3nuts

A Top-Down Peer-to-Peer Network

Overview

DELIS

Dynamically Evolving, Large-scale Information Systems

Information Society Technologies

University of Paderborn - Computer Science Department - Christian Schindelhauer

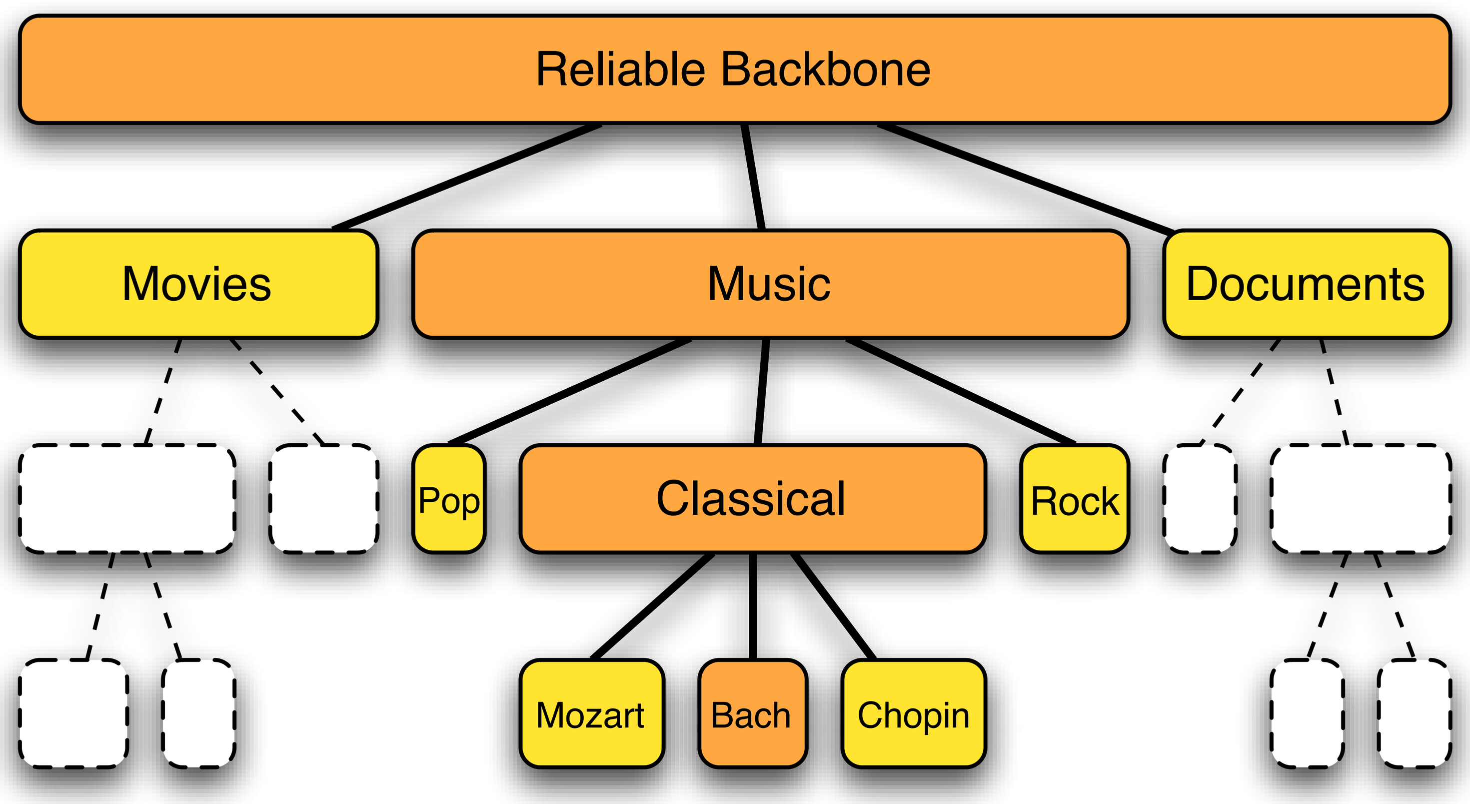
Abstract

3nuts is a Peer-to-Peer network combining the benefits of reliable random graphs and semantic search trees. The goal of 3nuts is to overcome the restricted query languages induced by the use of distributed hash tables. In many currently used peer-to-peer networks these distributed hash tables do not support the efficient exploration of the semantic neighborhood of a data entry. In our top-down approach semantic relationships are preserved by 3nuts' semantic search tree. There, peers are assigned to the data and not vice versa.

3nuts allows nontrivial lookups, like range queries, neighborhood search, etc. All network operations in 3nuts are local and distributed, i.e. simple handshake operations maintain the network structure. Besides this, 3nuts provides fair load balancing, fast data access and guaranteed robustness, proved by rigorous analysis.

Network Structure

Random graphs are 3nuts' main component to ensure robustness. Therefore, every node of the semantic search tree constitutes a random graph. A random graph has several advantages over deterministic network structures, e.g. small maintenance costs, small diameter, small degree, expansion property, flexibility.

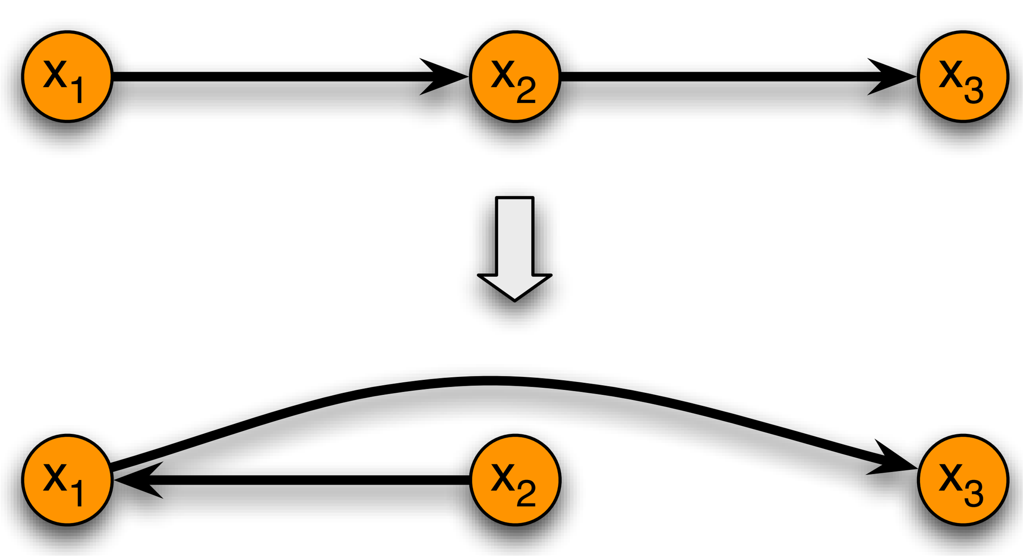


In 3nuts each node of the desired search tree is represented by a random connected multi-digraph with constant degree. Starting with the Reliable Backbone containing all the peers as the root of the search tree, the peers are recursively assigned to the subtrees of the current node. This process is continued until there is only a single peer left or the node represents a leaf of the search tree. Therefore, a peer describes a path from the root to a leaf of the search tree and participates in each of the random graphs representing the corresponding nodes. To allow routing within $O(\log n)$ hops, each peer maintains so called shortcut links to a random peer of each node neighboring its own path.

Random Graph Maintenance

In dynamic networks it is mandatory to maintain the network topology. This is also the case for random graphs. In [1] we introduce a simple operation called **Push&Pull** to maintain truly random multi-digraphs. A Push&Pull operation does not need more resources than the necessary checking a peer's neighborhood. Albeit its simplicity, the Push&Pull operation delivers a uniform probability distribution over the set of connected out-regular multi-digraphs.

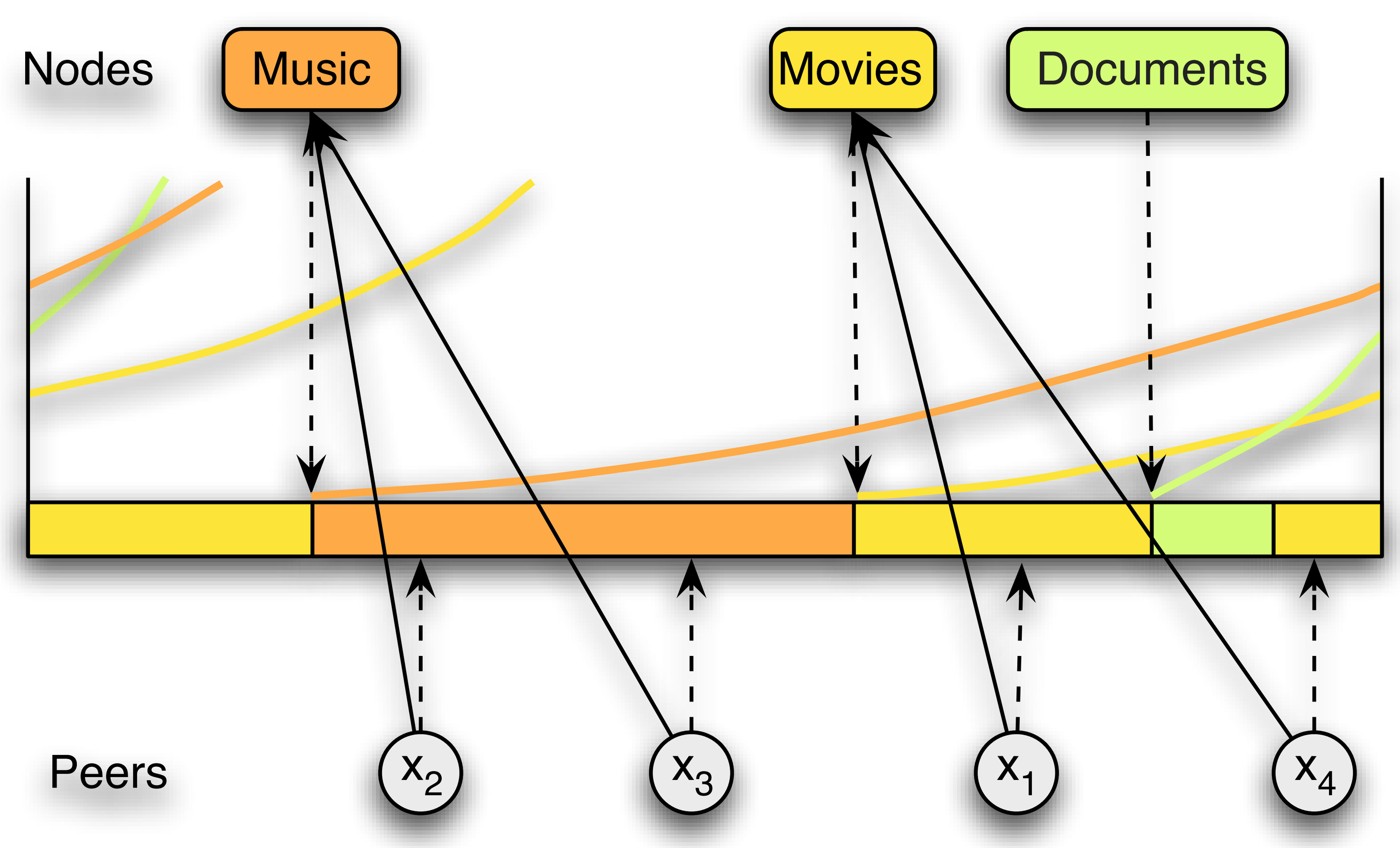
In 3nuts each peer initiates Push&Pull operations periodically, so that the network inherits the reliability of truly random graphs.



Load Balancing

For the recursive assignment of Peers to subtrees 3nuts uses **Distributed Heterogeneous Hash Tables** (DHHTs) [2]. Since 3nuts hashes peers onto subtrees and not vice versa, the number of peers assigned to a subtree is proportional to the current load of the subtree, e.g. the amount or the popularity of all data in this subtree. Using the technique introduced in [2], we achieve a provable fair load balancing.

By the combination of DHHTs and search trees each peer possesses all information to reassign itself in the case of load changes, arrival or departure of peers.



Epidemic Maintenance Algorithms

3nuts distributes maintenance information within each random network using epidemic algorithms. It is crucial for epidemic algorithms to build in termination mechanisms. 3nuts uses the median counter algorithm of [3] for randomized rumor spreading. These operations are integrated into the periodical Push&Pull operations. This way information can be reliably broadcasted within $O(\log n)$ parallel time steps and with $O(n \log \log n)$ update operations.

References

[1] Peter Mahlmann and Christian Schindelhauer, *Distributed Random DiGraph Transformations for Peer-to-Peer Networks*, under submission.

[2] Christian Schindelhauer and Gunnar Schomaker, *Weighted Distributed Hash Tables*, SPAA 2005, Las Vegas (NV), USA.

[3] Richard Karp, Scott Schenker, Christian Schindelhauer, Berthold Vöcking, *Randomized Rumor Spreading*, FOCS 2000.

Contact

Heinz Nixdorf Institute
Christian Schindelhauer
Fürstenallee 11
33102 Paderborn, Germany

Fax: +49 5251 60-6482
Phone: +49 5251 60-6692
E-Mail: schindel@uni-paderborn.de
URL: www.upb.de/cs/schindel.html

Additional Authors

Peter Mahlmann
Computer Science Department
Heinz Nixdorf Institute
University of Paderborn

Thomas Janson
Computer Science Department
Heinz Nixdorf Institute
University of Paderborn

DELIS: Integrated European Project founded by the "Complex Systems" Proactive Initiative within the Sixth Framework Programm