



# Code Governance – Governance of Code

Oxford, February 6, 2003

Kei Ishii, Bernd Lutterbeck,  
and Barbara van Schewick



BL1>>>

## Governing the Internet

Governing the Internet is a bit like building houses. May be, this metaphor holds for a while. At any rate, you need something like

- Bricks
- Mortar
- a concept  
and
- a will.

However, houses are static and stable constructions, while the Net is a dynamic setting, where users play a crucial part in it.

Insofar the metaphor may be misleading.

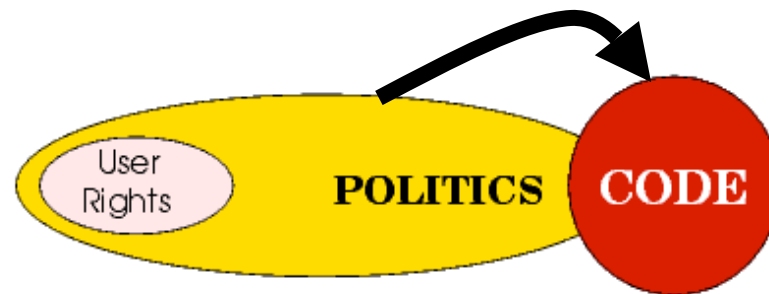
We might be wrong: For us the actual problems with Internet governance are not only the result of the conflicting political interests. It seems to us that they are also the result of thinking in the wrong metaphors.



# Is Architecture Politics?

---

Today's Conference is about ...



**BL>>>2**

## **What today's conference is about**

Implementation of (non-code) policy objectives  
into the code

The academic interests follow mostly the black  
arrow pointing from left to right.

This kind of questioning was certainly sufficient  
for the first generation of the Net.

For the future it is not.



BL>>>3

## The Berlin Group of Internet Governance

In have installed a research group **Internet Governance** at the Technical University.

This talk is a report from our ongoing work, namely the dissertations of Kei Ishii and Barbara van Schewick. Barbara works together with Larry Lessig in Stanford.



# Taking home message

---

- A closer look to
- pure code
- modern economics  
and
- experiences from European integration  
could improve our understanding of politics  
and policies for Internet governance.



**BL>>>4**

## **Kei Ishii's work**

I start with Kei Ishii.

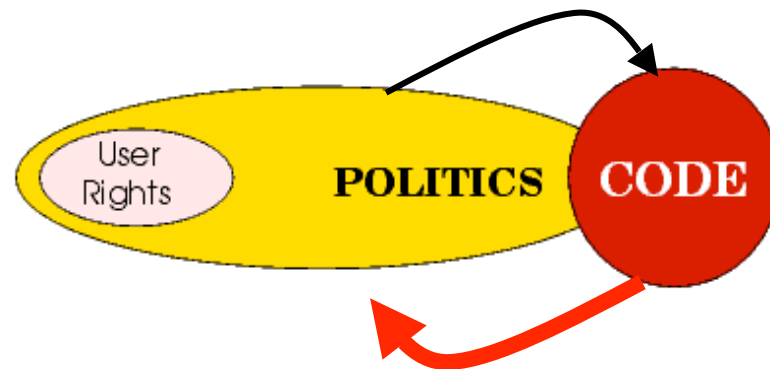
His dissertation concentrates on the red arrow pointing from right to left.



# Extension 1: «There must be more to life...»

---

Today's Conference is about ...





# «Code» is more than a constraint

---

- **Code shapes the governance setting:**
  - its *architecture* imposes constraints on political choices, and opens up other choices,
  - its design & implementation processes are distinct political processes for code governance



# My point today

---

## Code shapes governance settings

...through *architecture*

...through *coding processes*



# The Internet Relay Chat

---

- The IRC is an Internet application in which its users "chat":
  - they exchange short text messages in real time,
  - inside of discussion groups called "channels"
- Created in 1988 by an finnish student
- it currently serves hundreds of thousands of users at any time of day
- Important: The IRC setting is self-governed, and they govern through CODE
- Let's take a look at the IRC network architecture and how it shapes their 'constitution'



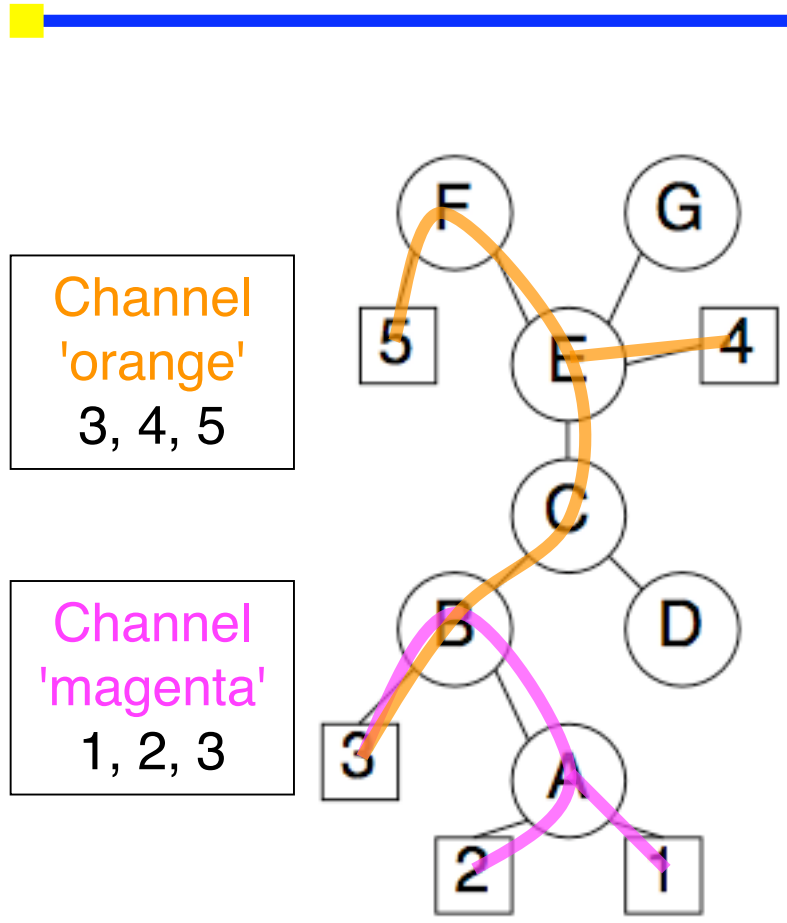
# IRC Architecture

---

- Technical consideration: minimize bandwidth use
  - Tree Topology
  - Duplication of global state in every knode
- Result: all user data can be sent in a most bandwidth-efficient way, because every server knows where exactly to send which data
- Governance consequence: a bias towards bottom-up



# IRC: Architecture and Implications



Channel  
'orange'  
3, 4, 5

Channel  
'magenta'  
1, 2, 3

Global state in every server

Minimum bandwidth use

No central root server  
All (hub) servers are 'equal'

Governance Bias:  
**Bottom-up**



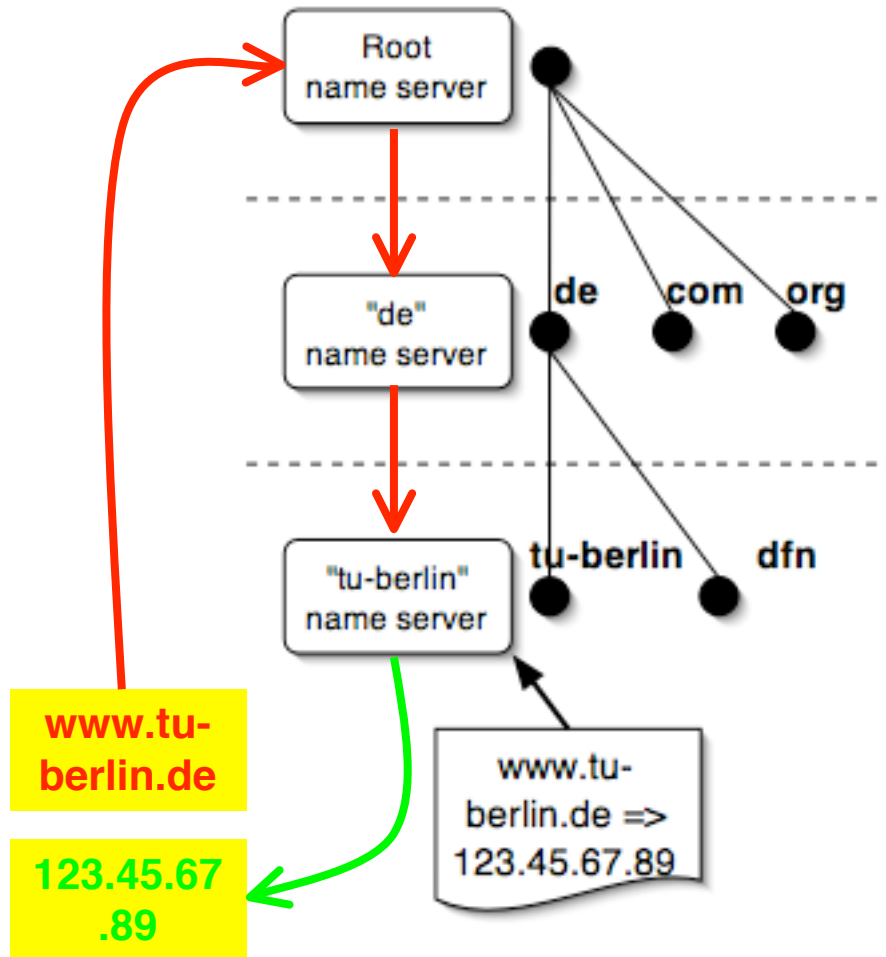
# Comparison to DNS

---

- Technical consideration: Decentralize data administration
  - Tree topology
- Data distributed (not duplicated in every node like in IRC)
- Governance consequence: top-down



# DNS: Architecture and Implications



'Local' servers only

decentral administration

Top-Bottom routing

Governance Bias:

**Top-down**

# Cooperation in IRC

---

- So the principal technical architecture of the Internet Relay Chat induces a more or less 'cooperative' governance between the server administrators.
- How does this work out?





# Open server vs. Closed Server Dispute

---

- 2nd year of IRC (1990), the question arose who should be allowed to connect a server to the IRC network?
- Two fraction: Open servers (anyone can connect) vs. closed servers (only qualified servers)
- After heated (and nasty) dispute, a group took initiative: Proclaimed net forking
- Two networks: Anet (open), EFnet (closed)
- In order to preserve split, EFnet installed the Q-line
- If a server connects to an open server, it is q-lined.
- Must be installed on each server; q-lined server must be listed in every server (otherwise network breaks)
- The net result: Anet vanished, EFnet prevailed

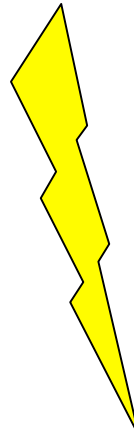


# IRC: Open Server vs. Closed Server

---

**Question:** Open or Closed server network?

«Open Server» side:  
**Anarchynet (Anet)**



«Closed Server» side:  
**"Eris-free" net (EFnet)**

Code change:  
**Quarantine line**  
(q-line)

**Result:** Anet vanished; EFnet prevailed

## Lessons: Technical governance innovations

---

- Code disputes are governance disputes
- There was a consensus-forming discussion
- Q-line *changes the architectures*: as new technical governance innovation
- In this example: only one prevailed.
- In others, both nets prevailed with different code policies: both code gov alternatives worked out
- ==>>>
- Intricate design & implementation processes to shape setting and to resolve conflicts
- So let me reiterate my point today:



# My point today (reiterated)

---

Code shapes governance settings

...through *architecture*:

Code imposes governance choices

...through *coding processes*:

Technical innovations are governance innovations



# BL>>>5

**I continue with the approach of Barbara van Schewick**

Her core question is:

- What kind of economic structures for the production of software do we need to make innovation in the governance of the Net happen?
- One of her answers is: We need at least another bubble in the figure.



# Extension 2: Architecture is more than just politics

---

But Architecture is about ...



[Barbara van Schewick 2003]

**BL>>>6**

## **Results from economy**

Let me sum up briefly some results of recent findings in political economy and empirical research in economics.

These results reflect also some of our Berlin empirical research on open source software development.



# Innovation -- some findings of Economics

---

Recent evidence in Economics suggests that private property regimes and monetary incentives are not necessarily conditions for innovation.

Instead, patterns of reciprocal altruism, emergence of reputational systems, and intrinsic motivation can lead to alternative settings.

(See: Fehr/Gächter 2002; Ostrom 1990; Ostrom/Hess 2001; Frey 2002)





BL>>>7

## Consequences for governing the Net

These findings could be extremely valuable for the design of the governance principles of the next generation Internet.

I beg your pardon: Lack of time leads me to express them like a woodcut.



# Governing code – «the mortar»

---

## Principle 1

The building principles are evolutionary.

## Principle 2

The property regime has influence on the cohesion of «the bricks». In many cases code build on «commons» will do better work.

## Principle 3

The less heteronomous the action the better are the results.



# Let Code Governance Structures Evolve!

---

## Principle 1

**Ex ante values in technology are short-lived. Technological systems which have to cope with differing interests have to go through many versions in order to find a viable structure.**



# Innovation needs open competition!

---

## Principle 2

**Open up competition through open  
technical standards and open organization  
forms**

*See: Proposal (Draft) for a CHARTER ON SUSTAINABLE  
KNOWLEDGE SOCIETIES, World Summit on the  
Information Society (WSIS), Geneva, December 2003*



# Governing code – lessons from European integration

---

## Principle 3

Governing code could mean changing things the «Jean-Monnet-way»:

- Don't *prescribe* the final state
- Instead let competition work inside a given governance structure
- Strict enforcement of the fundamental freedoms

