Distributed Web Crawler Architecture
Introduction

• Distributed web crawler architecture is made up of work items (URLs to be accessed), a duplicate request detector, and a callback module

• Web crawlers are divided into crawler (generates work items) and fetcher modules (fetches work items)

• “Directed” crawlers: each crawler targets a specific type of web pages (eg. blog pages, real time news)
Work Item Service

• Intermediary work item service
  • receives work items from crawlers
  • queues them in work queues
  • each sends work items to fetchers periodically
  • fetcher sends fetched web page back to work item service
  • these web pages are sent back to crawlers present in callbacks list (URL list where each URL is mapped to target web crawlers)
  • if the URL associated with a work item (from crawler A) is already present in a work queue, callback is created with address of A so that when the web page is fetched, data is returned to A as well
• Work item service comprises of:
  • work items monitor: detects incoming work items
  • callback module: holds information about source web crawler for each work item (URL: address pairs)
• duplicate requests detector: doesn’t queue repeat requests, callback is updated instead
Bucket Service

• work items mapped to buckets based on URL/IP
• overcomes issue of overloading single web server (IP) by avoiding repetitive requests by different fetchers (in case two distinct domains have overlapping IPs)
• fetchers poll buckets to pick up work items; queues (associated with buckets) release work items at predetermined frequencies
• each bucket is associated with its own work queue