Endogenous versus exogenous dynamics and scaling laws in YouTube, Open Source Softwares and Cyber-risks

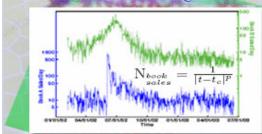
Didier SornetteCOChair of Entrepreneurial RisksRilETH ZurichThe(Swiss Federal Institute of Technology, Zurich)Department of Management, Technology and Economicshttp://www.er.ethz.ch/

Collaborators: Riley Crane (post-doc) Thomas Maillart (PhD student)



Social networks

Endogenous vs Exogenous Shocks



The response of the bookbuyers can be characterized with power laws and rigorously described using theories of epidemic propagation with memory in a network of acquaintances. This reflects

a self-similar fractal-like response of human decision making.

The organization of social networks displays many interesting scaling properties from "small-world" to "scale-free" which have consequences for the dynamical behavior of the system. The mathematical characterization of the topology is currently a very active area of research.

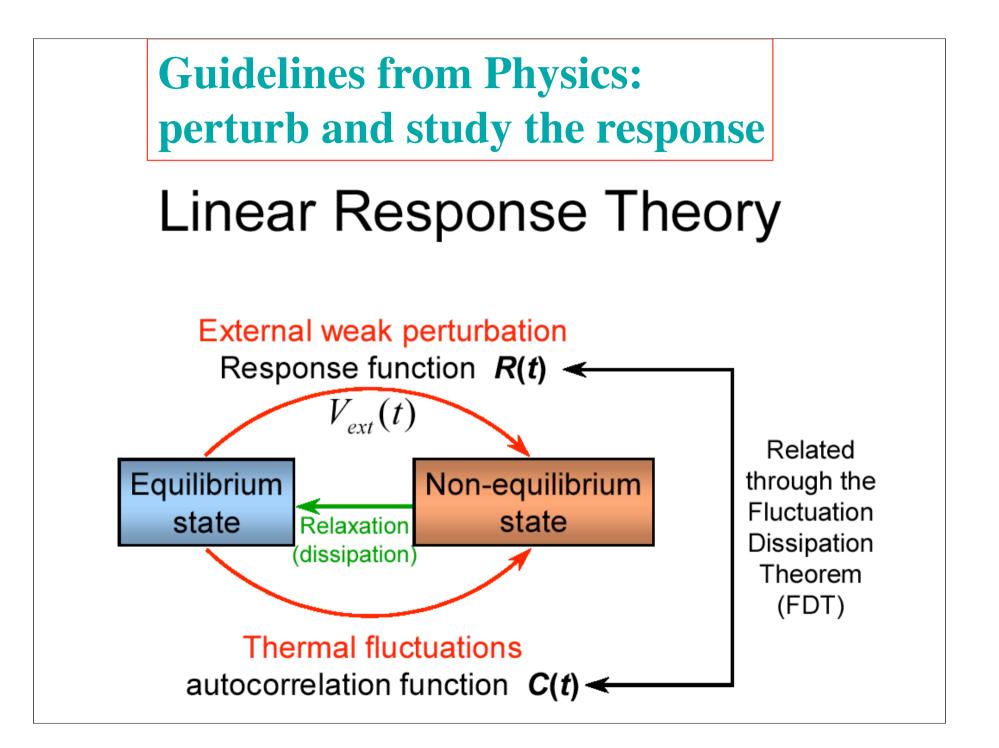
Publishers face the problem of optimal allocation of resources Complex Interacting Social Networks



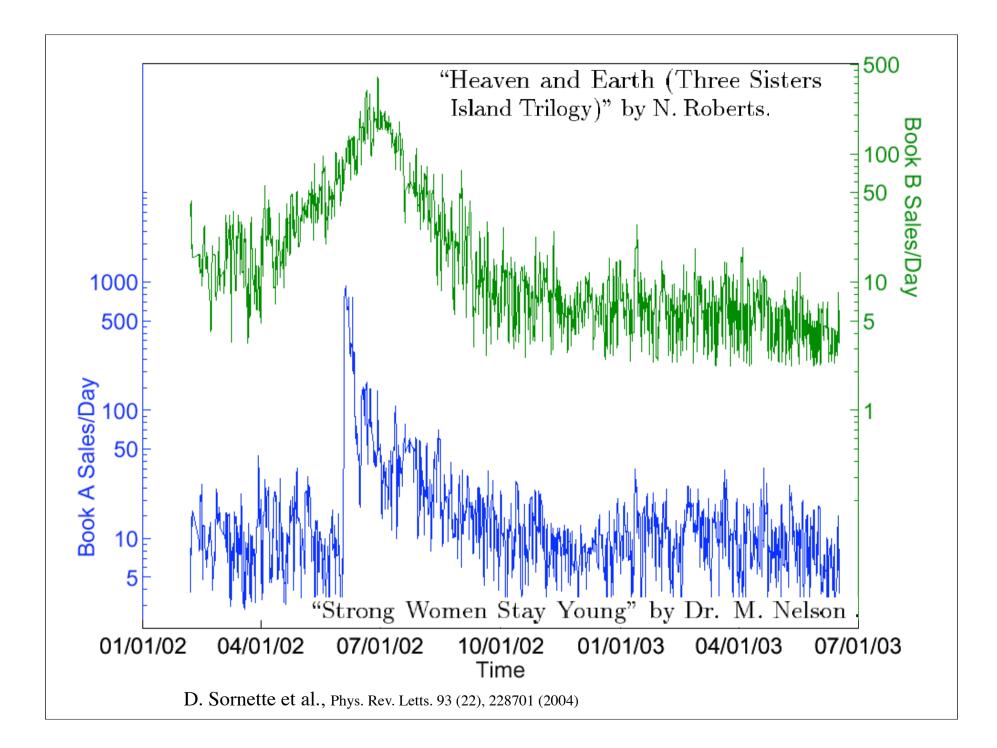
This research will give publishers the tools they need to optimally allocate their marketing and publishing resources allowing them to modify their business model in a constantly evolving marketplace.

Other Applications

Commercial sales (Books, CDs, DVDs, etc.), Theatre Attendance of Hollywood Movies Open Source Projects, Search Engines, Real Time Measure of the Social Climate,

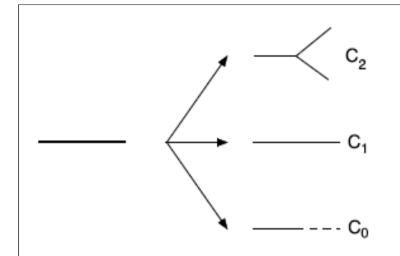






Epidemic processes by word-of-mouth





Simplest example of branching

Definition of the branching model: starting from an existing branch, with probability C_0 the branch stops at the next step; with probability C_1 , the branch continues to grow at the next step; with probability C_2 , it develops two branches

- 1. if $C_1 + 2C_2 < 1$, the average number of generations is finite and all earthquakes are finite. Their size distribution is an exponential.
- 2. if $C_1 + 2C_2 > 1$, the probability to generate a "run away" (i.e. an event of infinite size) becomes nonzero. This is similar to being above the threshold in the percolation model presented in Chap. 12.
- 3. if $C_1 + 2C_2 = 1$, the system is critical and the size distribution of events is a power law as we show below.

The critical condition $C_1 + 2C_2 = 1$ together with the normalization $C_0 + C_1 + C_2 = 1$ yields the condition $C_0 = C_2$ at criticality.

$$P(E) \simeq A e^{-aE} E^{-(1+\mu)}$$
 $a \simeq \frac{(C_0 - C_2)^2}{4C_0}$ $\mu = \frac{1}{2}$

Mean field theory of Hawkes self-exciting conditional Poisson Process

$$A(t) = \int_{-\infty}^{t} d\tau \ \eta(\tau) \ K(t-\tau)$$

Exogeneous shock

$$A(t) = \int_{-\infty}^{\infty} d\tau \ \eta(\tau) \ K(t-\tau)$$

Exogeneous shock

$$E_{\text{exo}}[A(t)] = A_0 \ K(t) + n\langle \eta \rangle$$

$$n = \int_{0}^{+\infty} \tau \ K(\tau)$$

Endogeneous shock

 $n = \int_0^{+\infty} \tau K(\tau)$

$$E[X(t)|Y = A_0] - E[X(t)] = (A_0 - E[Y]) \frac{Cov(X(t), Y)}{E[Y^2]}$$

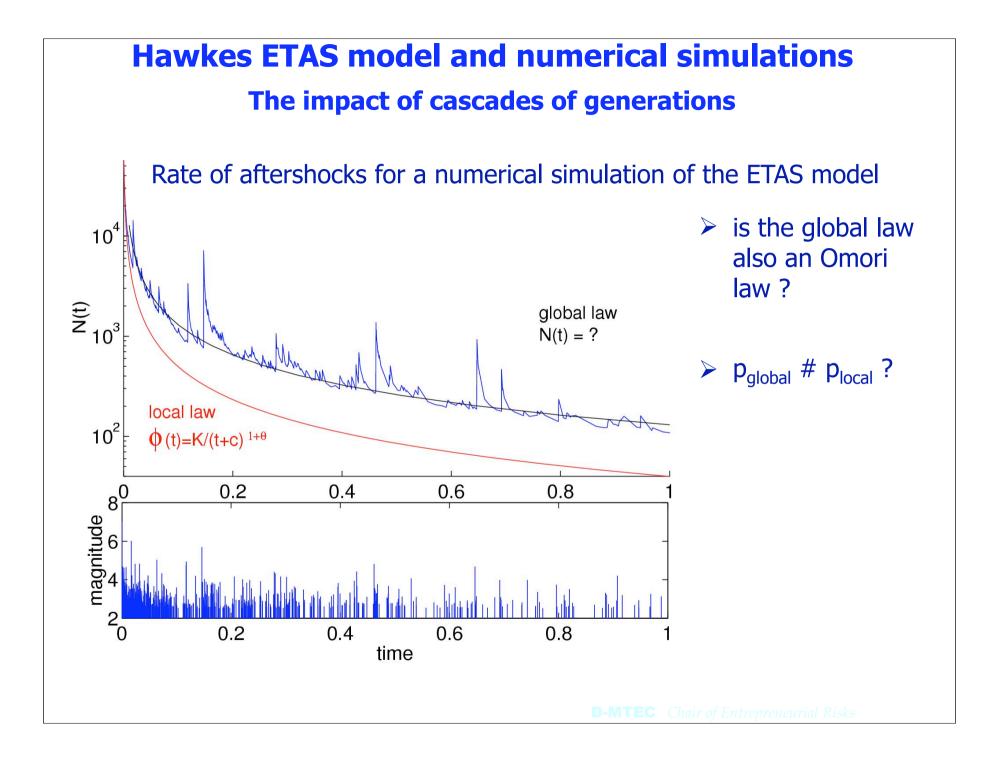
Cov(A(t), A(0)) =
$$\int_{-\infty}^{0} d\tau \ K(t - \tau) \ K(-\tau)$$

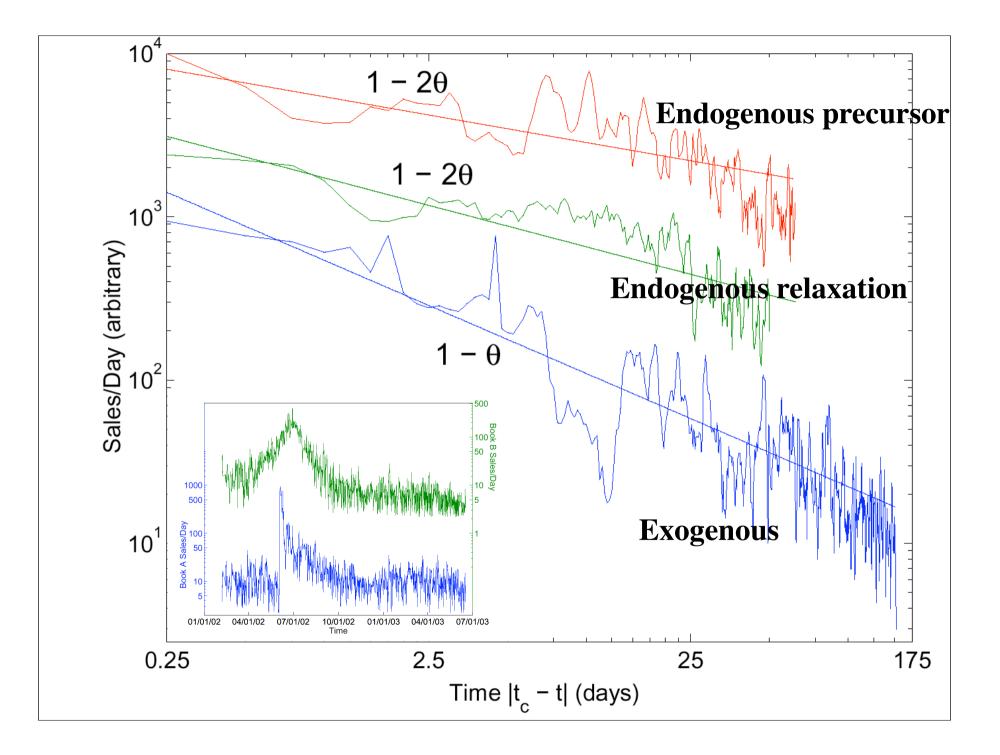
$$\operatorname{E}_{\text{endo}}[A(t)|A(0) = A_0] \propto A_0 \int_0^{+\infty} du \ K(t+u) \ K(u)$$

Theory: Null Hypotheses

 The tests are about the slopes of the response functions, conditional on the class of peak determined by the slope of the growth AT CRITICALITY n=1

	Endogenous	Exogenous
Foreshock (or growth)	$S(t) \propto rac{1}{\left t ight ^{1-2 heta}}$	Abrupt peak
Aftershock (or decay)	$S(t) \propto \frac{1}{t^{1-2\theta}}$	$S(t) \propto \frac{1}{t^{1-\theta}}$
Non-critical: $S(t) \propto \frac{1}{t^{1+\theta}}$		





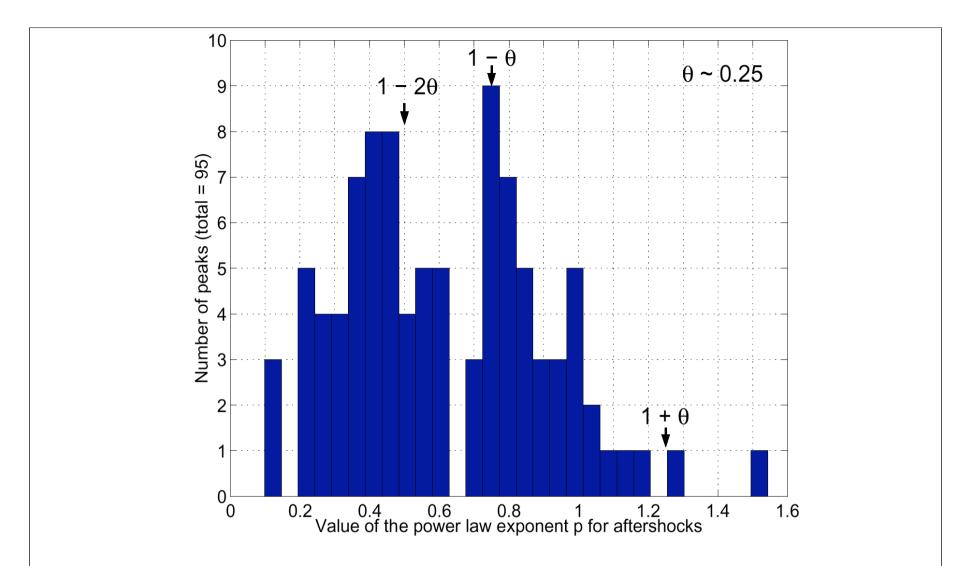
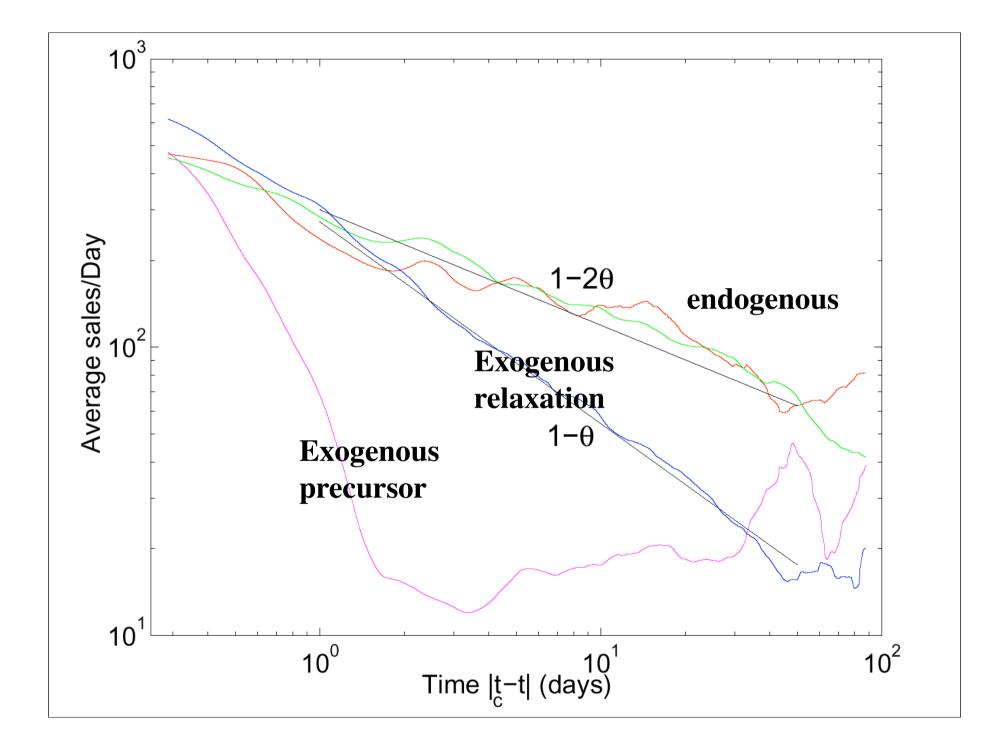
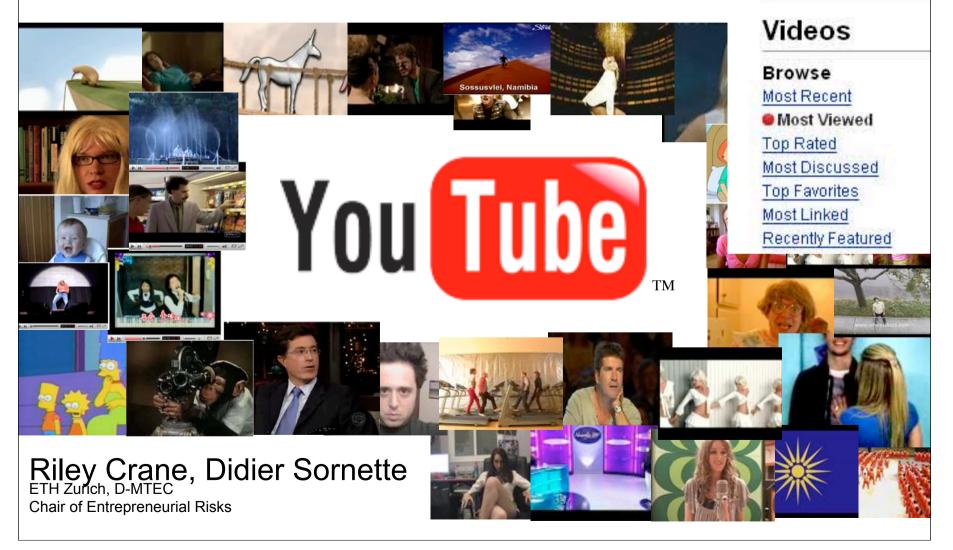


Figure 4: Histogram of the estimates of the power law exponents p of the relaxations of the sales of books following the largest peaks. The sample is obtained from 77 books, which yield 95 major peaks. One can clearly identify two classes of exponents: endogenous cluster for $1 - 2\theta$ close to 0.45 and exogenous cluster for $1 - \theta$ close to 0.75, compatible with the estimation $\theta = 0.3 \pm 0.1$. The tail of the distribution of exponents extending up to at $1 + \theta$ represents an exogenous crossover due to deviations from criticality (see text).



A Shocking Look At...



You Tube

Broadcast Yourself"

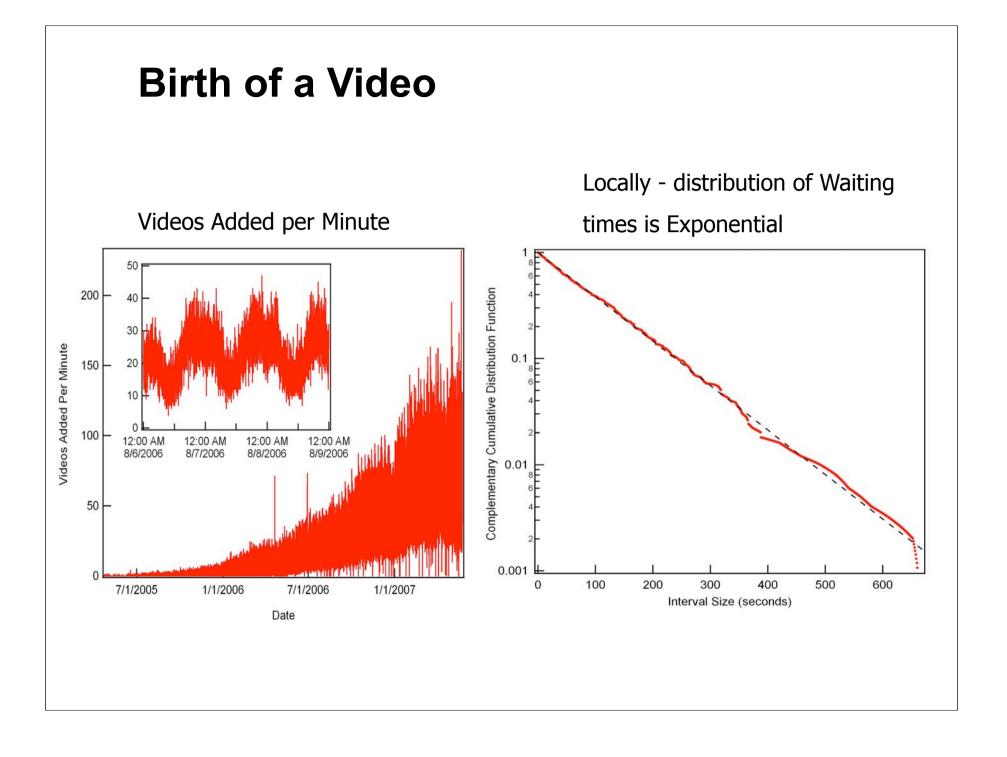


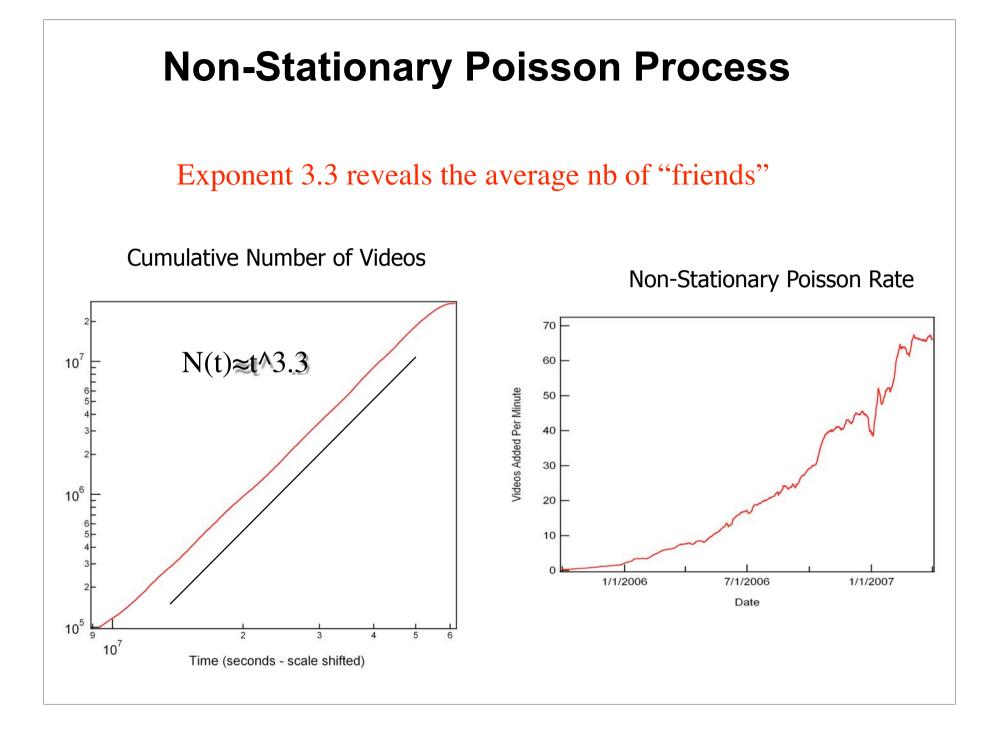
Overview

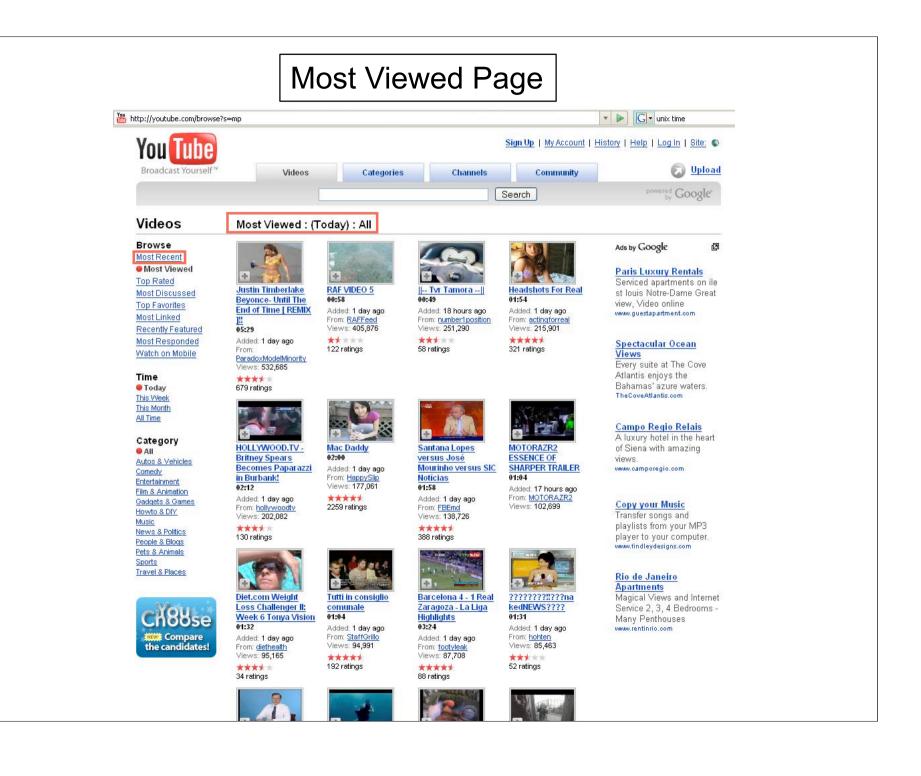
- Video Arrival and Site Growth
- Featuring Endogenous/Exogenous Shocks
- Dynamical Relaxation Following Shocks

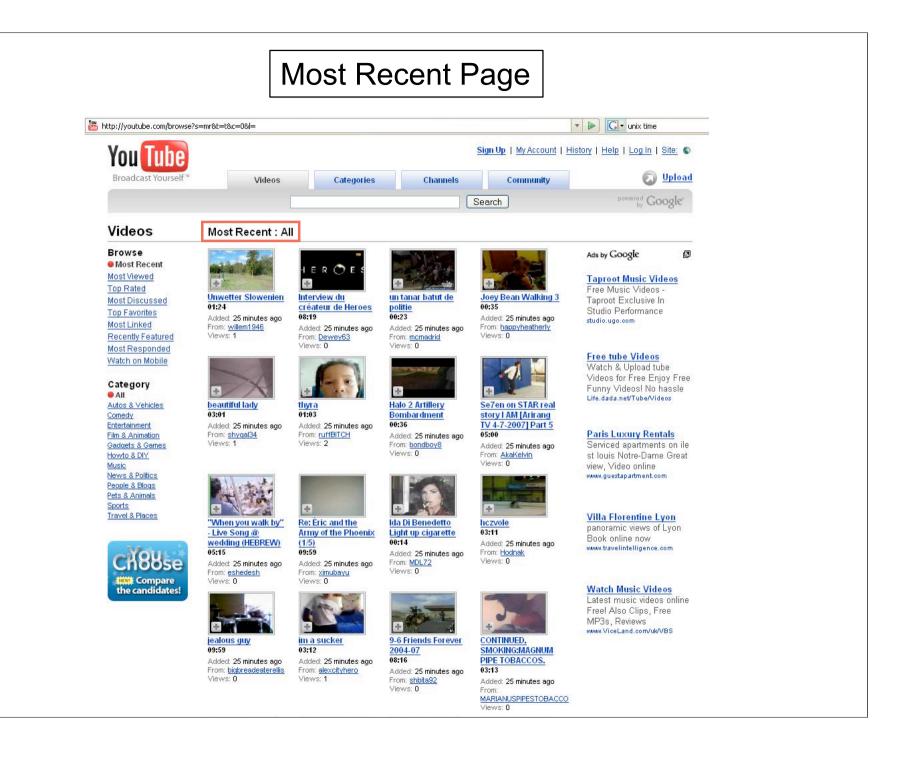
Perl script, via application programming interface (API) for the automated request of data. Stored in MySQL database

YouTube responds with a structured (XML) document containing information such as the cumulative number of times a video has been viewed (dynamic), along with descriptive information (static) concerning the user who posted the video, the title, tags, length, category, rating, comments, etc

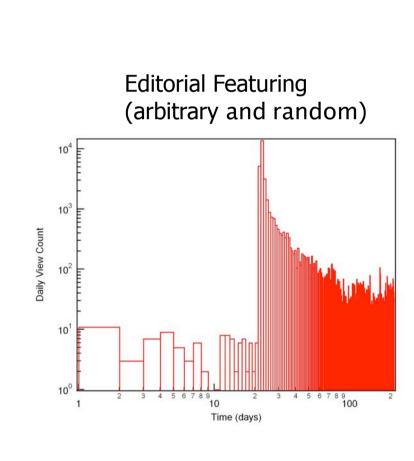




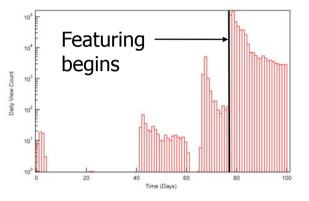




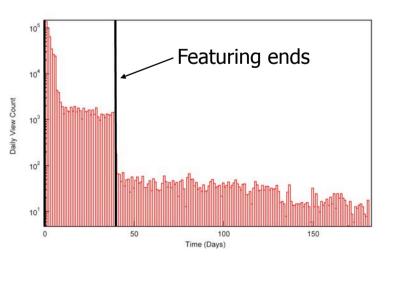
The Effect of Featuring

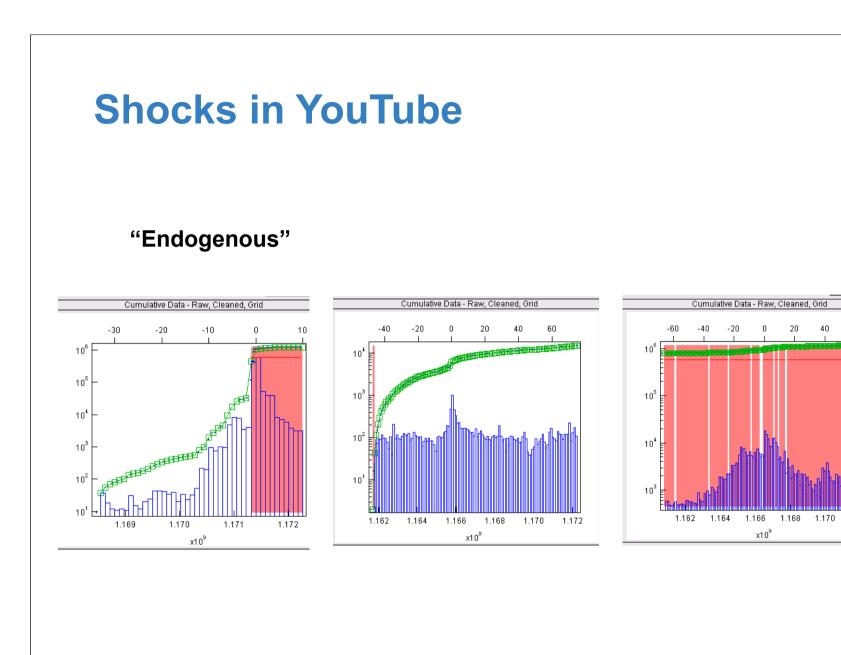


Growth of a Video before being featured

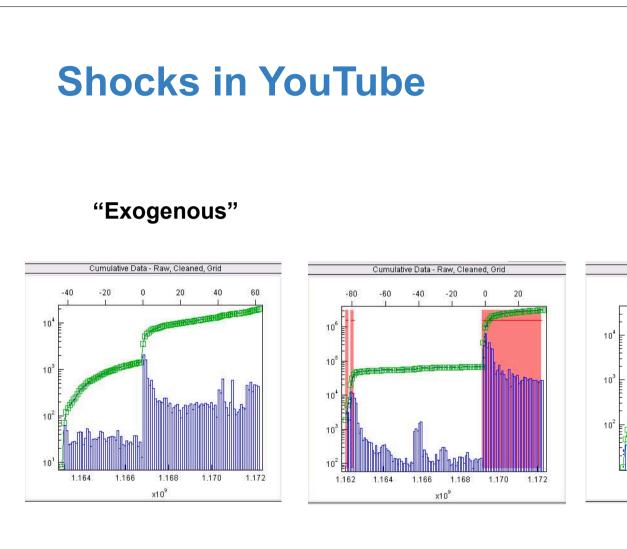


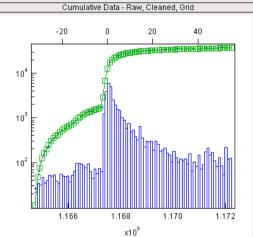
Decline of a video after being featured

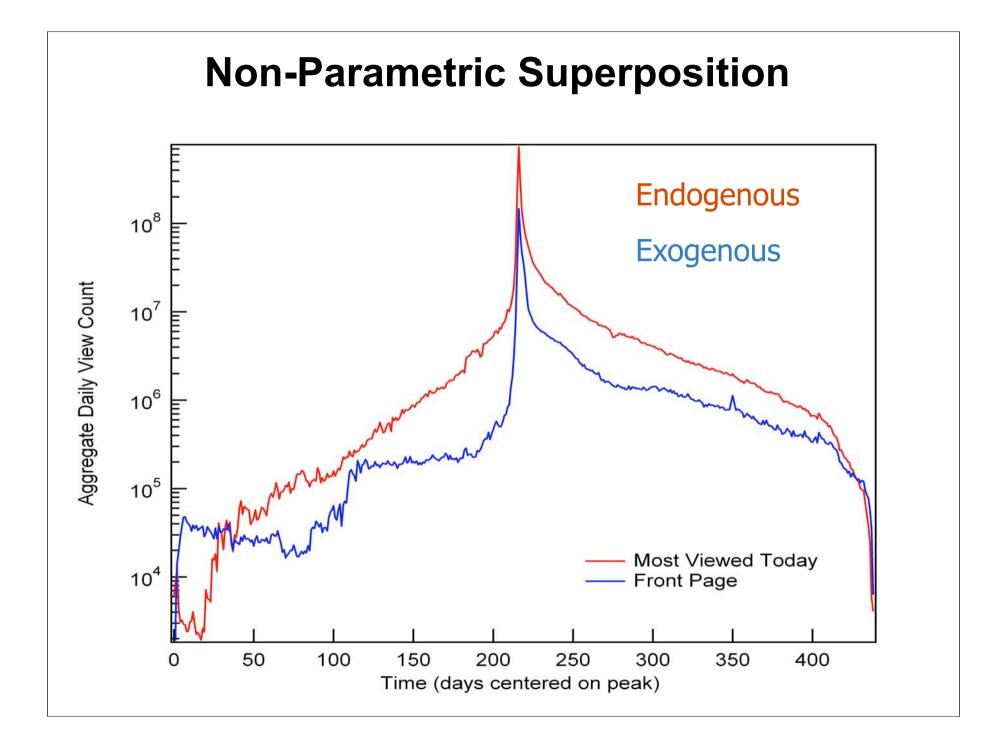


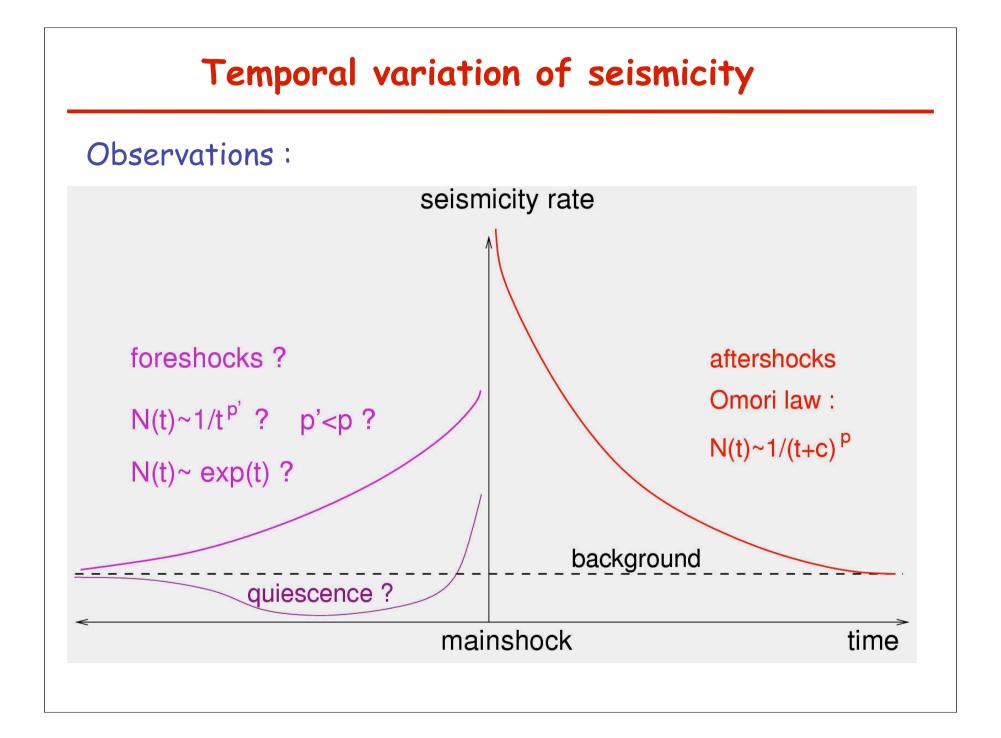


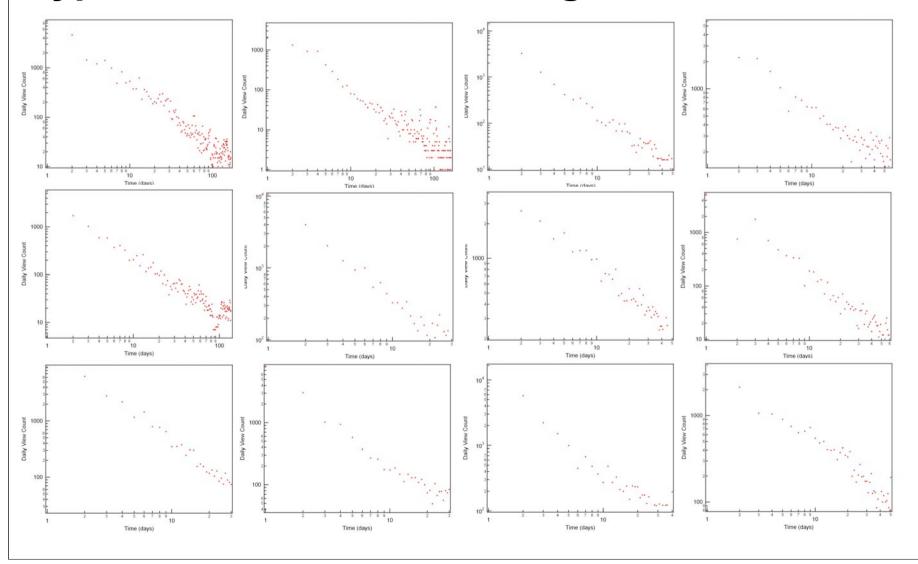
1.172

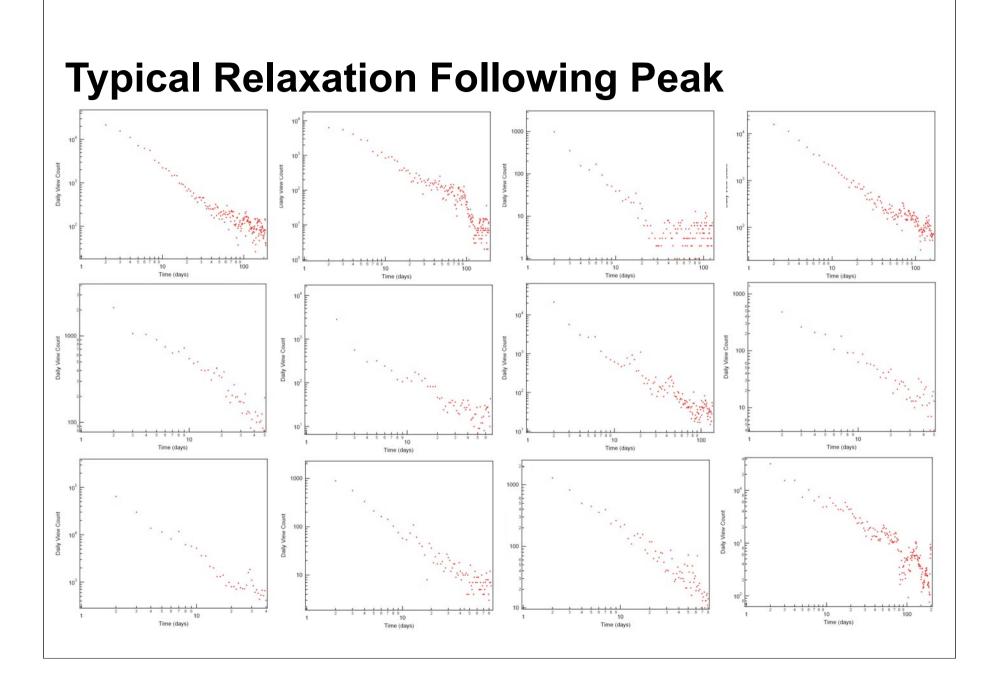


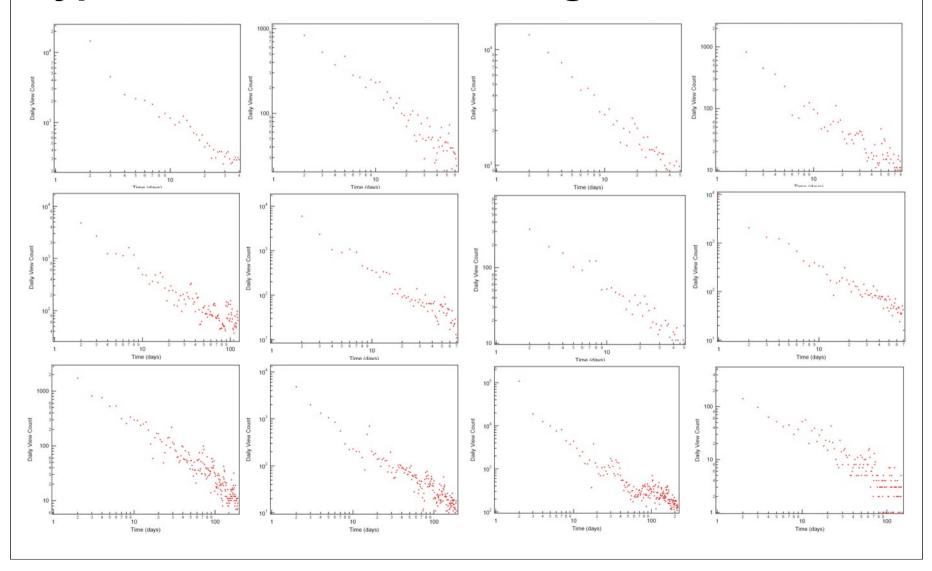


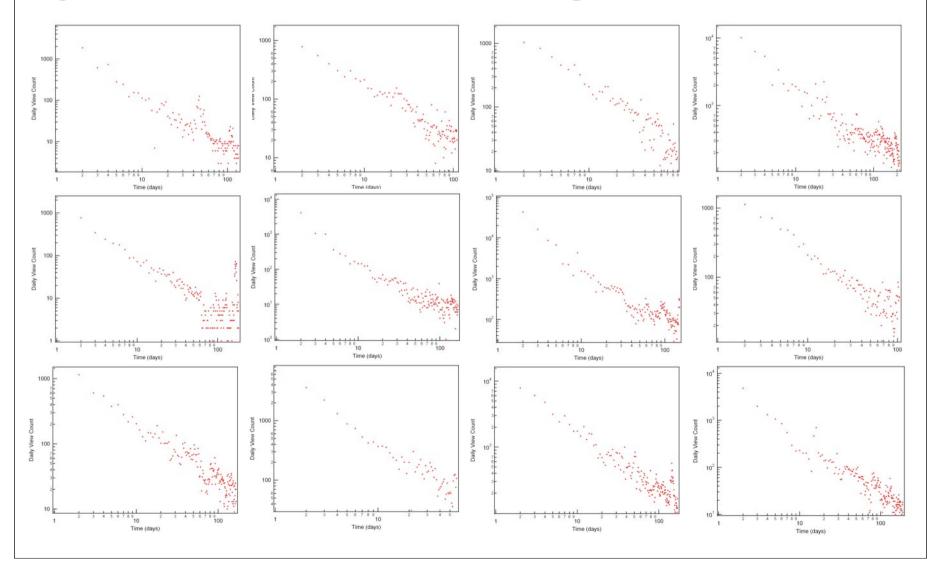


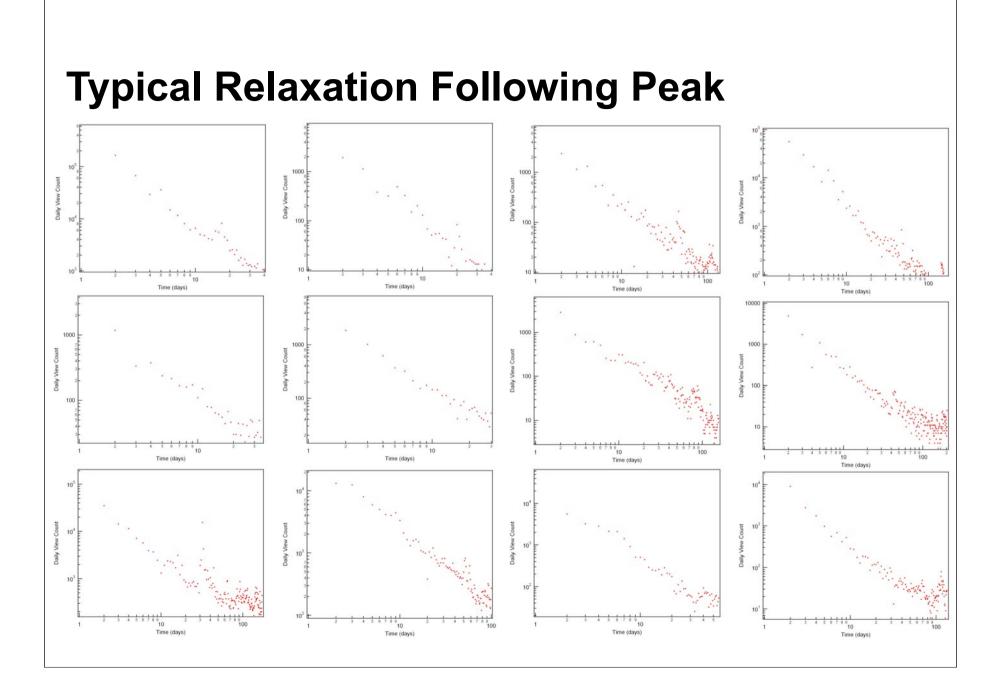


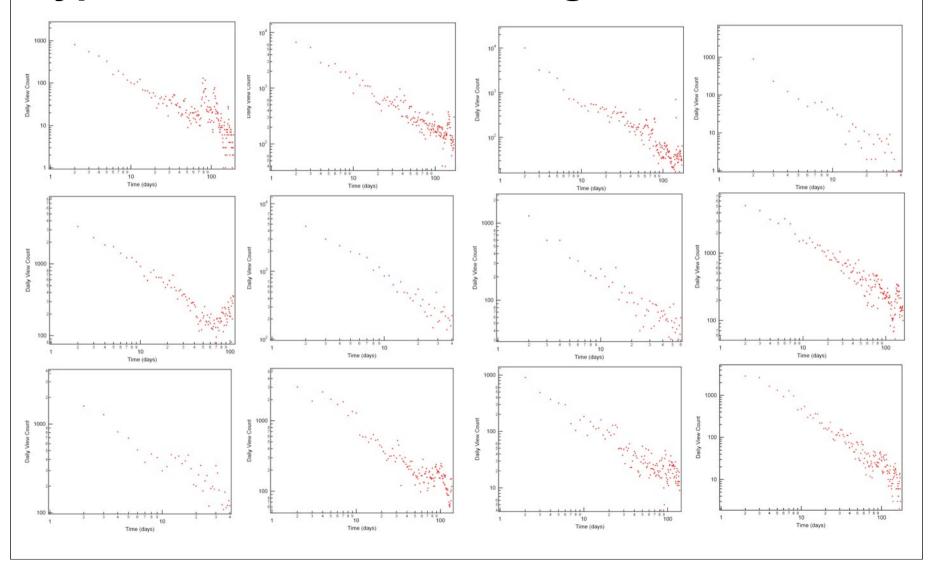






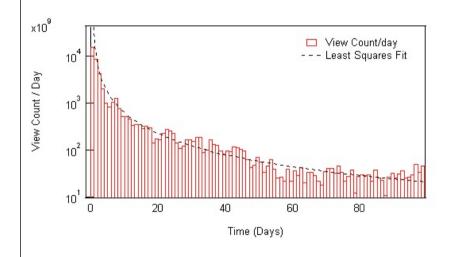






Typical Response

Shock: more than 100 views on a single day, and has at least 10 days following this peak. Of the 5 million videos we are tracking, 76% do not receive 100 views on any given day. Furthermore, 15% either don't have 10 days worth of data, or don't have 10 days following a qualified peak. This leaves us with roughly 9% (=421,487 videos).

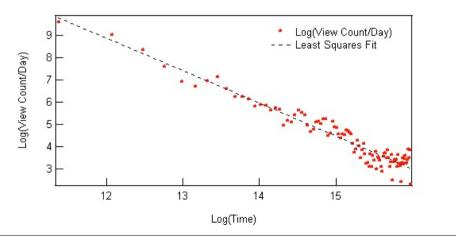


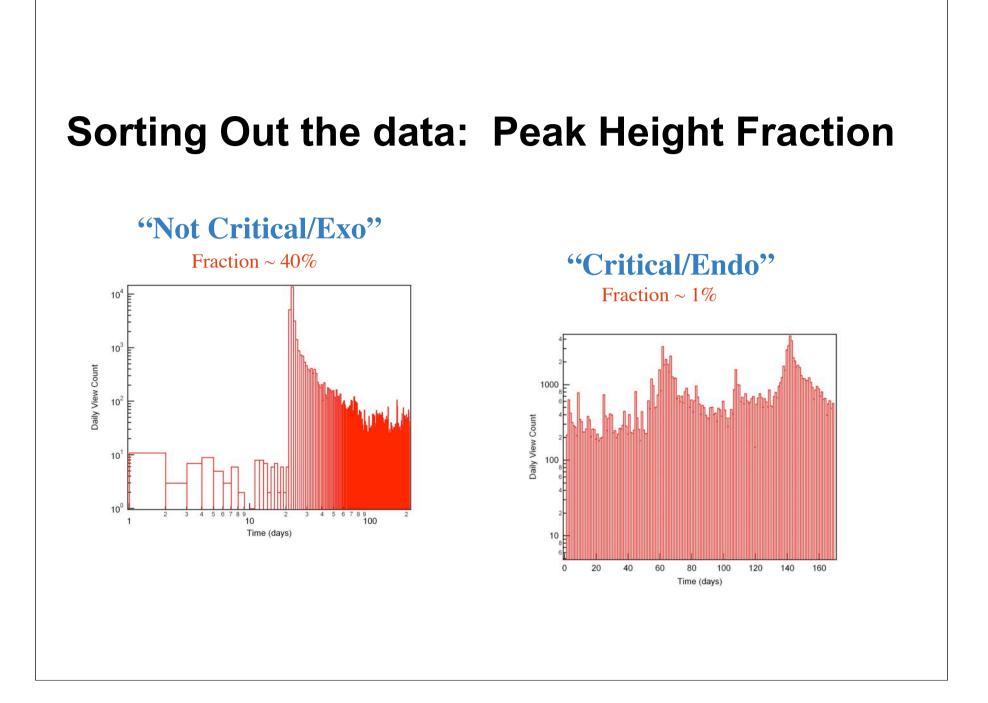
View Count per Day =
$$A(t - t_c)^p$$

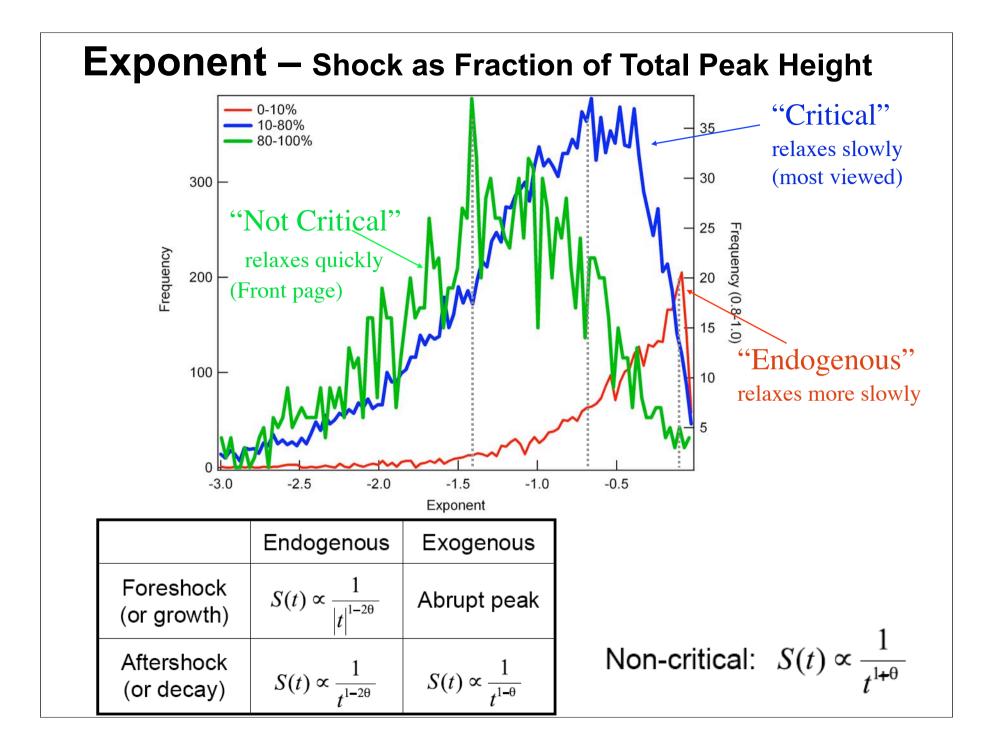
A = Amplitude

p = exponent governing decay

- A Least-Squares Fit is performed on the log-log data over the largest possible range.
- The exponent "p" is extracted







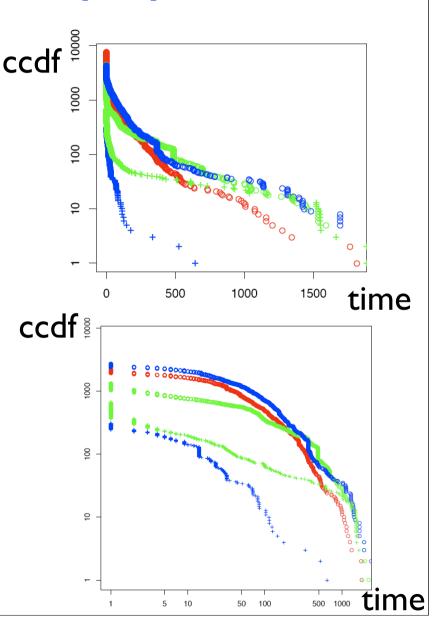
Software vulnerability dynamics

with S. Frei (ETH Zurich)

- vulnerability process is a good proxy of software resilience to bugs
- we identify 4 steps in vulnerability process:

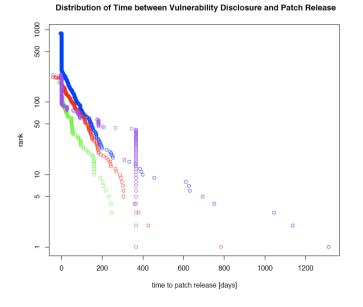
l. discovery (red)
 exploit (green)
 public disclosure (time reference)
 patch release (blue)

- exploits and patch can appear before disclosure (crosses) or after (circles)
- once again, response distribution in this process is heavily tailed
- very characteristic is the distribution of exploits (before disclosure, green crosses) which shows some patterns of power-law with phase transition, in lower tail

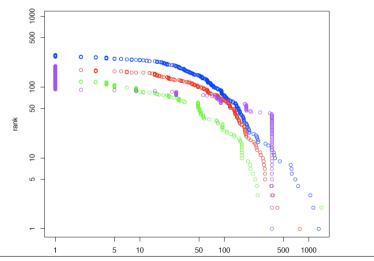


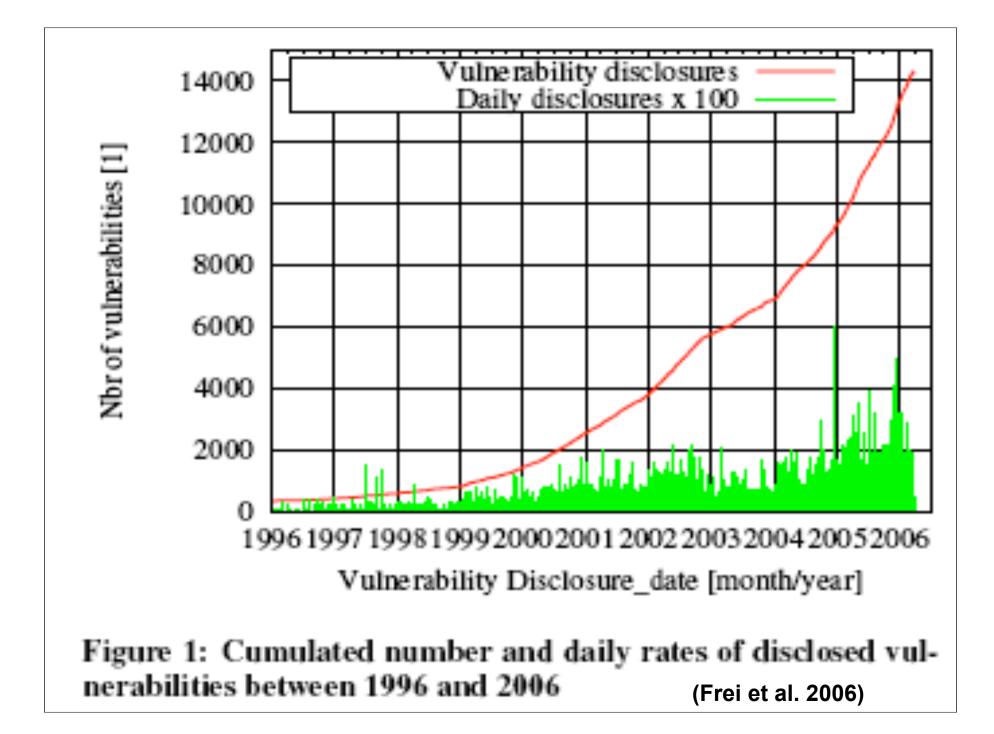
Software vulnerability dynamics

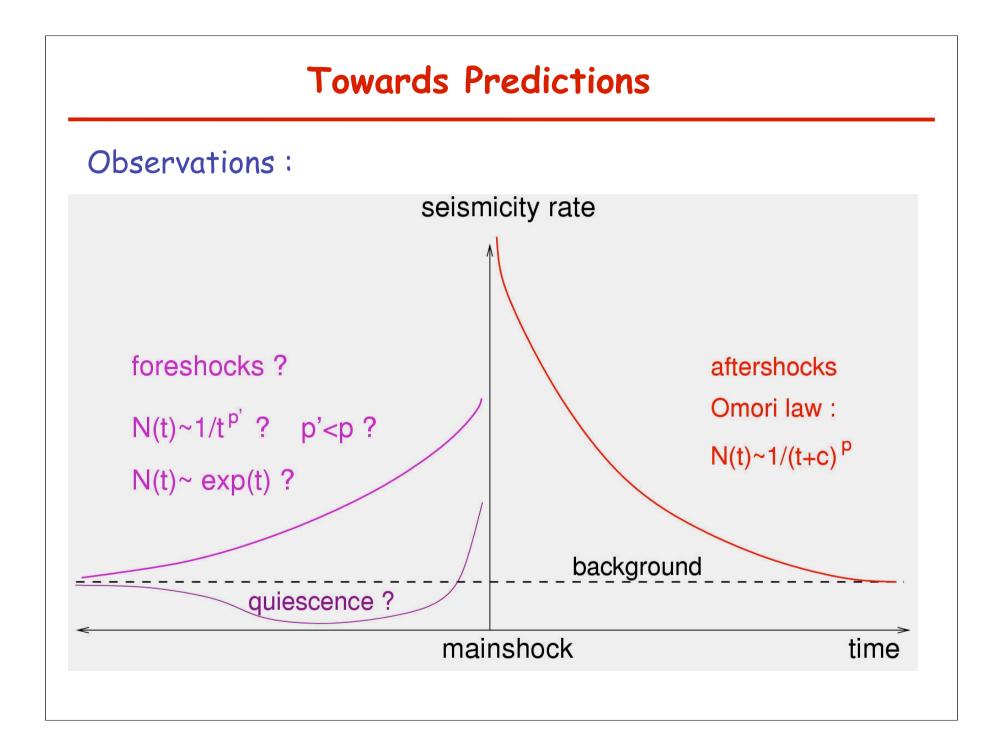
- Here we show comparison between types of softwares:
 - Microsoft (blue)
 Linux (red)
 Oracle (purple)
 Mozilla (green)
- We can see that time to patch distribution is also heavily tailed.
- While it varies differently according to considered software the allure remains somehow the similar, especially when we consider Microsoft (blue) and Linux (red).



Distribution of Time between Vulnerability Disclosure and Patch Release







Predicting the rise and fall of social and economic interactions by monitoring and modeling internet activities and commercial sales

- Books, Music, DVD,
- Electronics (audio and video, cameras and photography, software, computers and video games, cell phones...)
- Office
- Children and Babies
- Home and Garden (which includes pets)
- Gifts, Registries, Jewellery and Watches
- Apparel and Accessories
- Food
- Health, Personal Care, Beauty
- Sports and Outdoors
- Services (movies, restaurants, travel, cars, ...)
- Arts and Hobbies
- Friends and Favourites

Endogenous versus Exogenous

Extinctions

- -meteorite at the Cretaceous/Tertiary KT boundary
- -volcanic eruptions (Deccan traps)
- -self-organized critical events

Financial crashes

- -external shock
- -self-organized instability

Immune system

- -external viral or bacterial attack
- "internal" (dis-)organization

Brain (learning)

- -external inputs
- -internal self-organization and reinforcements (role of sleep)

Aviation industry recession

-September 11, 2001 -structural endogenous problems

Recovery after wars?

-internally generated (civil wars)-externally generated

Volatility bursts in financial time series -external shock -cumulative effect of "small" news

Earthquakes

-tectonic driving

-triggering

Parturition

-mother/foetus triggered? -mother-foetus complex?

Commercial success and sales -Ads

-epidemic network

Social unrests

-triggering factors-rotting of social tissue

Discoveries

-serendipity -maturation Motivation, Effort, Production in Open Source Software

Open Source Software (OSS) Contribution in Mozilla

- data mining in Concurrent Versioning System (CVS)
- analyze software (Mozilla Project) since its start (1998)
- focus on developers, debuggers, contribution

Open Source Software (OSS) Network (with Thomas Maillart)

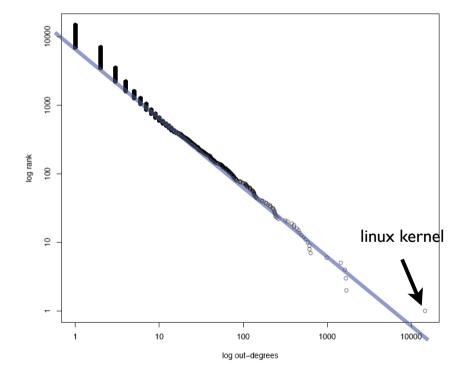
- One Key feature of OSS is the capability to reuse pieces of source code wherever they are useful or needed.
- Programs call other Programs, allowing development time savings, and long range updates
- We can define Links between programs (nodes) as Edges of an OSS Network. These Links are directed.
- We study the connectivity of nodes (out degrees) distribution among a particular OSS subset: packages included in Debian Linux Distribution

Open Source Software (OSS) Network

Debian software

Rank Ordering of out-degree in Package Reuse Network in Debian Distribution

- exponent = IZipf Law
- on four orders of magnitude
- this is the typical pattern of a scale free network

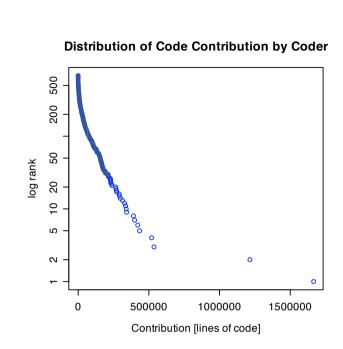


Open Source Software (OSS) Contribution in Mozilla

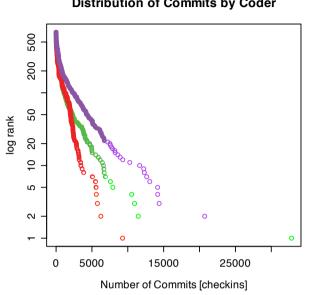
- Since no formal organisation occurs in OSS, we determine the activity of source code committers in time series
- We differentiate by developers (adding features) and debugger (adding robustness)
- We can clearly see that clusters of activity occur in time and space (source code tree)

Open Source Software (OSS) Contribution in Mozilla

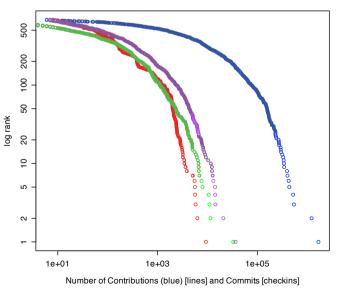
- distribution of contributions (both development and debugging) is heavy tailed
- we analyze distribution of deviations between development and debugging per committer to find coder's profiles.
- Many are developing as much as debugging
- But some develop far much more than debugging
- This distribution is also heavy tailed



- The graphs show 4 distributions: •
- I. checkins in the source code repository (purple)
- 2. contribution in lines of code (blue)
- 3. bugs treated (red)
- 4. difference between checking and bugs (green)
- We can see that all distributions are heavy tailed, denoting a ٠ wide dispersion in coders contributions as well as in their specialization (developers or debuggers)
- From the green graph, we assert that developers mainly also • debug (lower tail) while have a tendency to only develop (upper tail)



Distribution of Contributions (in blue) and Commits by Coder



Distribution of Commits by Coder

Open Source Software (OSS) Contribution in Mozilla - Activity Maps -

- Clusters of activity appear by visual inspection
- Coders tend to work in localized space (source code tree) and time
- Open question: are there cascading effects, source code development ? Intuitevely Yes!
- What are the sources of these cascading effects? exogenous? endogenous? How does source code development process evolves in the life of OSS project?

