

LiquidXML: Adaptive XML content redistribution in DHT P2P Networks

Asterios Katsifodimos

PhD Student

INRIA Saclay (Leo team) & Université Paris-Sud 11

Joint work with:

Ioana Manolescu

Jesus Camacho-Rodriguez

Outline

- Introduction
- ~~What is a DHT~~
- ~~The ViP2P Platform~~
- →LiquidXML

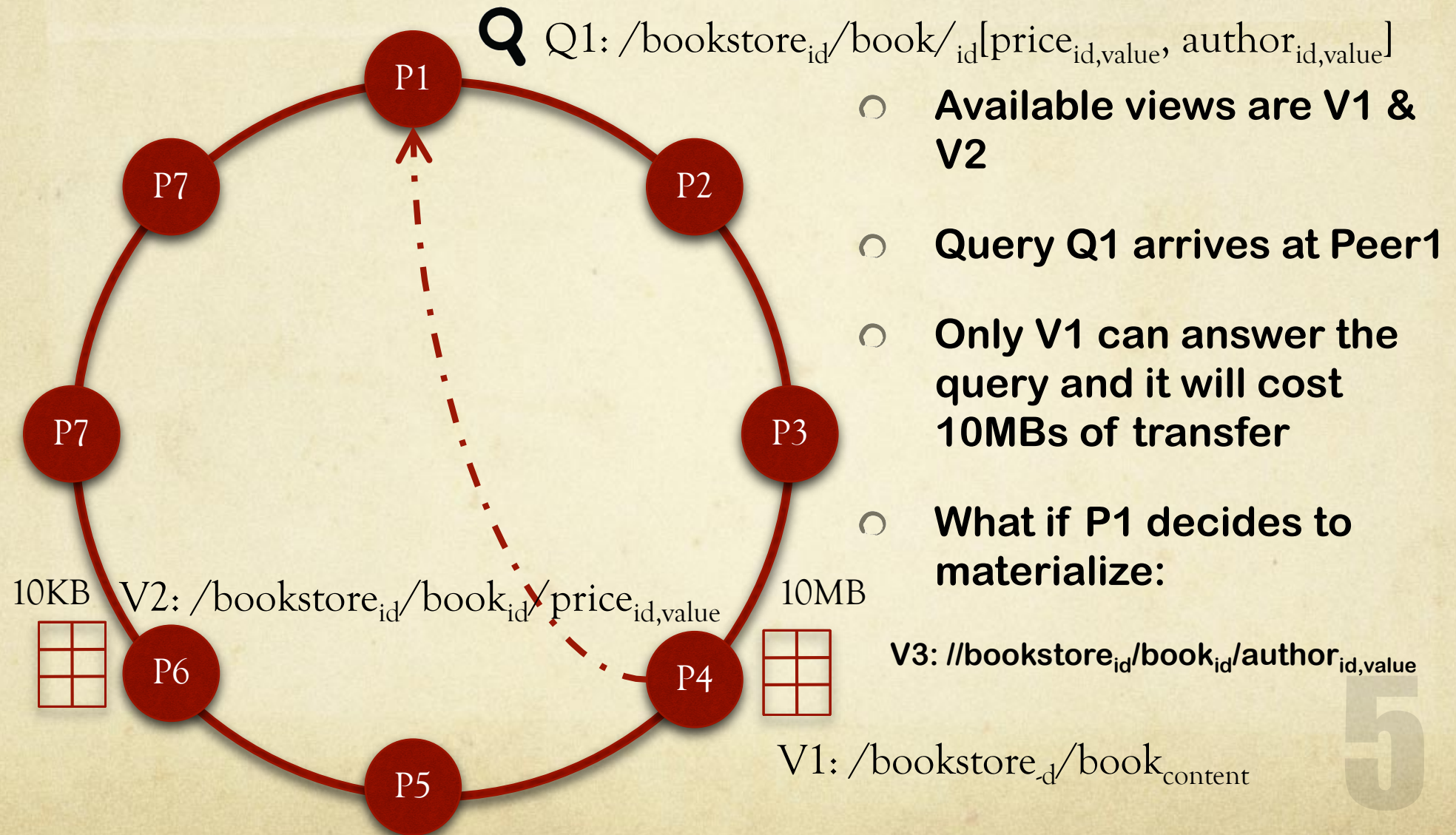
LiquidXML

Adaptive views for ViP2P

From static to liquid views

- **ViP2P drawbacks:**
 - Views do not adapt to the query & data workload
 - Users may not be willing to explicitly declare views
 - Even if they do: they cannot predict all the possible queries
 - Missing views can lead to unanswered queries
- **Our solution:**
 - Build views according to the query & data workload

From static to liquid views (2)



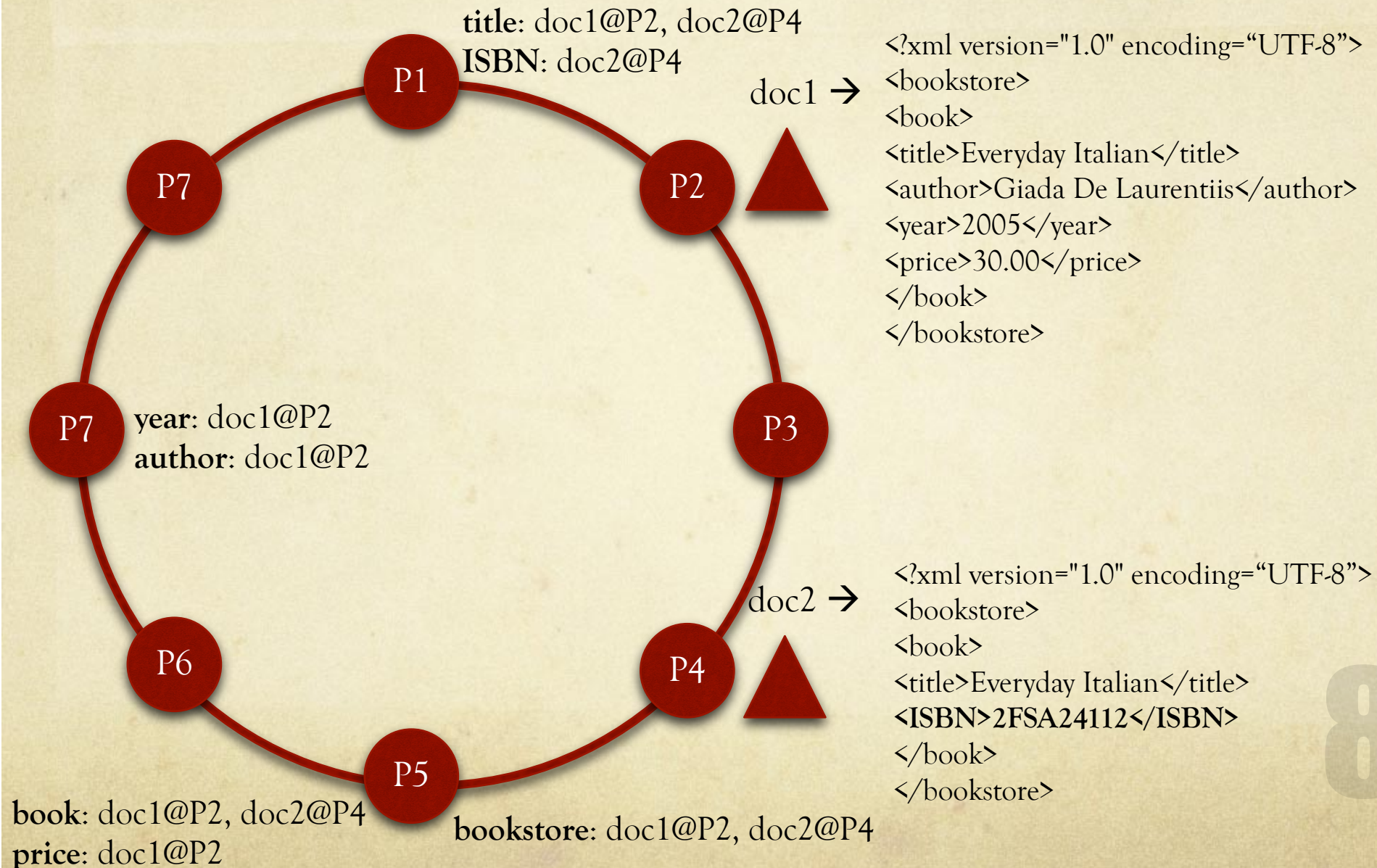
LiquidXML: ingredients

- Adaptive views
 - Peers materialize views trying to eliminate query response time
 - Views evolve with time
- Peers:
 - Have a limited space budget S for storing views (or else it would materialize all queries)
 - Query statistics: queries & frequency
 - Document synopses:
 - Index a synopsis (TreeSketch) for each document
 - Allows to make selectivity estimations

LiquidXML: baseline views

- The network starts without any views
- Baseline (document-level) views
 - For each tag t , index the documents containing it in the DHT
- All possible queries can be answered using baseline document-level views
 - By locating relevant documents and
 - Shipping the query to the respective peers
 - Merging the results received

Document level views



Adapting to the query workload : the recipe

1. Collect queries keeping statistics for an amount of time
2. Find subtrees that are “frequent*” in the workload
3. For those subtrees (candidate views) calculate:
their **cost** (in terms of space) & their **benefit** in
case they are materialized
(according to the query workload)
4. Materialize the set of candidate views that:
 1. fill the available space budget and
 2. achieve the maximum benefit

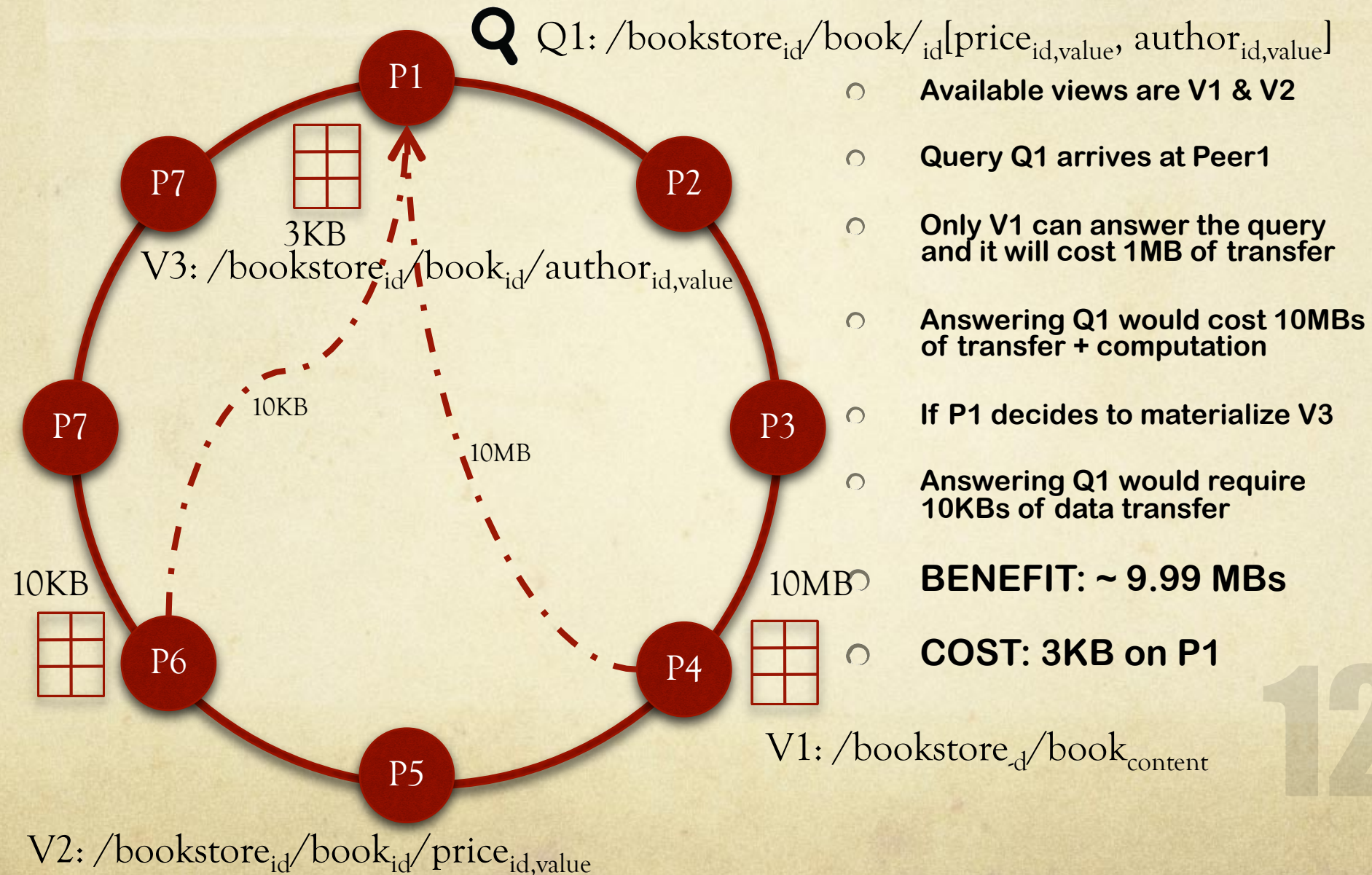
View cost: size estimation

- Synopses of all published documents are indexed in the DHT
- For estimating the size of a view V :
 - Locate the documents that contribute to it
 - Retrieve their synopses
 - Using the synopses, calculate each document's (d) contribution to the view v ($|v(d)|$)
 - The size of the view is the sum of contributions of individual documents

View benefit estimation

- If a candidate view is materialized it changes the rewriting of some queries
- According to the query workload we can calculate the benefit of materializing a view
- Example follows...

Cost & Benefit of materializing a view



Questions