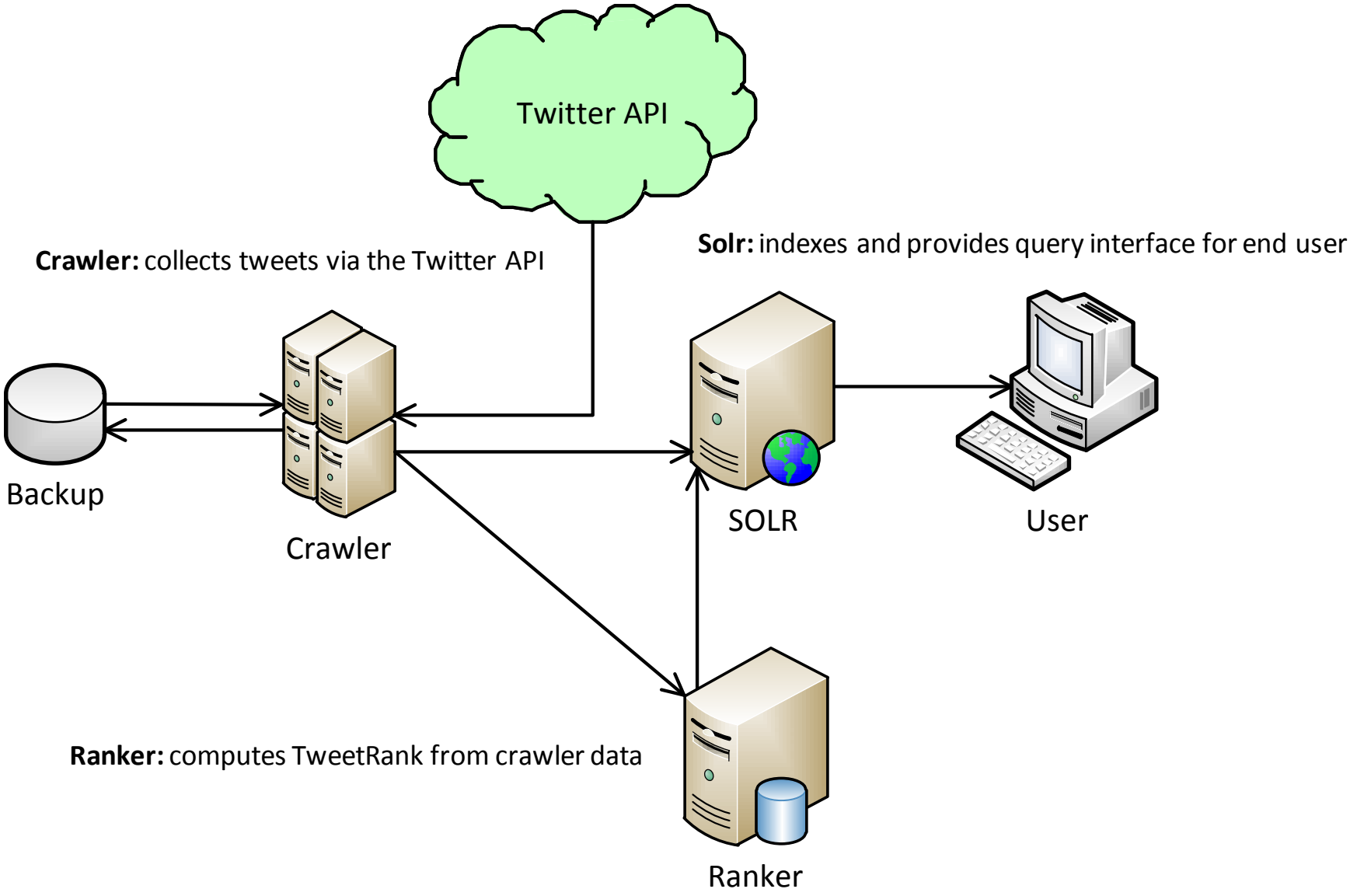
TweetRank

TweetRank is an attempt to apply the PageRank algorithm on Twitter statuses (tweets).

It uses a different rank calculation which considers attributes such as number of replies/retweets and hash tags.

Victor Hallberg	victorha@kth.se
Johan Stjernberg	stjer@kth.se
Joan Puigcerver I Perez	joanpip@kth.se
Alexander Hjalmarsson	alehja@kth.se
Christoffer Rydberg	chrryd@kth.se

Components

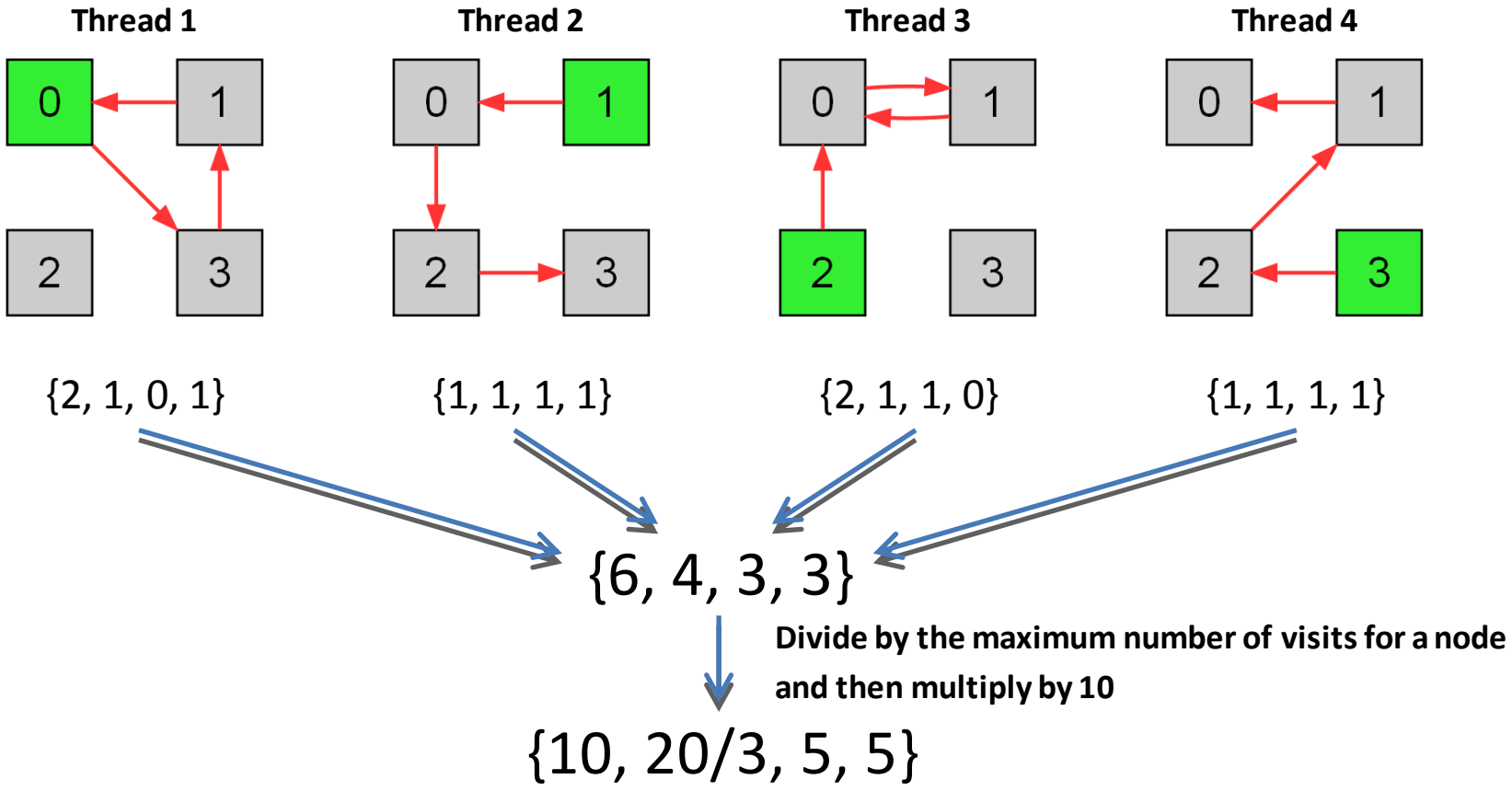

Twitter API
Crawler: collects tweets via the Twitter API
Solr: indexes and provides query interface for end user
Backup
Crawler
SOLR
User
Ranker: computes TweetRank from crawler data
Ranker

Crawler

- Uses the Twitter *HTTP REST* API
  - Twitter limits the number of queries to 150 per hour
  - Crawler gathers as much data as possible from each query
  - Use multiple proxies to bypass the query limit
  - Runs on multiple threads in multiple machines
- How does it work?
  1. Query the tweets and friends from the first user in a queue of users
  2. Add friends and user mentions in each tweet to the user queue

Ranker

- Uses the complete path *Monte Carlo* algorithm, stopping at dangling nodes
- Starts a randomized walk from each node at least 100 times (at most total tweets / 100)
- Random path length - 20% chance at each node that the surfer stops
- Ranker runs on multiple threads, where each thread computes one walk at a time
- Final rank is calculated as the normalized sum of visits for each node from every walk:



Solr / Lucene

- Handles indexing and searching.
- Crawler sends tweets to be indexed by Solr through *HTTP POST* requests (in XML format)
- Current TweetRank data is fetched from a text file on the server
  - Enables rank updates without having to replace (re-index) existing documents
  - Utilizes the *ExternalFileField* format in Solr
- Scores for individual statuses are calculated as a product of:
  - TweetTrank
  - *tf-idf* for terms, where matches against hash tags are boosted

Results