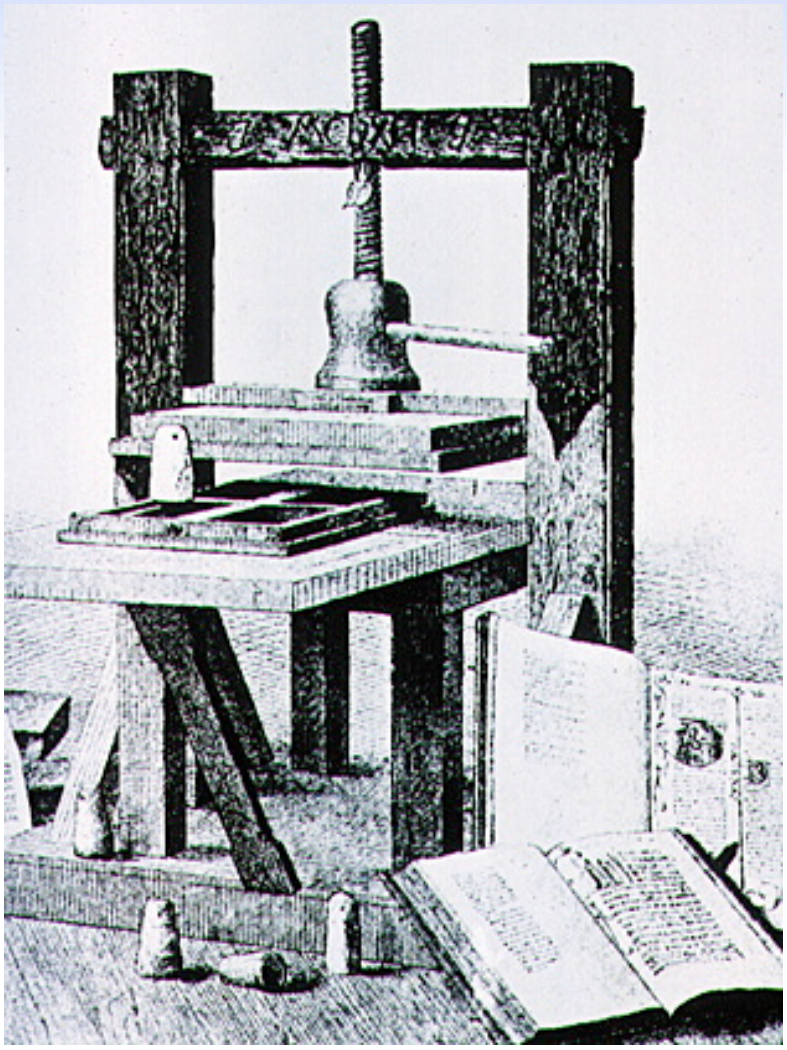
An Open Printing Environment

Barak Zahavy
Academic Information Systems
Columbia University

October 21, 2004

IT From A Higher Vantage Point



Printing at Colleges and Universities

- Creating “Limits or rules” for printing is a top way schools can cut IT costs (*Chronicle for Higher Education*, 10/4/02*)

* http://chronicle.com/prm/weekly/v49/i06/06a03901.htm

Outline

- Why a printing system?
- Columbia's model
- Initial System: *Jake*
- Improved System: *NINJa*
- *NINJa*: Example Case
- *NINJa*: Benefits
- Questions/Comments

Why do we need it?

- Develop a printing system to:
  - Reduce waste and abuse
  - Regulate printing cost
  - Provide reliable monitoring
  - Provide accurate statistics

Printing at Columbia University

- Academic Information Systems (AcIS) maintains printers for student computing facilities and libraries
- AcIS supports 80+ printers
- 16,000 unique users printed 8,000,000 pages 9/1/03-8/31/04

Model

- A release station kiosk atop each printer displays the printer's queue
- User sends job to be printed
- User selects a job from the queue and authenticates
- User is authenticated and authorized to print
- Job is printed, printing account deducted

What does it look like?

Initial System: “Jake”

- Configuration
  - Thin-client release station and multiple Unix print servers
  - Central print servers used lpd to handle print jobs
  - Windows and Mac support enabled via Samba and Netatalk
- Key features
  - Used central unique user IDs for authentication and authorization
  - Page quota system
  - Simplex/duplex printing option at the kiosk
  - Secured jobs for authenticated clients
  - No licensing fees, locally written
  - Used 1991 thru 2002

Initial System (Jake): Limitations

- Groups of printers had single points of failure
- No true database model
- Diagnosing problems was overly complex
- System had problems printing from modern applications and operating systems

… Because of these limitations, the system did not scale efficiently

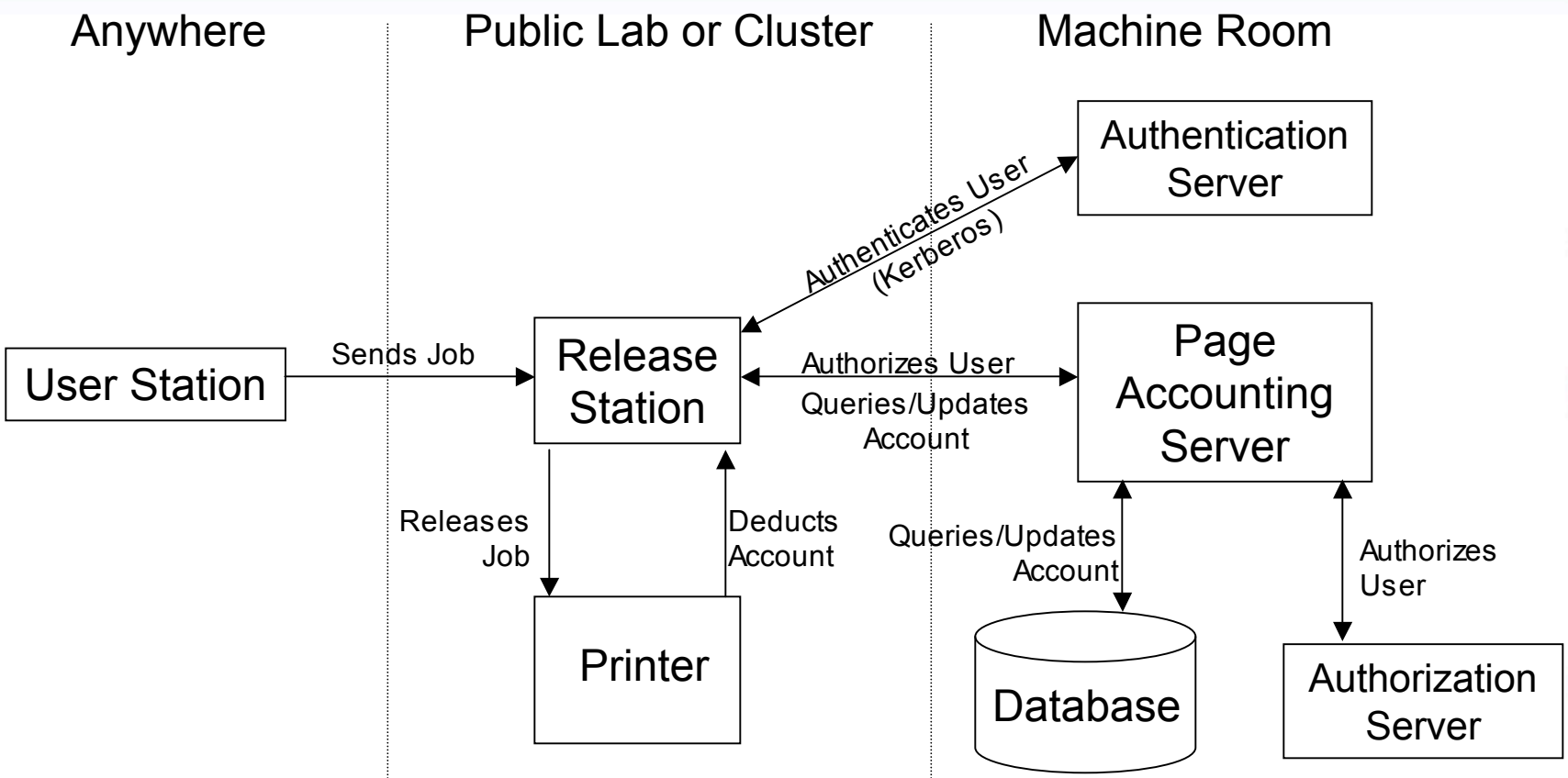
Goals of improved system

- Minimize points of failure
- Build a distributed model
- Database backend
- Easily maintained
- Design to be pluggable and use open standards
- Support credit card online transactions
- Scalable to 100+ printers

“NINJa” Printing System Design Goals

- Distributed model: each NINJa station is a print server in itself
- Uses PXE network boot technology -- downloads images from a server via tftp
- Uses a page control server for page accounting
- SQL database backend
- Open source, open standards
- Central Kerberos authentication and LDAP authorization
- Support credit card purchases via a secure web server

NINJa Printing System: Model



NINJa: User Station

- Windows, Mac, Linux, Solaris
- Wired or Wireless network
- Campus run or personally owned
- All major applications

NINJa: Printer

- Hewlett-Packard 4000 and 8000 series
- Postscript
- HP JetDirect: network interface and security
- HP JetAdmin: administration

NINJa: Print Release Station/Server

- OS independent - written in Java
- Stripped down Remote Boot Linux station, with X-windows, LPRng, Kerberos, and Java
- Display is Customizable HTML
- Authenticates via Kerberos 5
- Communicates with separate page counting server (sends username and page request)
- Queues and releases print jobs to printer

NINJa: Page Accounting System

- Page quotas based on unique user ID authorization and printer types
- Weekly or semesterly quotas (flexible)
- Purchase of additional pages (which do not expire)
- Distinction between quota pages and purchased (non-expiring) pages
- Supports a pay-per-page model

NINJa: Page Accounting System (Cont'd)

- Pluggable authorization module via open standard identification system (e.g. LDAP)
- Page transactions for sales, refunds, credits, and queries
- Suite of tools for transactions, queries, and reporting

NINJa: Database

- Stores print usage
- Stores page sales transactions
- Data model that organizes printers, quotas, departments, pricing, and affiliations
- Portable SQL (Oracle)
- Seamless failover to backup

Deployment at Columbia

- 80+ printers maintained
- Minimum support (Besides supplies and repair)
- Usage for 9/1/03 to 8/31/04:
  - 16,000 users
  - 1,650,000 print jobs
  - 8,000,000 pages
  - $40,000 in sales

Statistics Generation

Printing Demographic Breakdown: 1/1/04 thru 5/31/04

Primary University Affil	Unique Individuals Printing	Total Jobs	Total Pages Printed	Unique Individuals Who Bought Pages	Total Pages Sold	Gross Dollars Income*
Faculty	271	10,150	65,597	10	8,650	$865.00
Staff	454	13,130	63,974	17	4,340	$434.00
Grad Student	3,582	135,943	738,266	346	74,780	$7,478.00
Undergrad Student	8,495	589,546	2,913,251	338	53,246	$5,324.60
Professional Student	2,468	46,329	195,338	127	25,830	$2,583.00
Other Student	192	4,296	13,371	8	450	$45.00
Grace Period Student	122	3,776	14,424	24	8,100	$810.00
Alum	80	2,731	10,121	64	13,650	$1,365.00
Other	8	2,121	3,704	2	800	$80.00
No Match With Affil File	132	2,955	11,712	75	12,270	$1,227.00
	15,804	***810,977***	***4,029,758***	***1,011***	***202,116***	***$20,211.60***

Case study: Implementing NINJa in the Libraries

	'02-'03	'03-'04
AcIS	4,000,000	6,000,000
Libraries	6,500,000	2,000,000
Total	10,500,000	8,000,000

(Pages Printed)

- Difference: 2.5 million pages @ $.02/page = $50,000
- Revenue '03-'04: $58,000 in page sales and account upgrades
- Costs lowered by $108,000

Benefits

- Regulates printing expenses
- Provides accurate statistics
- Provides reliable monitoring
- Reduces waste
- Utilizes central ID system
- Performance scaling and reliability via a distributed model

Recap

- Reasons for having a printing system
- Initial system and its limitations
- NINJa:
  - Goals
  - Design
  - Implementation example
  - Benefits

Thanks!

- http://www.columbia.edu/acis/dev/projects/ninja
- Questions?