Olive

One-click Execution of Internet-Archived Software

Mahadev Satyanarayanan

School of Computer Science

Carnegie Mellon University

Supported by:

April 2011 Workshop at IBM

Computer Scientists and Digital Library Community

Benjamin Gilbert, CMU

Catherine Zhang, IBM

Chris Shillum, Elsevier

Claire Cocco, OCLC

Clifford Lynch, CNI

Doug Freimuth, IBM

Erika Linke, CMU

Giovanni Pacifici, IBM

Glenn Ammons, IBM

Gloriana St Clair, CMU

Jeff Brody, IBM

Joan Smith, Emory

Kevin Guthrie, ITHAKA

M Satyanarayanan (Satya), CMU

Maria Ebling, IBM

Michael Witt, Purdue

Mike Handy, Library of Congress

Mike Lesk, Rutgers

Mladen Vouk, NCSU

Naomi Nelson, Duke

Patrick Drehrer, RENCI

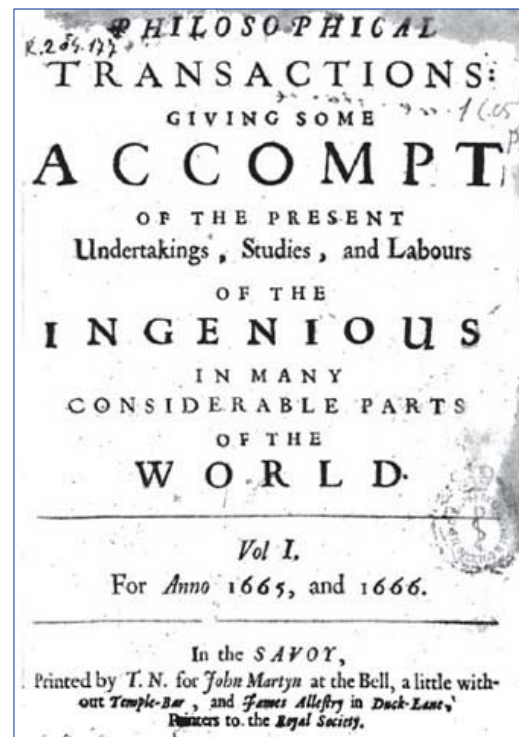Paul Jones, Ibiblio.org

Peng Ning, NCSU

Steve Griffin, Library of Congress
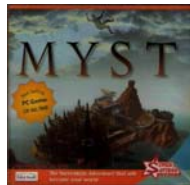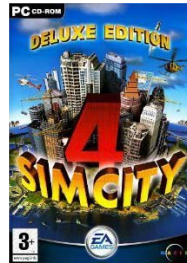
Tamar Eilam, IBM

Timothy Dilauro, Johns Hopkins

Vasanth Bala (Vas), IBM

Vijay Kumar, MIT

Archiving Static Content

What About Executable Content?
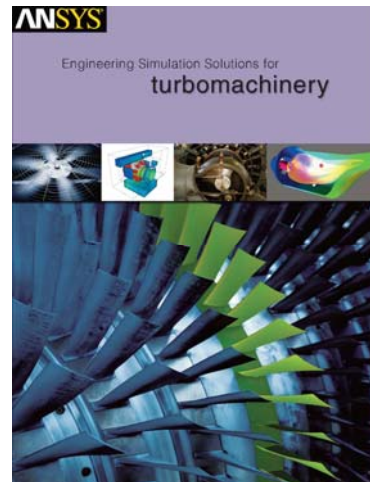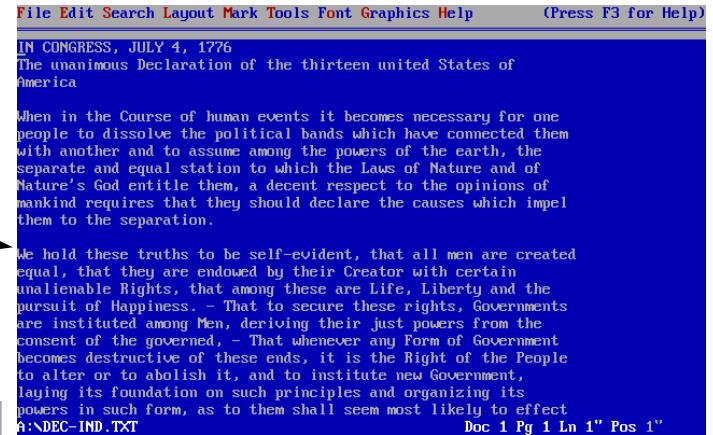

Games

Application-specific content

WordPerfect 1.0 doc
Can you read it today?
100 years from now?

Simulation model
Can you re-run old model with new data?
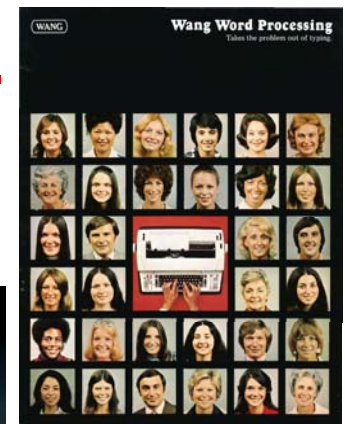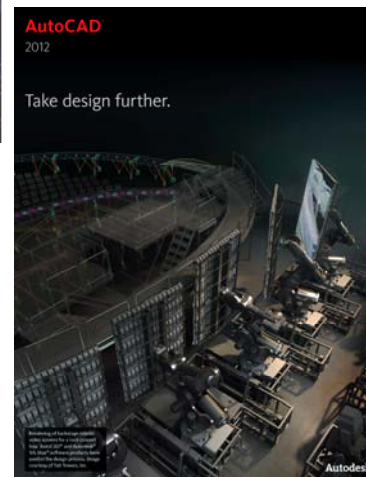
Original Wang doc
Can you read it today?
100 years from now?

Execution Fidelity

Ability to precisely reproduce execution

- **many moving parts**
  - hardware, operating system, dynamically linked libraries, configuration parameters, language settings, timezone settings, …
- **very difficult to achieve execution fidelity**

Problem faced by every software vendor

- **large market ⇒ configuration flexibility ⇒ complexity**
- **results in *high time to value* for customer**

Exactly this problem is faced in archiving executable content

- **no help desk for 100-year old software!**
- **how do we fake the existence of now-obsolete environments?**

Computer Museum

For hardware
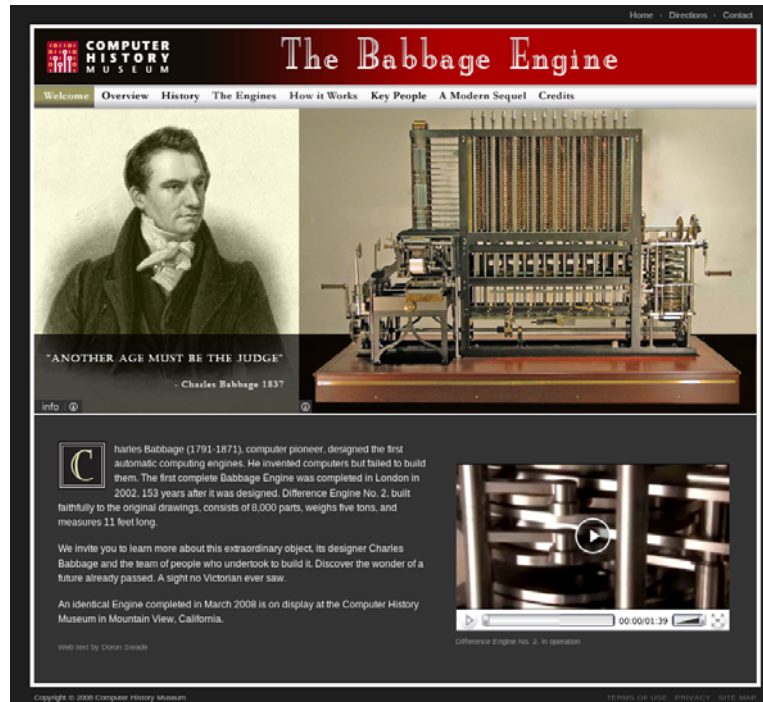
For software: ??????

**30-year functional life
How to test patches?**

Solution: Take It With You

Transform the problem into a scaling problem

- **pack up and carry the entire environment with you**
  (including the OS)

- **transitive closure of everything you need**

Central idea of a (hardware) *virtual machine (VM)*

VMs Can Relieve Pain

VM library of pre-tested software combinations

- use VMs as the basis of software distribution
- outsource configuration pain to third parties
- value to both software vendor and customer

User- and enterprise-specific data outside VM (e.g. dist. file sys.)

Broader value: *archive executable content*

- reproducibility for scientific results
- executable embodiment of archival results (e.g. SIGGRAPH)
- escrow SaaS (software as a service) code as insurance
- *… many others …*

> **VMs can do for software what VLSI did for hardware**
>
> **(hide complexity and increase robustness)**

But VMs are Huge!

10 GB VM

- **@ 100 Mbps → at least 800 seconds (13 minutes) download**

- **@ 10 Mbps → at least 8000 seconds (over two hours) download**

No one will wait that long to look at something briefly!

How do we achieve quick launch?

Video Streaming
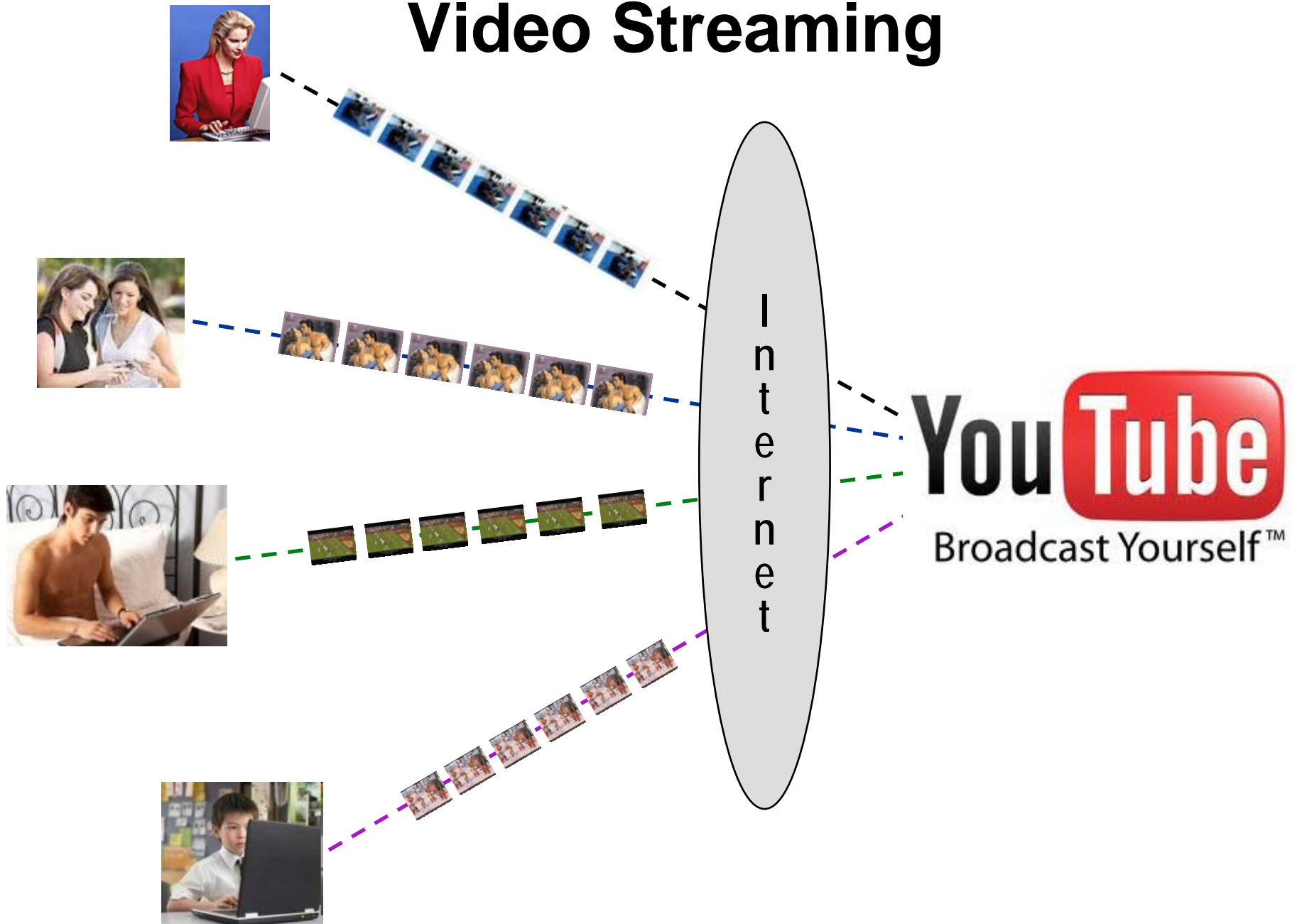
VM Streaming Not So Easy

Access to VM image is not linear

Reference pattern depends on many runtime factors

- data dependencies

- human interaction

- spatial and temporal locality (program behavior)

Borrow an old idea from operating systems

- *demand paging*

- intercept missing VM pieces and fetch over Internet

- prefetching may speed up performance if predictions good

VM-based Cloud Computing Today

VM Storage

Local / Cloud

VM Execution

Cloud / Local

local-cloud   **cloud-cloud**

Remote execution
(e.g. Grid Computing)

Managed execution
(e.g. Amazon EC2)

Classic PC model
(e.g. laptops)

Transient PC
(e.g. ISR)

local-local   **cloud-local**

better availability   *better safety*

more comute power

crisper interaction

Transient PC Model
(≅ AFS++)

Cloud

VM changes

Suspend

Resume

Suspend

Resume

Transient PC #1

Transient PC #2

"Cyber Dishwasher"

Frankfurt Airport

Transient PC #3

Olive Vision

Open archive of VM images accessible Internet-wide

Workshop clarified Olive concepts and value proposition

- **clean separation of locations for**
    - **VM image archiving**
    - **instance creation**

- **ability to compose VMs (e.g. for licensing reasons)**

Many Olive concepts explored and fleshed out

Global Olive

Few instances
1 to 10 worldwide

Cloud Olive

Numerous instances
10 to 1000 worldwide

One or few per
Fortune 500 enterprise

Edge Olive

Myriad instances
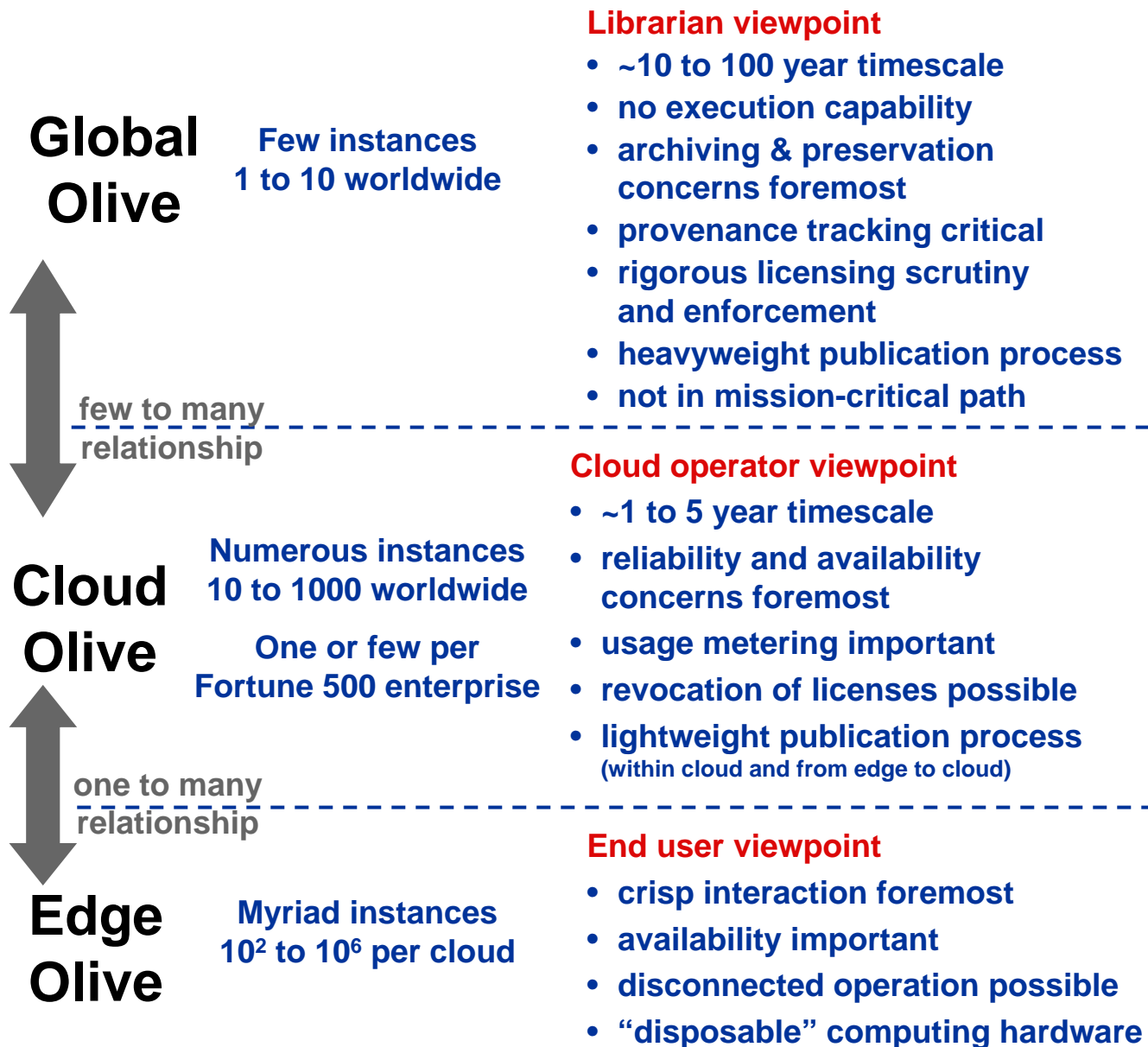10^2 to 10^6 per cloud

few to many
relationship

one to many
relationship

Librarian viewpoint
- **~10 to 100 year timescale**
- **no execution capability**
- **archiving & preservation concerns foremost**
- **provenance tracking critical**
- **rigorous licensing scrutiny and enforcement**
- **heavyweight publication process**
- **not in mission-critical path**

Cloud operator viewpoint
- **~1 to 5 year timescale**
- **reliability and availability concerns foremost**
- **usage metering important**
- **revocation of licenses possible**
- **lightweight publication process (within cloud and from edge to cloud)**

End user viewpoint
- **crisp interaction foremost**
- **availability important**
- **disconnected operation possible**
- **"disposable" computing hardware**

Olive Demo

(work in progress)