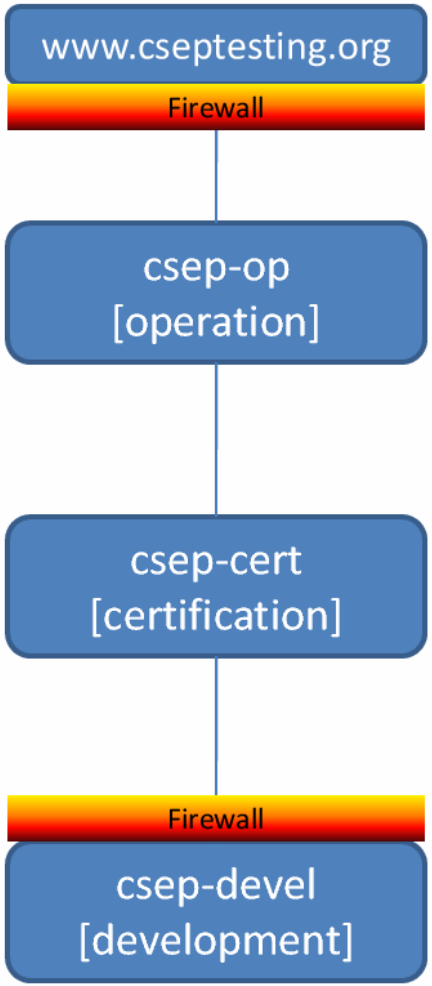
Testing Center

Hardware
2 Dual-Core AMD Opteron 2220
8 GB RAM
3.4 TB RAID 5 disks

OS/Kernel
Fedora Core 7 – 64 bit
2.6.22.1-41.fc7
gcc 4..1.2

Related Software
apache ant-1.7
R-2.5.0
elementtree-1.2.6-20050316
matplotlib-0.90.1
mpich2-1.0.3
CruiseControl

Identical hardware & software



Webserver
Restricted publication of results
Restricted dispatcher download
Test center setup information

Operational Machine
Automated Nightly builds from dispatcher
Publication of results to web server
Back up results
Quarterly Implementation of new models

Integration/Cert Machine
Automated Nightly builds from dispatcher
Emulates actual operation machine
Fulfill software dependencies
Implementation of new models

Development Machine
Import new models
Develop dispatcher
Fulfill software dependencies

Summary

Source inversion validation efforts may learn from CSEP

CSEP partly has similar needs (e.g., input data)

CSEP provides an excellent infrastructur for experiments

Maybe this can be a starting point for an interdisciplinary effort?