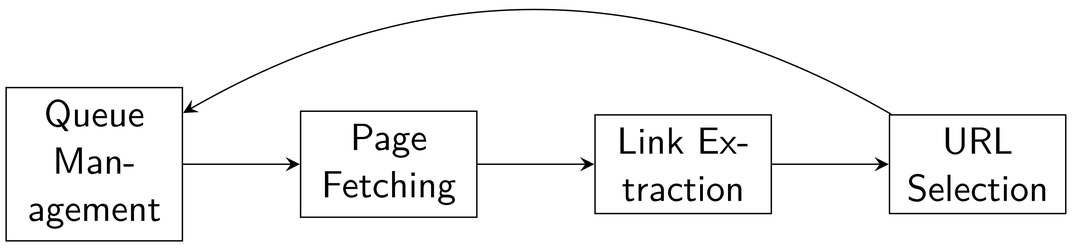
Demonstrating Intelligent Crawling and Archiving of Web Applications

Muhammad Faheem
Institut Mines–Télécom
Télécom ParisTech; CNRS LTCI
Paris, France
muhammad.faheem@telecom-paristech.fr

Pierre Senellart
Télécom ParisTech
& The University of Hong Kong
Hong Kong
pierre.senellart@telecom-paristech.fr

Traditional crawler

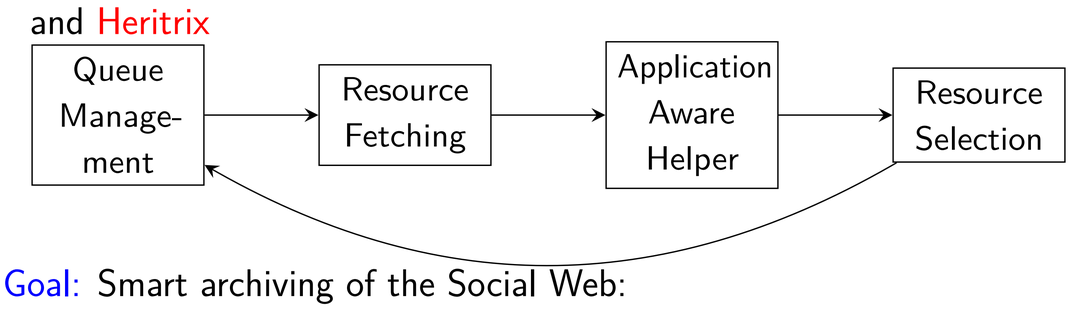
Traditional crawling: independent of the nature of the sites and their content management system



$\Longrightarrow$ Many HTTP requests, no guarantee of content quality

Application-aware helper

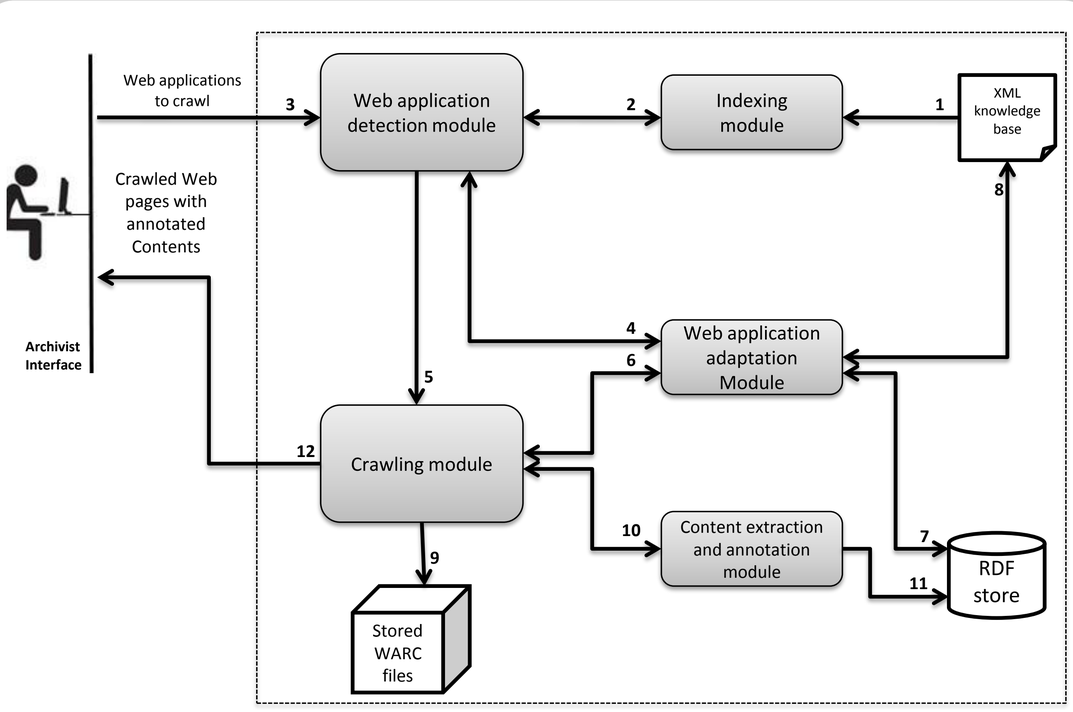
- Different crawling techniques for different Web sites
- Detect the type of Web application, kind of Web pages inside this Web application, and decide crawling actions accordingly
- Directly targets useful content-rich areas, avoids archive redundancy, and enriches the archive with semantic description of the content
- Implemented in 2 Web crawlers: Internet Memory Foundation crawler and Heritrix



Goal: Smart archiving of the Social Web:

1. Performing intelligent Crawling
2. Archiving Web objects

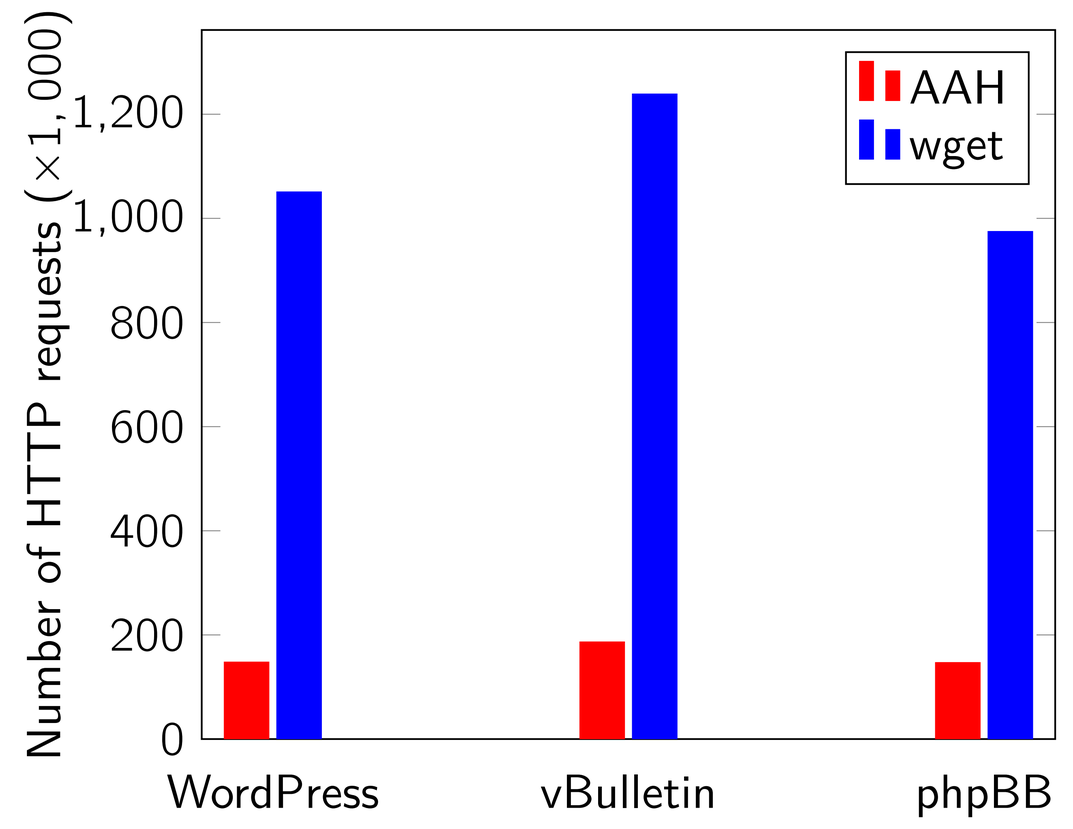
Architecture



Methodology

- Knowledge base of known Web application types, algorithms for flexible and adaptive matching of Web applications to these types
  - Declarative, XML-based format
  - Integrated with YFilter for efficient indexing of KB.
- Type detected using URL patterns, HTTP metadata, textual content, XPath patterns, etc. E.g., vBulletin Web forum: `contains(//script/@src,'vbulletin_global.js')`
- Different crawling actions for different kinds of Web pages under a specific Web application
- Crawling action: not just a list of URLs; can be any action that uses REST API, complicated interaction with AJAX-based application, and extracts semantic Web objects

Crawl efficiency



Crawl effectiveness