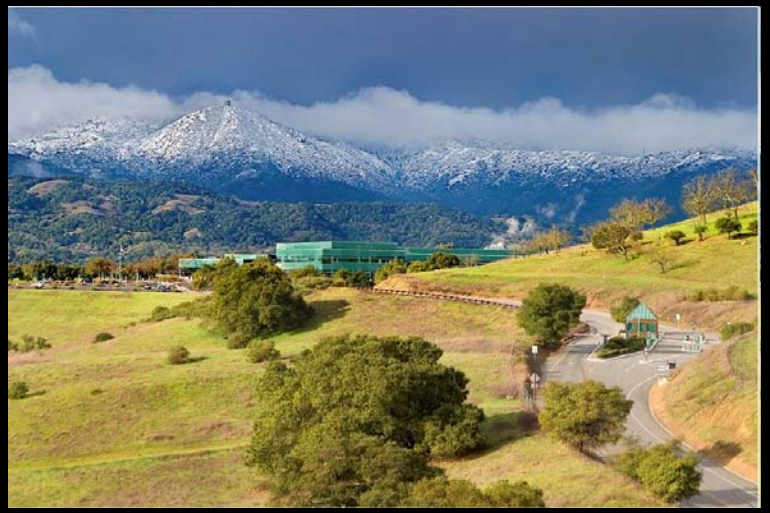
A SpatioTemporal Model for Influenza

Stefan Edlund[1], Michal Bromberg[2], Gabriel Chodick[3], Judith Douglas[1], Daniel Ford[1], Zalman Kaufman[2], Justin Lessler[4], Rachel Marom[3], Yossi Mesika[5], Roni Ram[5], Varda Shalev[3], and James Kaufman[1]

IBM Almaden Research Center[1], Israel Center for Disease Control (ICDC)[2], Maccabi Health Care Services[3], John Hopkins School of Public Health[4], IBM Haifa Research Center[5]

This project is being developed under Contract Number FA7014-07-C-0004, with the U.S. Air Force Surgeon General's Office (AF/SG) and administered by the Air Force District of Washington (AFDW).



August 2009

© 2009 IBM Corporation

Talk Overview

- **Our task**
  - Disease modeling, data, sources of errors in the data, estimating model accuracy
- **Our solution**
  - Compartmental disease model with forcing function
  - STEM and experiments
- **Initial results**
- **Report on recent results using model optimization techniques**
- **Future work**

© 2009 IBM Corporation

The problem at hand

- **Given 10 years worth of influenza incidence reports**
  - Build a model for influenza and fit the model to the reference data
    - Several options: Agent-based models, compartmental models, stochastic Markov-chains
  - Determine how quickly the model lose accuracy
    - Need a method to compare the result of model against the actual reference data

© 2009 IBM Corporation

The data

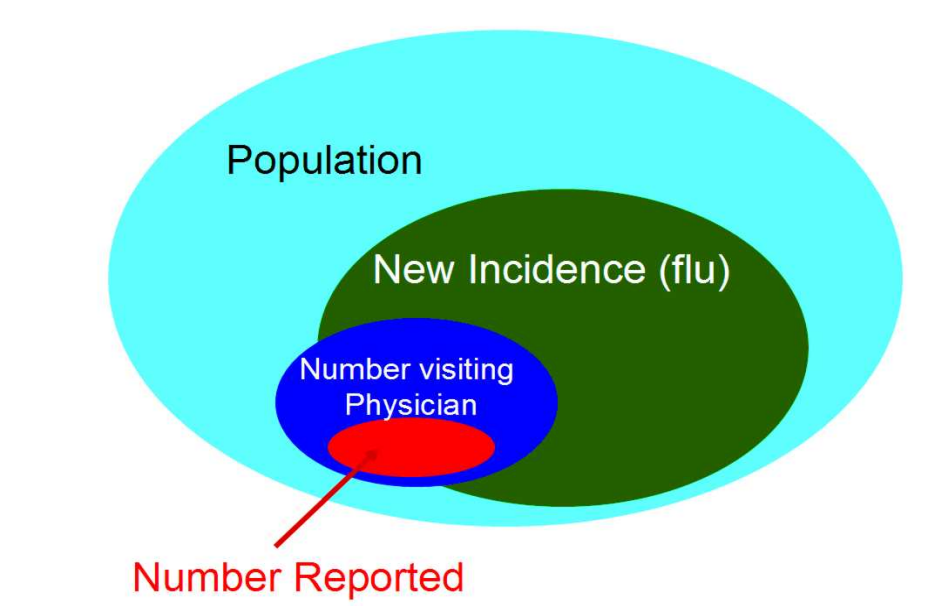
- **Data provided by the Israel Center for Disease Control (ICDC)**
  - Originated from Maccabi Health Care Services, 2nd largest HMO in Israel serving approx. 25 % of population
  - 10 years of summarized daily case reports of “Influenza Like Illness” (ILI)
    - Mapped to 49 administrative regions of the 15 Israeli sub-districts

© 2009 IBM Corporation

Sources of errors in the data

Misdiagnosis, under-reporting

Venn diagram



© 2009 IBM Corporation

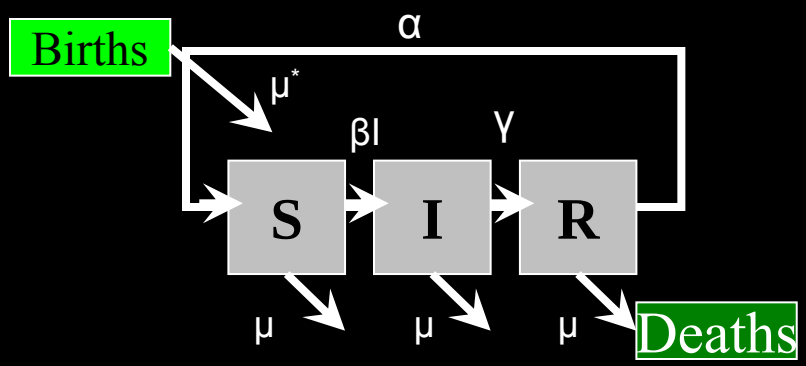
Attacking the problem

- **We picked a compartmental disease model. Why?**
  - Commonly used in epidemiology and deeply studied
  - Models can be implemented and simulated efficiently
  - Useful when “law of large numbers” applies
    - Agent-based models useful for small scenarios tracking individual cases
  - Open source software exists to implement the model
    - Spatio-Temporal Epidemiological Modeler (STEM)
- **What is a compartmental disease model?**

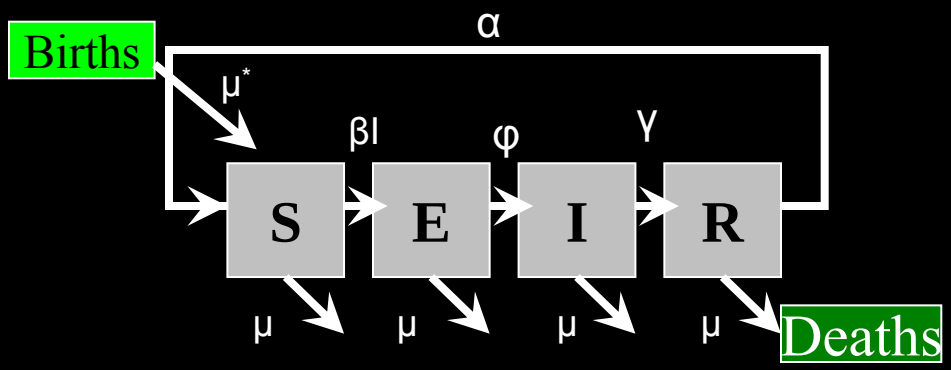
© 2009 IBM Corporation

Epidemiological Compartment Models

Standard SIR model



Standard SEIR model



S: susceptible
E: exposed but not yet infectious
I: infectious
R: recovered

$$\frac{dS(t)}{dt} = -\beta(\frac{S(t)}{P})I(t) + \alpha R(t) + \mu(P - S(t))$$

$$\frac{dI(t)}{dt} = \beta(\frac{S(t)}{P})I(t) - \gamma I(t) - \mu I(t)$$

$$\frac{dR(t)}{dt} = \gamma I(t) - \alpha R(t) - \mu R(t)$$

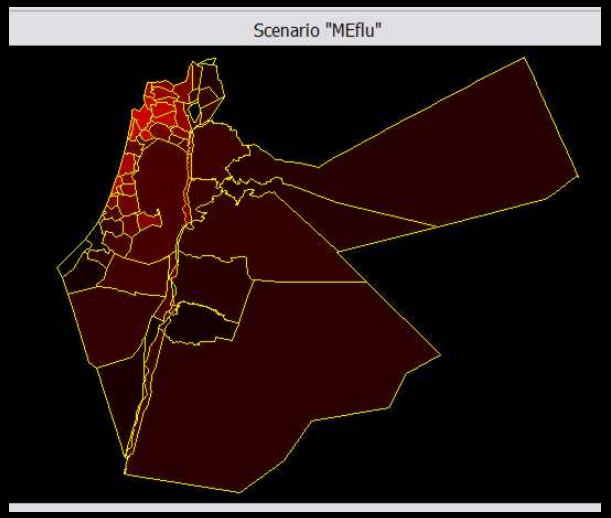
© 2009 IBM Corporation

The software

- **The Spatio Temporal Epidemiological Modeler (STEM)**
  - Available for free at http://www.eclipse.org/stem
  - Licensed under the Eclipse Open License
- **Collaboration by**
  - IBM research in California and Haifa, Israel
  - Universities: John Hopkins School of Public Health, University of Vermont and others

© 2009 IBM Corporation

STEM implements the latest Java™ based Component Software Architecture



Every STEM component is a “Plug-in” or “bundle” that can be independently developed, deployed, used, extended, and shared

Existing plugins define geography, transportation systems, and population for the 244 countries and dependent areas

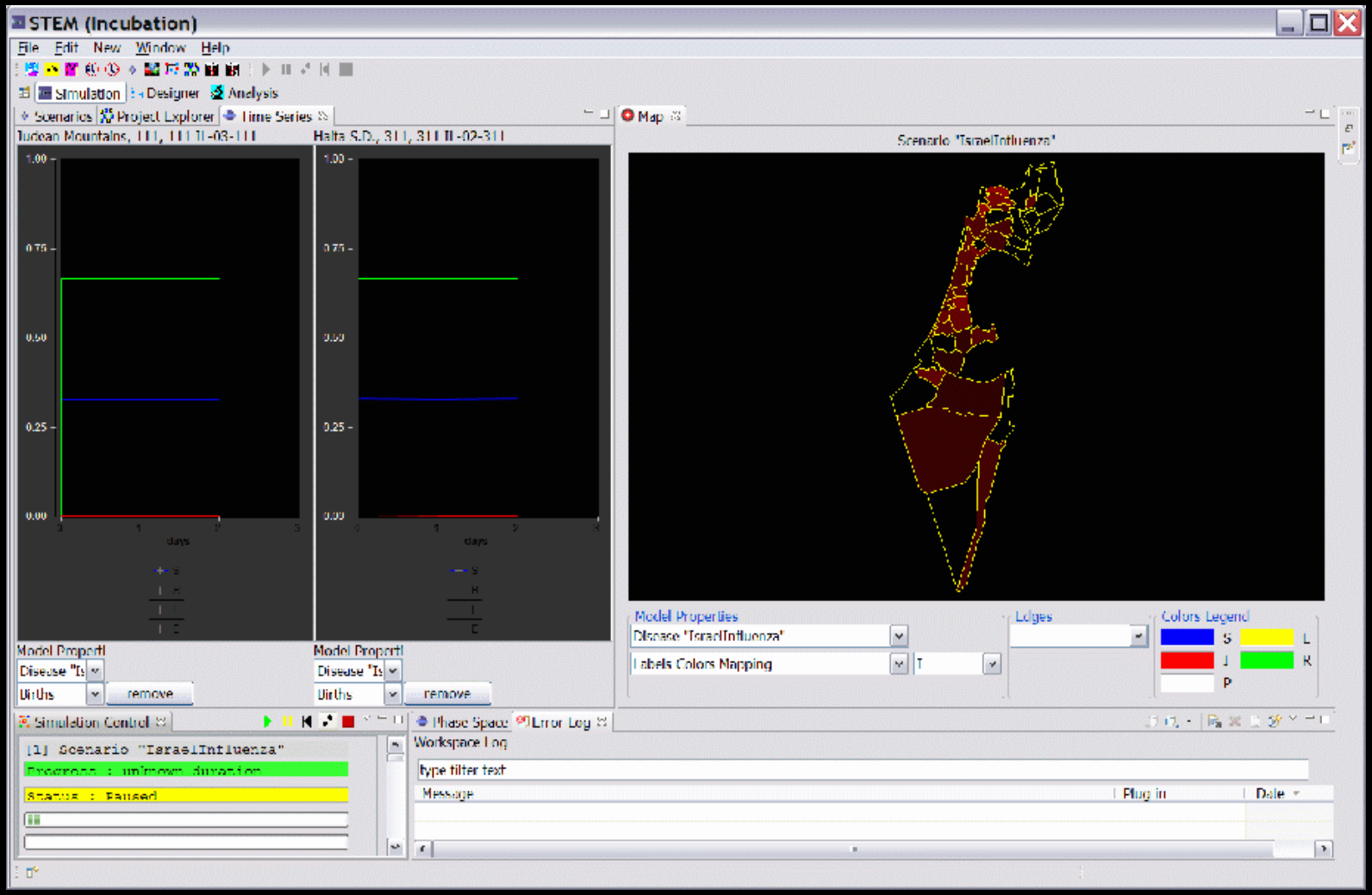
The STEM framework come from the Eclipse Foundation (Equinox)

- Based on industry-standard (OSGi) Software architecture
- Makes it possible to easily build one model on top of another

Users can

- create their own models for a country or region
- build on existing models and create new ones, making STEM extensible, flexible, and re-usable.

© 2009 IBM Corporation



© 2009 IBM Corporation

STEM

- **Comes pre-built with common compartment models**
  - SI, SIR, SEIR and stochastic variants
- **Models have a spatial component:**
  - The number of new infections depend on the number of infected individuals in neighboring regions.

$$\Delta I_j \propto \frac{\beta}{P_j} S_j I_j + \sum_k \frac{\beta}{P_j} \frac{m_{jk} P_j}{P_j + P_k} S_j I_k + \frac{\beta}{P_k} \frac{m_{jk} P_k}{P_j + P_k} S_j I_k$$

© 2009 IBM Corporation

STEM Core Integration Engine

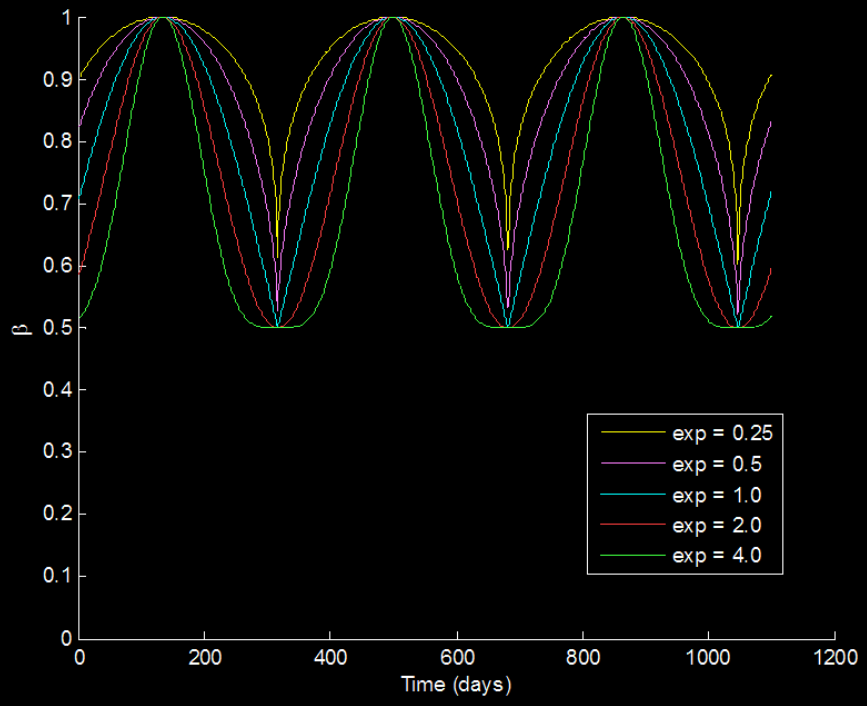
- **Choice Between**
  - Finite Difference Solver (fast, good for demos)
  - Runge-Kutta-Fehlberg (RKF45) *adaptive* integration
    - Adaptive step size
    - Very efficient
    - Synchronized across threads (multi-core engine)
    - Accurate
    - Computational error is estimated and controlled and an appropriate step size set automatically
  - User Contributed Solvers

© 2009 IBM Corporation

Influenza model

- **We extended STEM with a new model**
  - A “forcing” SIR disease model that captures the seasonality of the flu.

$$\beta(t) = \beta_o[(1.0 - a) + a\,|\sin(\varpi' t + \varphi)|^{\lambda}]$$



Transmission rate function

© 2009 IBM Corporation

Evaluating the model

- **STEM comes with built-in tools to compare models**
- **First, we need to transform the Israel data to the STEM format**
  - Comma separated files where columns are regions and rows are time (days):
    - 3 files:
      - S.csv (# of susceptible people)
      - I.csv (# of infectious people)
      - R.csv (# of recovered people)

	A	B	C	D	E	F	G
1	iteration	time	IL-00-411	IL-00-412	IL-00-421	IL-00-422	IL-00-
2	0	1998-01-01	455.5809	130.719	422.2649	1747.043	727.
3	1	1998-01-02	820.0456	104.5752	875.2399	3144.677	138
4	2	1998-01-03	656.0364	214.3791	700.1919	2515.742	11(
5	3	1998-01-04	2164.92	302.2222	1481.459	4341.984	285(
6	4	1998-01-05	2916.446	241.7778	1876.146	6312.531	359(
7	5	1998-01-06	3062.087	324.1412	2230.283	6505.894	454(
8	6	1998-01-07	3269.715	259.3129	2897.47	6296.617	451
9	7	1998-01-08	3344.701	207.4504	3047.343	8094.618	593(
10	8	1998-01-09	3586.923	296.6792	3205.629	8004.357	598(
11	9	1998-01-10	2869.538	237.3434	2602.891	6403.486	479(
12	10	1998-01-11	3206.792	712.7505	4232.025	9417.602	53(
13	11	1998-01-12	3203.247	570.2004	4383.7	10227.44	589(
14	12	1998-01-13	3747.108	456.1603	5234.407	10656.93	704
15	13	1998-01-14	3817.732	495.6472	5799.81	10927.73	730

© 2009 IBM Corporation

Data preparation

- **S, I, R was calculated by summing finite difference using incidence:**

$$\Delta S = -i(t) + \alpha R(t) + \mu(P - S(t))$$
$$\Delta I = i(t) - \gamma I(t) - \mu I(t)$$
$$\Delta R = \gamma I(t) - \alpha R(t) - \mu R(t)$$

- Assumptions:
  - Recovery rate γ was 0.1(ten day infectious period)
  - Immunity loss rate α was 0.001 (immunity lasing approx. 3 years)
- Birth and death rate from actual population data for Israel
- Initial recovered population about 65 %
  - From estimates that in a given year about 30-40 % of people are susceptible to the seasonal flu.

© 2009 IBM Corporation

Error function

- **For each location and at every time step, calculate a root mean square error (RMSE) between the model and the reference data**
  - In this report, we only looked at the I (infectious) data
- **Find a "best fit" by minimizing RMSE over time**

$$RMSE(I_s, I_r) = \frac{\sum_{t \in T} RMSE(I_s, I_r, t)}{|T|}$$

© 2009 IBM Corporation

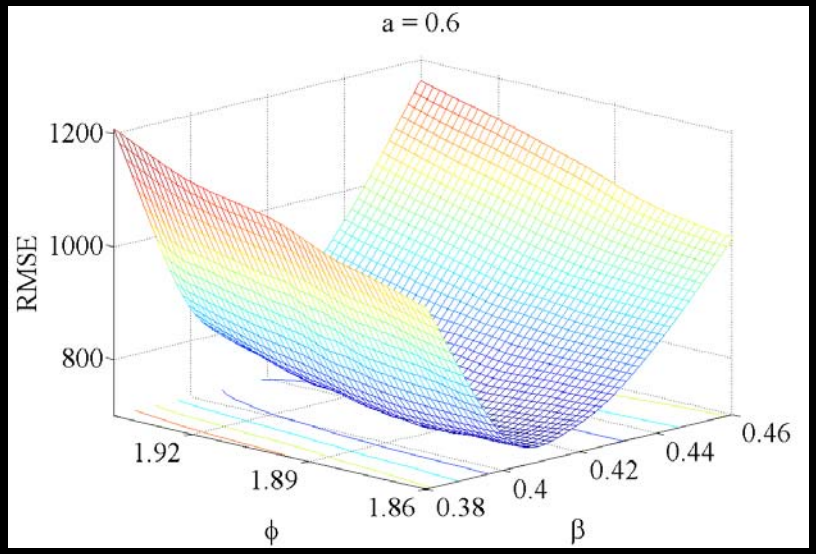
Initial results

- **We manually ran simulation in STEM and varied 5 parameters:**
  - β (transmission rate), m_{jk} (mixing rate between neighboring regions), λ (modulation exponent), *a* (modulation amplitude) and φ (modulation phase shift).
  - Simplification: m_{jk} was assumed identical for each region
- **A large parameter space was examined**
  - Very time consuming!
- **Initially fitted only against the first two years of data**

© 2009 IBM Corporation

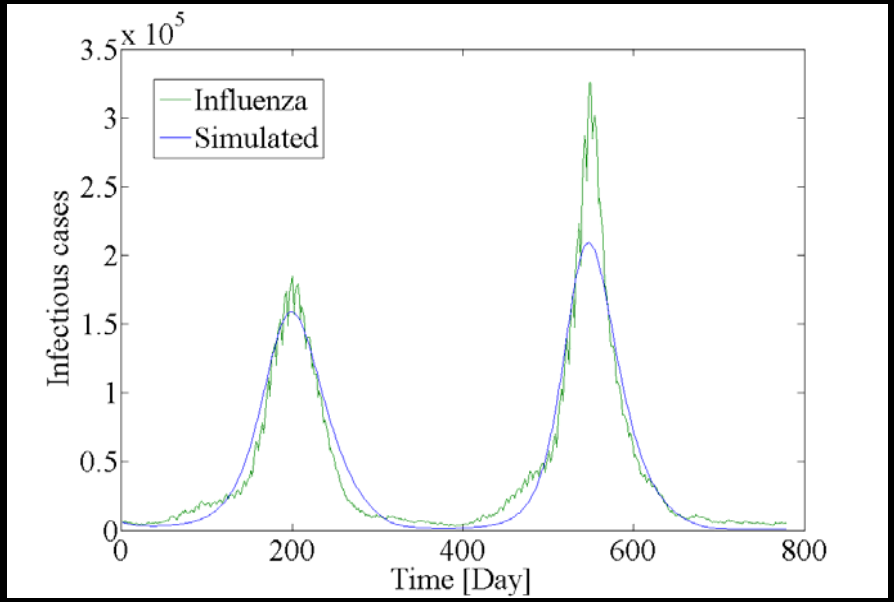
Initial results

need to explore a large phase space of parameters!!



β = 0.41, *m* = 0.9, λ = 0.55, *a* = 0.6 and φ = 1.87

Well mixed population

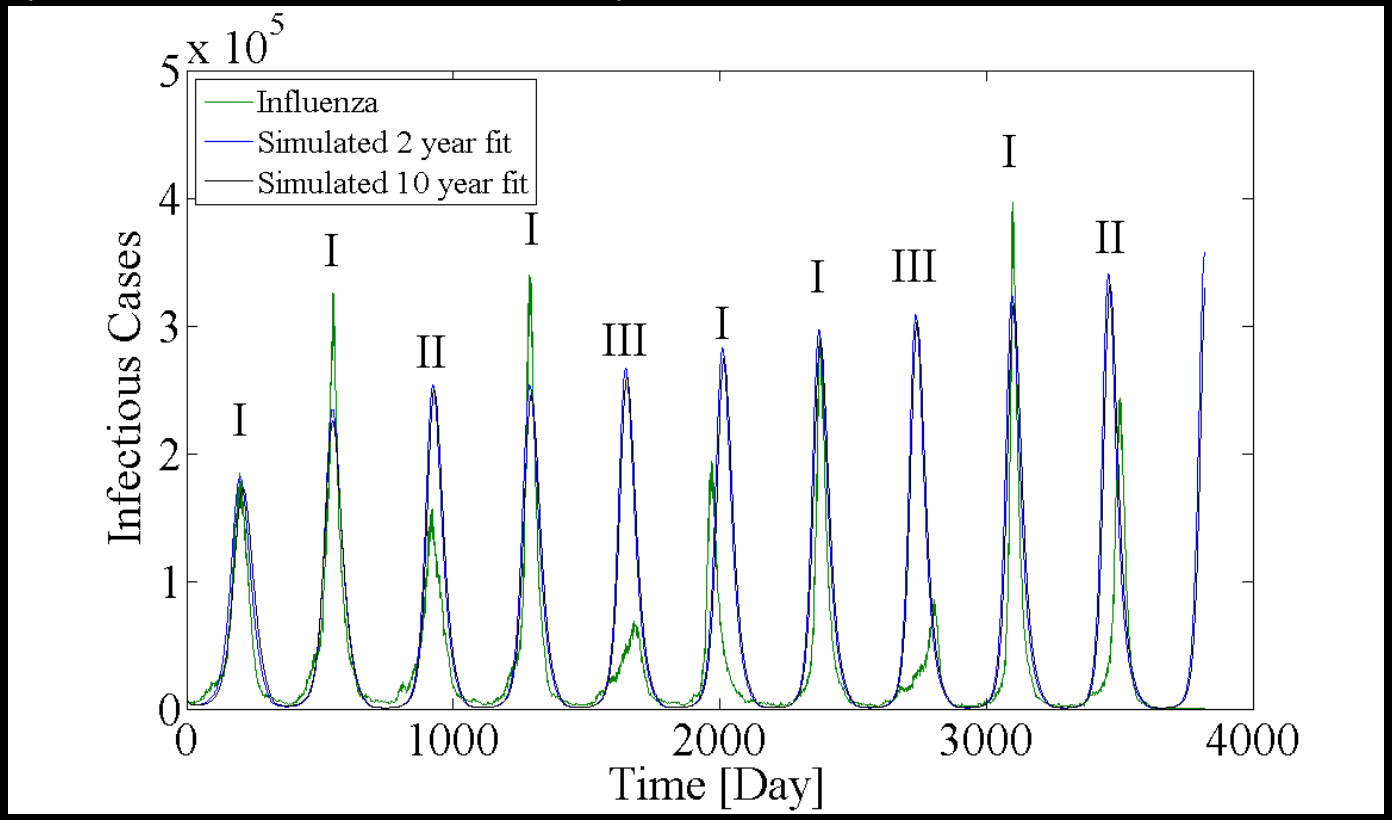


© 2009 IBM Corporation

Initial results

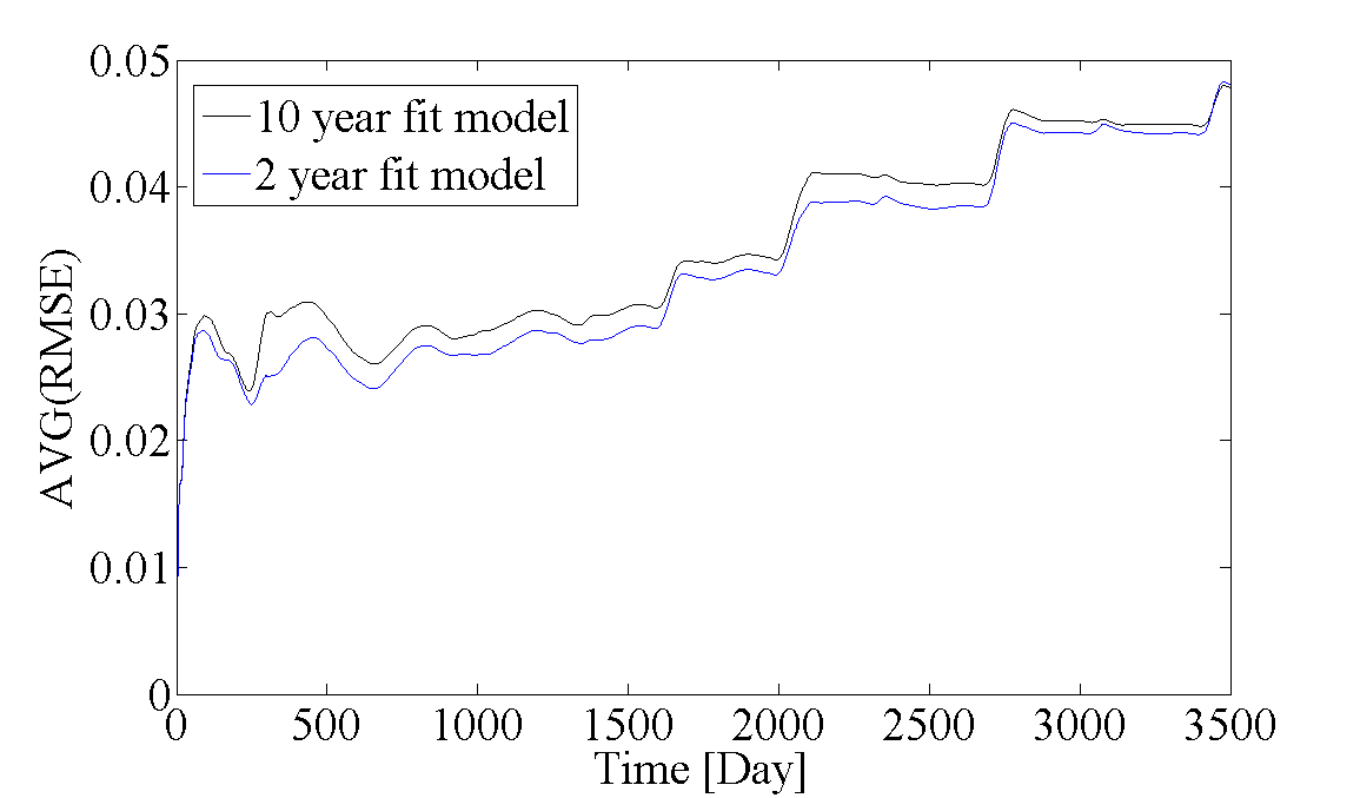
- **We also fitted against all 10 years of available data**
  - β = 0.40, *m* = 0.9, λ = 0.55, *a* = 0.6 and φ = 1.85

I = A/H3N2, II = A/H1N1 and III = B



© 2009 IBM Corporation

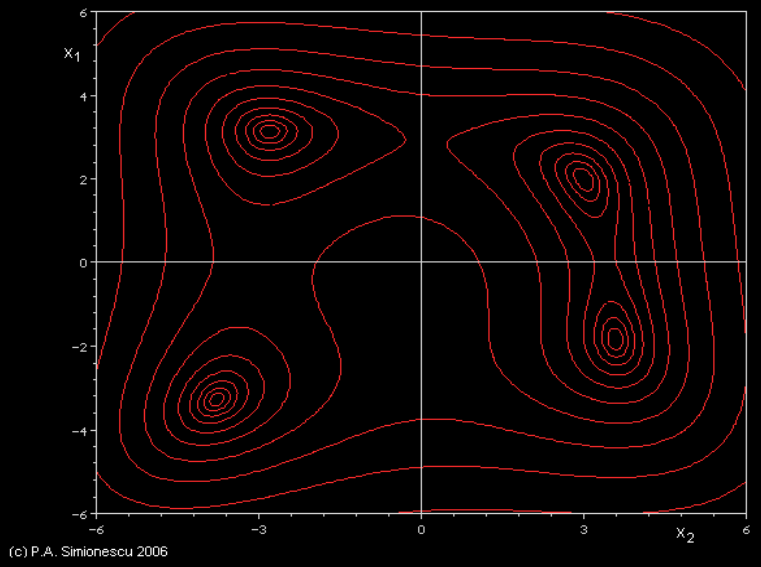
Model accuracy loss over time



© 2009 IBM Corporation

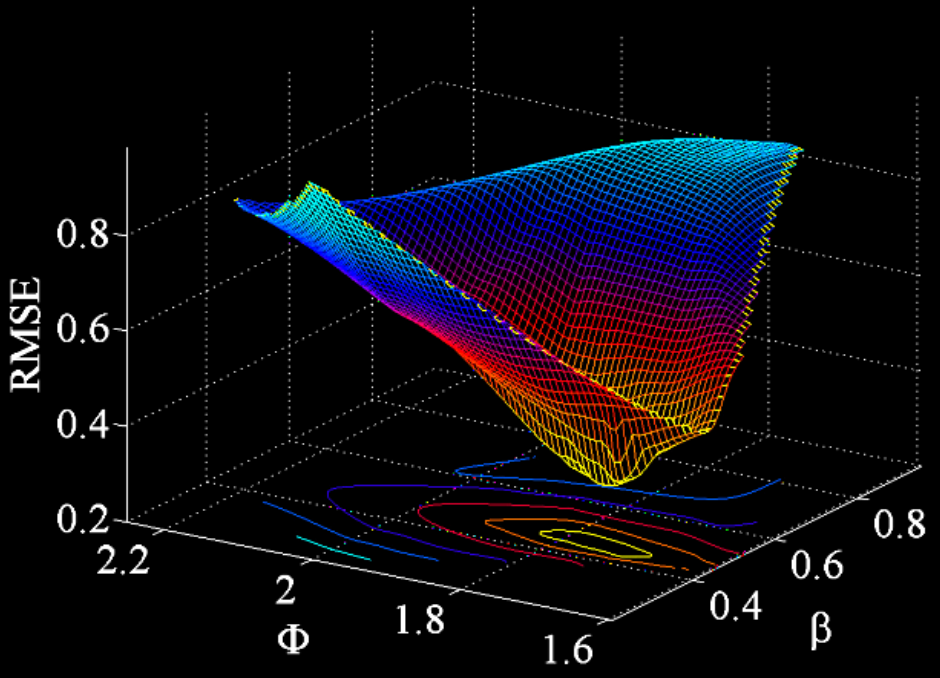
Recent results

- **A recent feature in STEM allows for automatically walking the parameter space and find an optimal set of parameters**
- **Nelder-Mead algorithm**



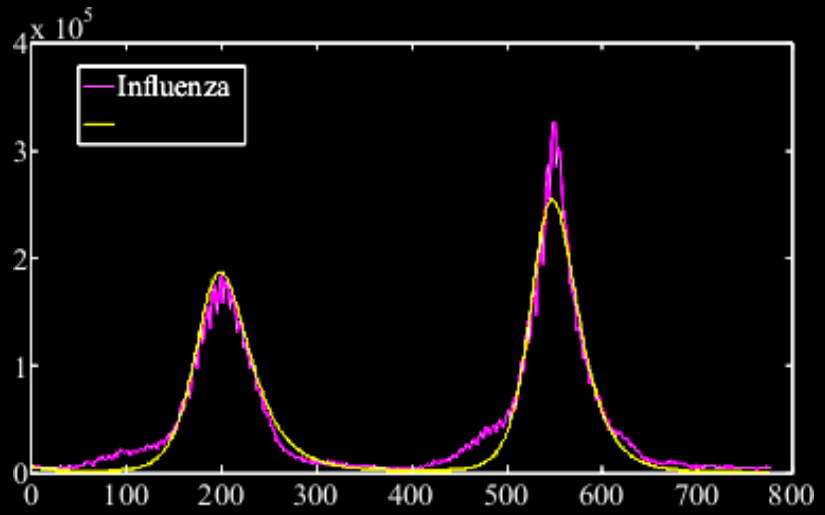
© 2009 IBM Corporation

Two Year Fit, Single SIR(S)



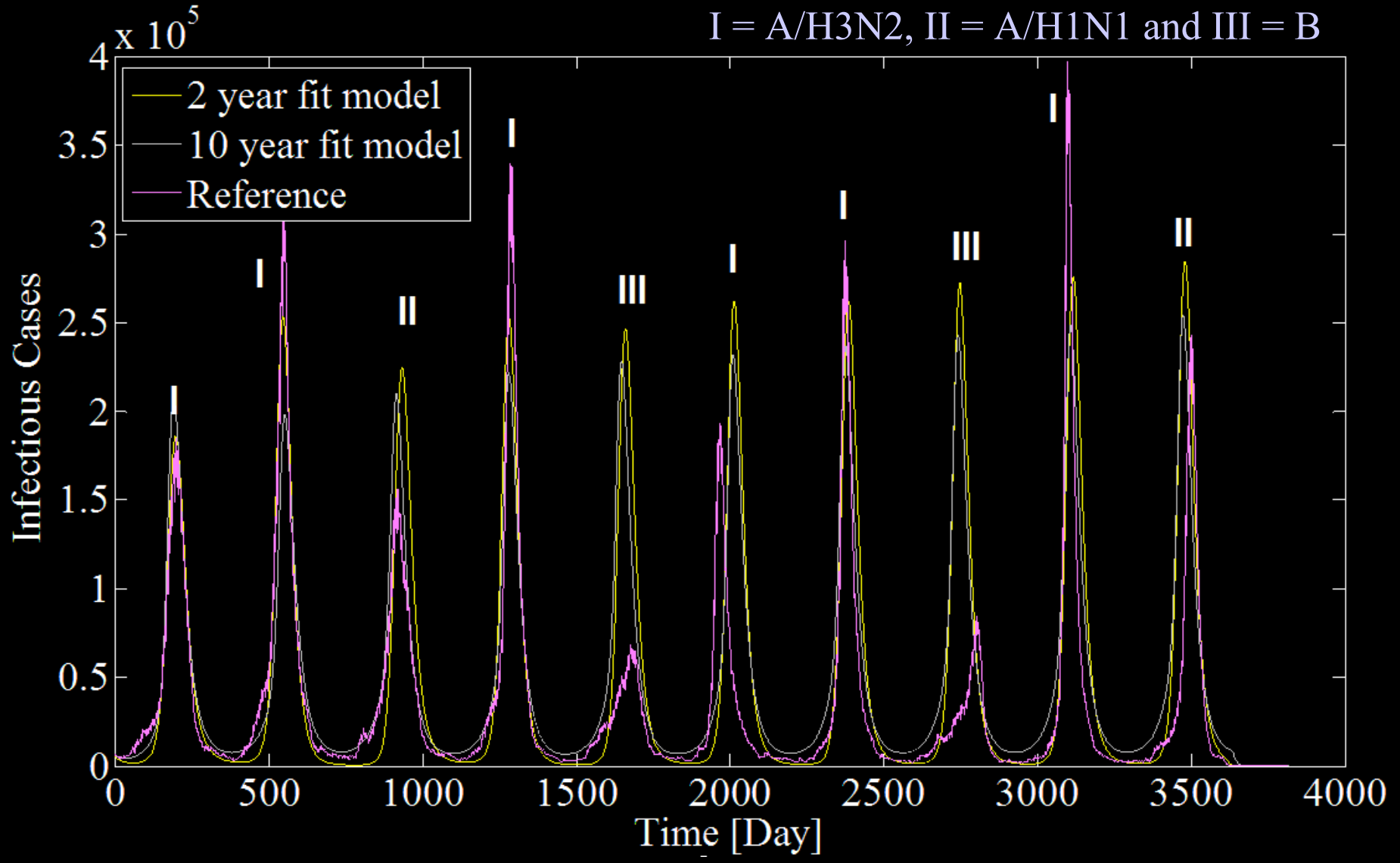
Minimum at φ = 1.75, $\beta = 0.45$,
m = 0.9, λ = 5.7 and a = .34

Best fit for first 2 waves



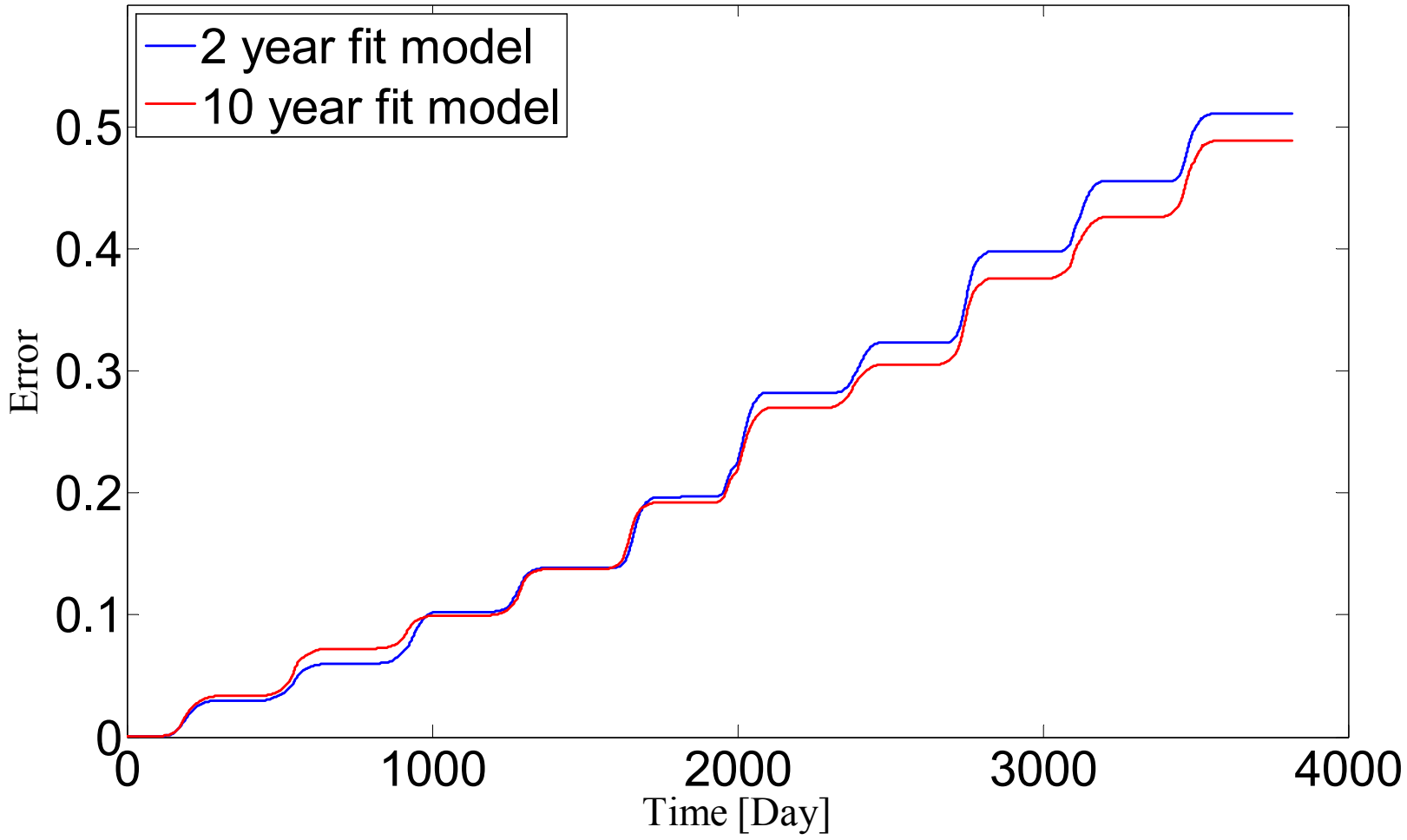
© 2009 IBM Corporation

A single SIR(S) model: 2 year and 10 year fits to historic data



© 2009 IBM Corporation

* Modified error function ignoring data points between flu seasons



© 2009 IBM Corporation

Future work

- **The model we used is a simplification**
- **The dominant Flu strain circulating each year is different**
  - We need a true “multi-serotype” model in STEM to capture it
  - A fitted multi-serotype model would allow us to ask “what if” questions:
    - E.g. If the dominant strain next year is H1N1, how many cases can we expect?
    - Can be used as input to epidemiologists and public health officials to determine flu-shot ingredients.
- **Some initial results on a single strain H3N2 model has been implemented and submitted to ACM TOMACS**
- **A true “global” model without geographic limitations will be needed**

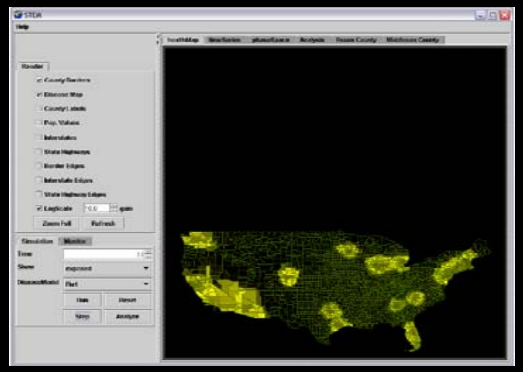
© 2009 IBM Corporation

Goal: Evaluation of Public Health Policies

HCN/HL7

Weather

Air Traffic



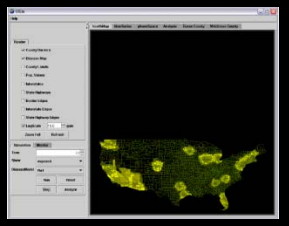
Masks

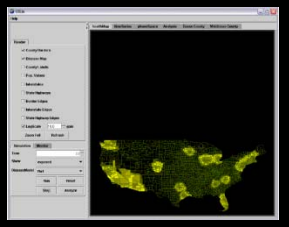
No Air travel

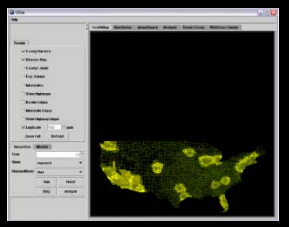
Vaccinate

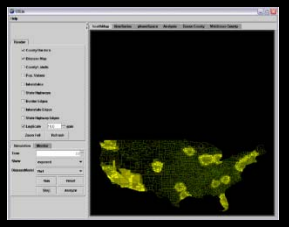
Masks/No Air

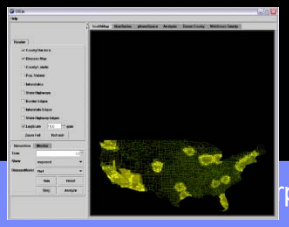
?











Multiple parallel scenarios identically initialized from current real world conditions and simulate each simultaneously forward in time.

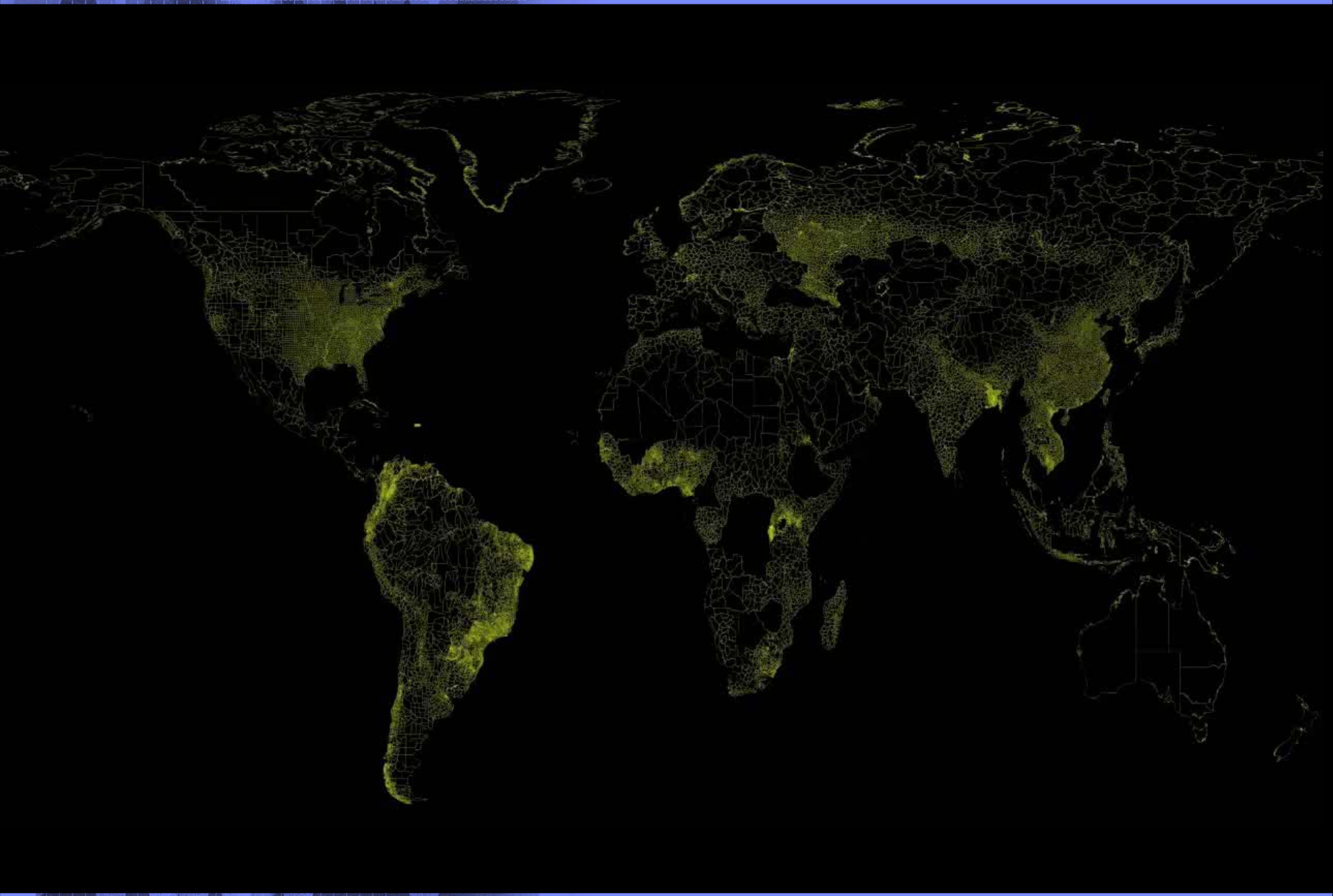
STEM links

- **Main Web Site: http://www.eclipse.org/stem**
- **YouTube tutorial: http://www.youtube.com/watch?v=LfiibQX4IFE**
  - http://www.youtube.com/watch?v=3S5DbjCHsx4 (Spanish version)

© 2009 IBM Corporation

Global Flu Model Animation

© 2009 IBM Corporation



© 2009 IBM Corporation

Questions ???

© 2009 IBM Corporation

BACKUP

© 2009 IBM Corporation

Why Open Source?
Need for an International Community

Eclipse Open Healthcare Framework

UPMC
Pitt
Stanford
MIT

iCDC

U. Helsinki

MECIDS

University of Vermont (UVM)

In discussion/meetings planned

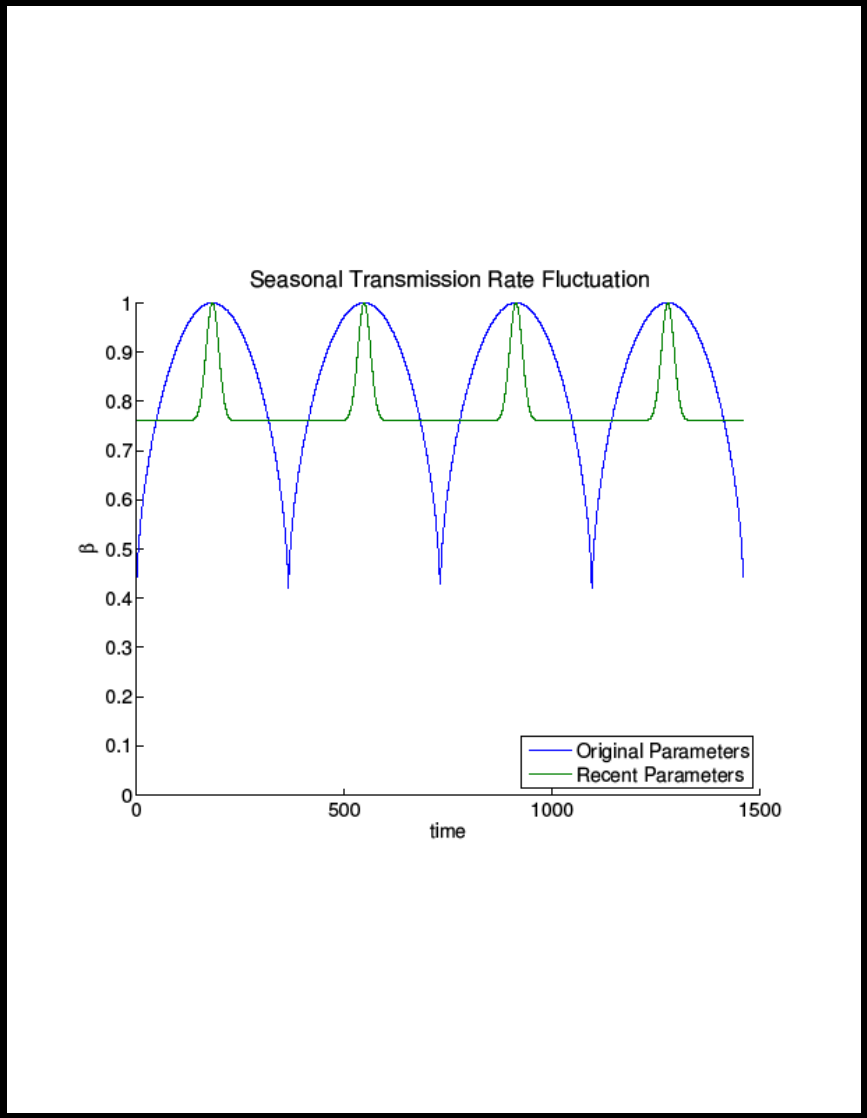
Northrop Grumman

France (Government, Universities)

UC Davis

University of Edinburgh

© 2009 IBM Corporation



© 2009 IBM Corporation