Visualizing Collections of Internet Archives

John Berlin, Joel Rodriguez, Slobodan Milanko

Motivation

- Current solutions are limiting; small focus
  - Unable to effectively retrieve mementos without knowing exact URI-M
- Archivers are unable to visually experience collections
  - Lack of resource evolution
  - Lack of comparison
  - Lack of statistics
- User driven, clutter free, design for moderate collections
- Archivers desire to visually share collections with each other

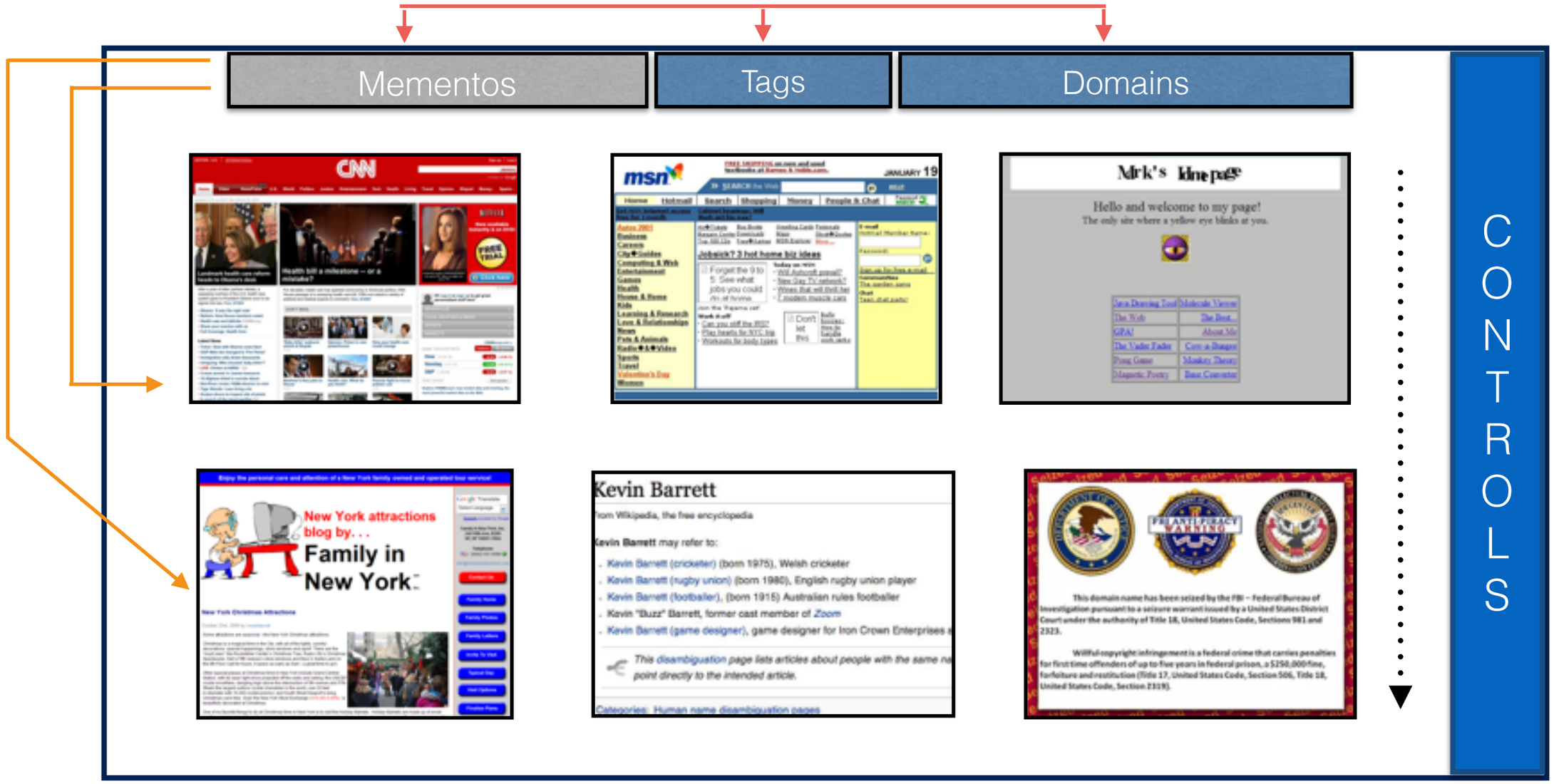
Dataset Description

- We're abstracting the user's archive collection as a table
- Key attribute (URI) that map to categorical or ordered values
  - Archiver - Categorical
  - Tags - Categorical
  - Number of archives - Quantitative
  - URI-M - Categorical
  - Time/date - Ordered
- The table approach
  - Facilitates ordering
  - Filtering of data

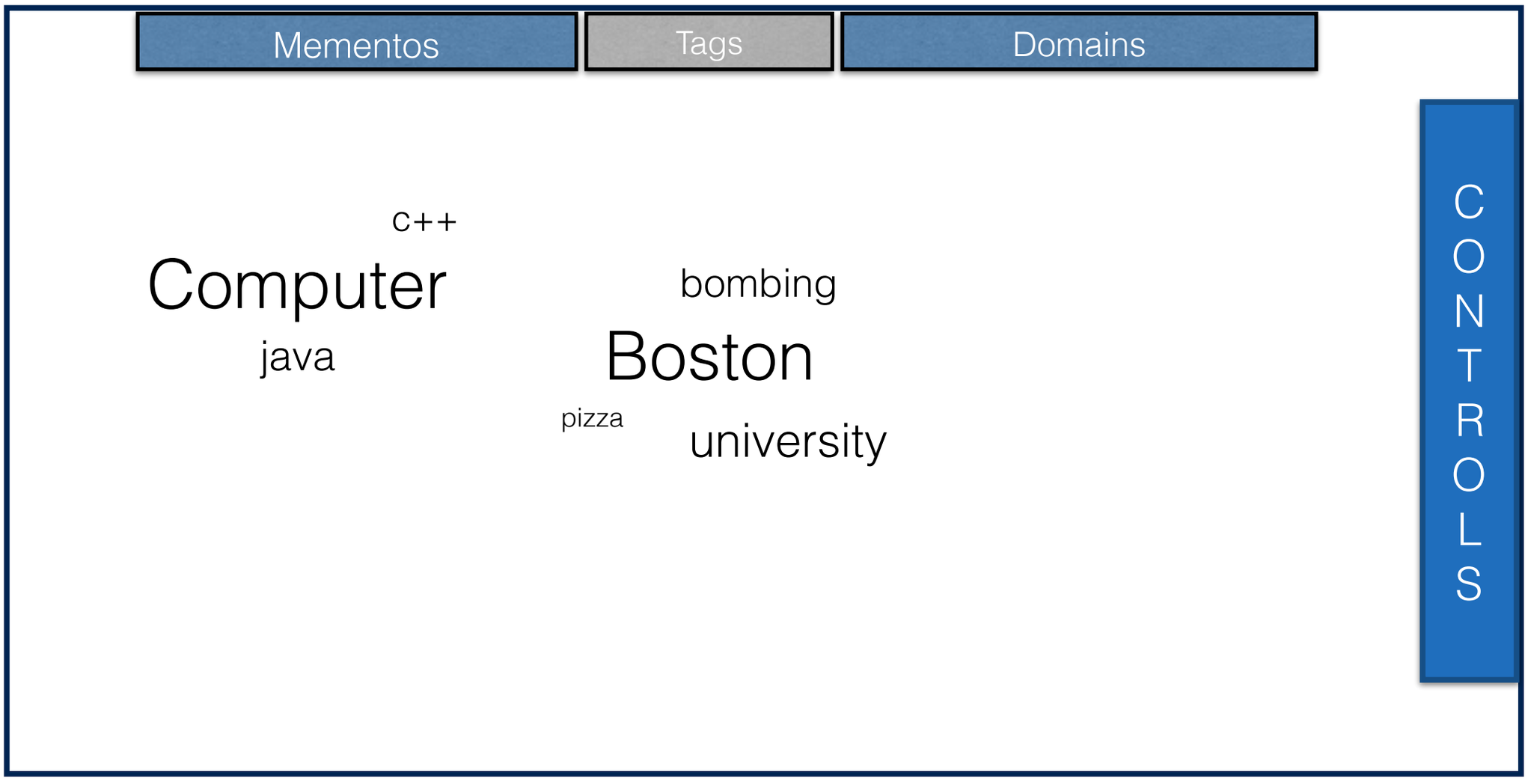
Abstract Tasks

- Lookup, browse, explore and locate can be achieved easily with the table abstraction.
- Users start with an overview first, zoom and filter later, and retrieve details on demand
- Users will be able to identify, compare and summarize different archive records
- Analysis through discovery, presentation, and enjoyment
- Observe interesting characteristics, trends and relationships

Potential Design

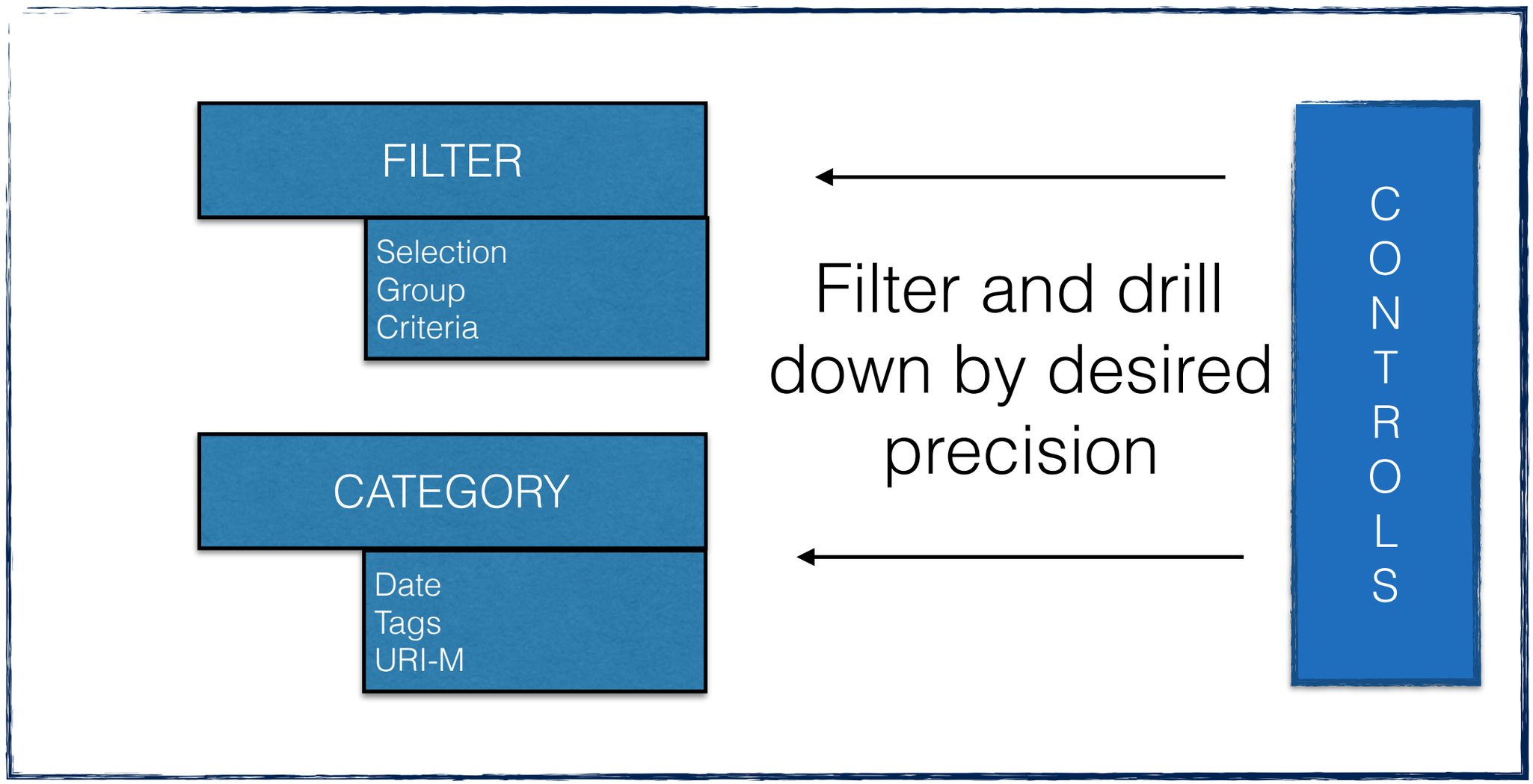


Continued: Potential Design



Continued: Potential Design

User Driven Control



Detail

www.msn.com/news/june

June 22, 2001

www.msn.com/news/june

June 22, 2005

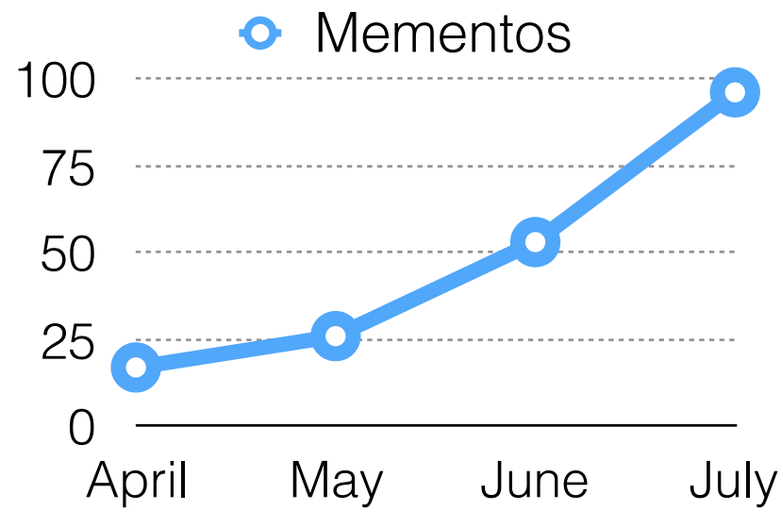
www.msn.com/news/may

May 22, 2005

BACK

Detail: From Domain

tags



TODO::: other charts from email

Needed Tools

- Mink, WARCreate
  - Create sample archives for local testing
- Wayback Machine (Local/Remote)
  - Get an understanding for look of archives
- WAIL
  - Local archiving and hosting
- PhantomJS (snapshots) and D3

Group Roles

- Entire Group
  - Code - Develop a working product
  - Test - Ensure visualization accepts dynamic data
  - Survey for feedback
  - Create demo
- Dan & Joel
  - Create final presentation
- John
  - Final presenter of solution

Estimation

- 4-6 weeks to develop
- 1 week to demo and test
- Weekly feedback to ensure satisfaction