HomeViews

Geambasu *et al.*, 2007

Presented by Jeremy Handcock

Motivation

- We have a lot of personal digital data
- Existing solutions for organizing and sharing aren’t ideal

Typical Scenario

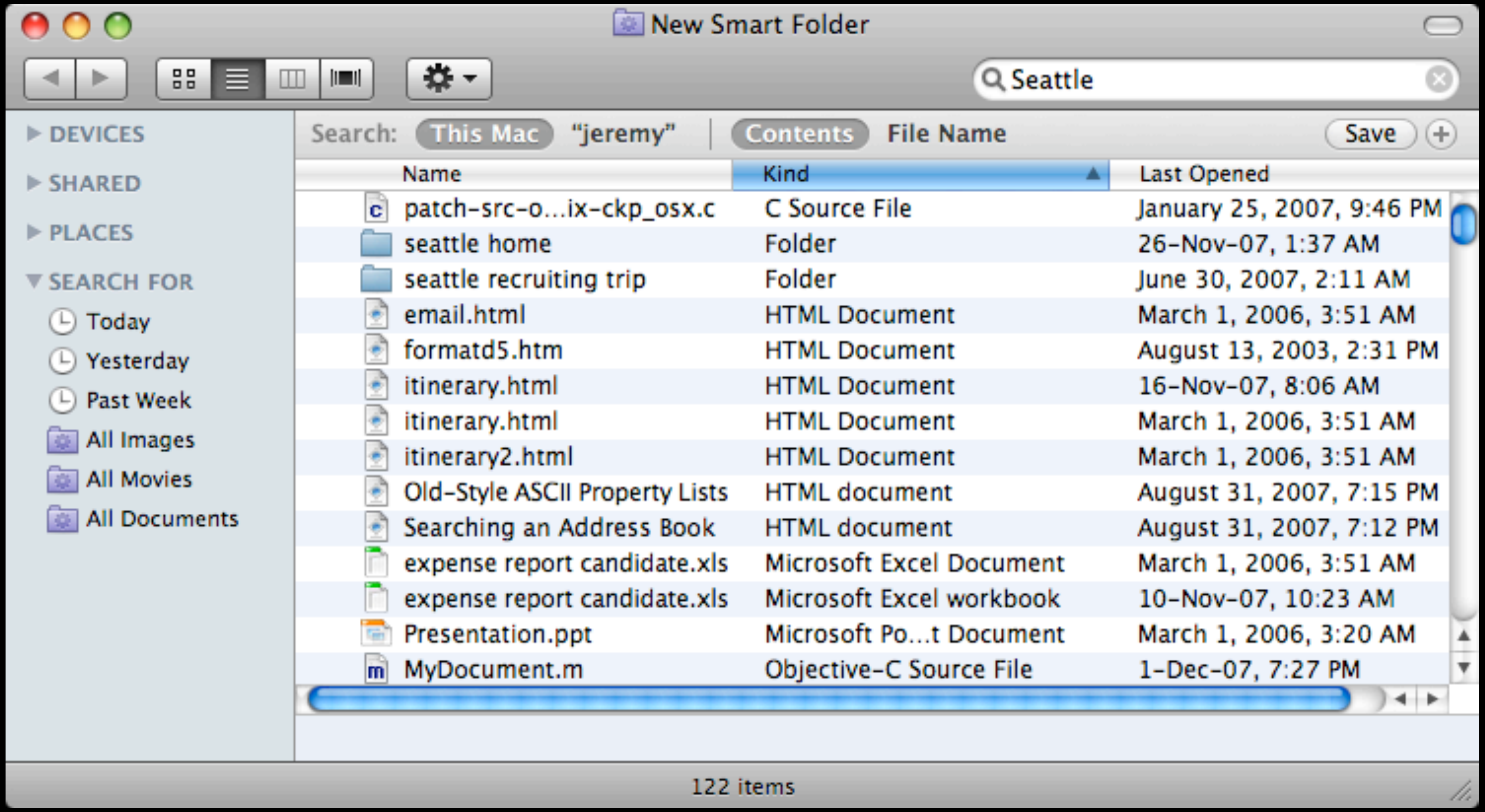
Existing Solutions: Sharing

- One must manually upload content to centralized repository
- Centralized repositories typically require registration for new users
- Inevitably you end up with multiple copies
- Dynamic data isn’t supported

Existing Solutions: Organizing

- Desktop search tools
- Create abstract data views or smart folders
- Requires metadata for non-textual objects

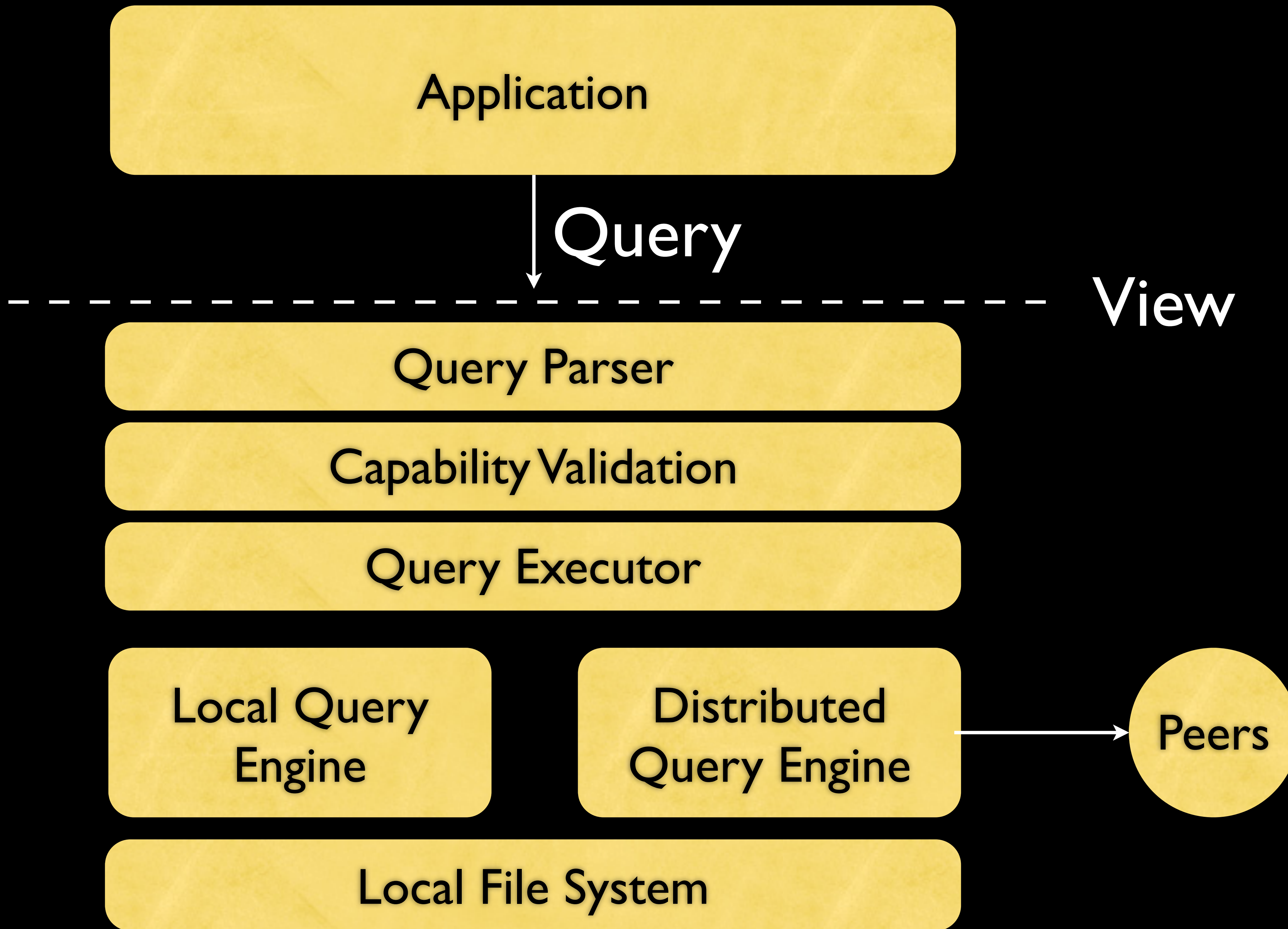
Existing Solutions: Organizing



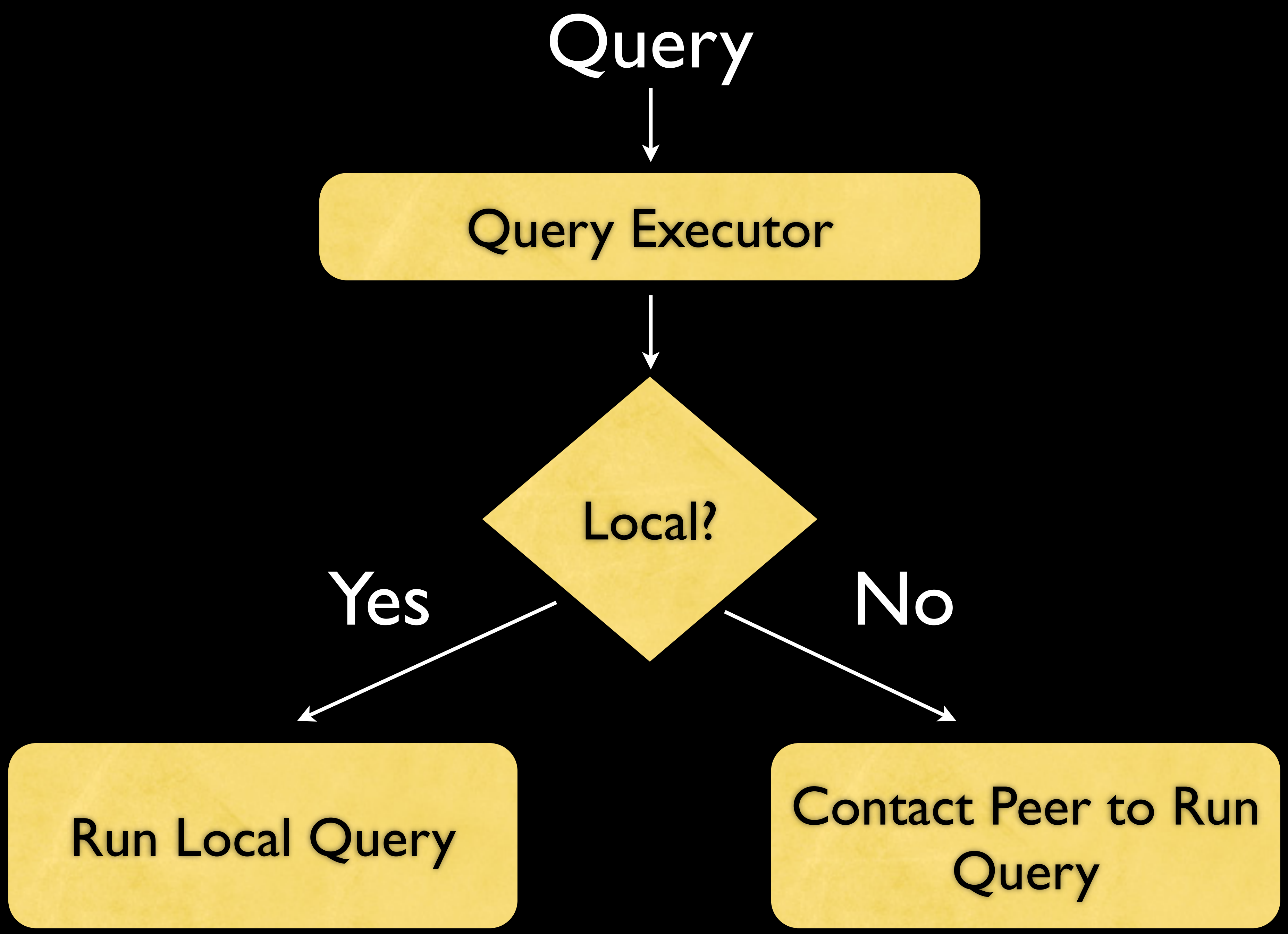
Idea!

- Decentralize data using P2P
- Share data using location-independent abstract views
- Provide lightweight access control
- Put it all in a middleware framework

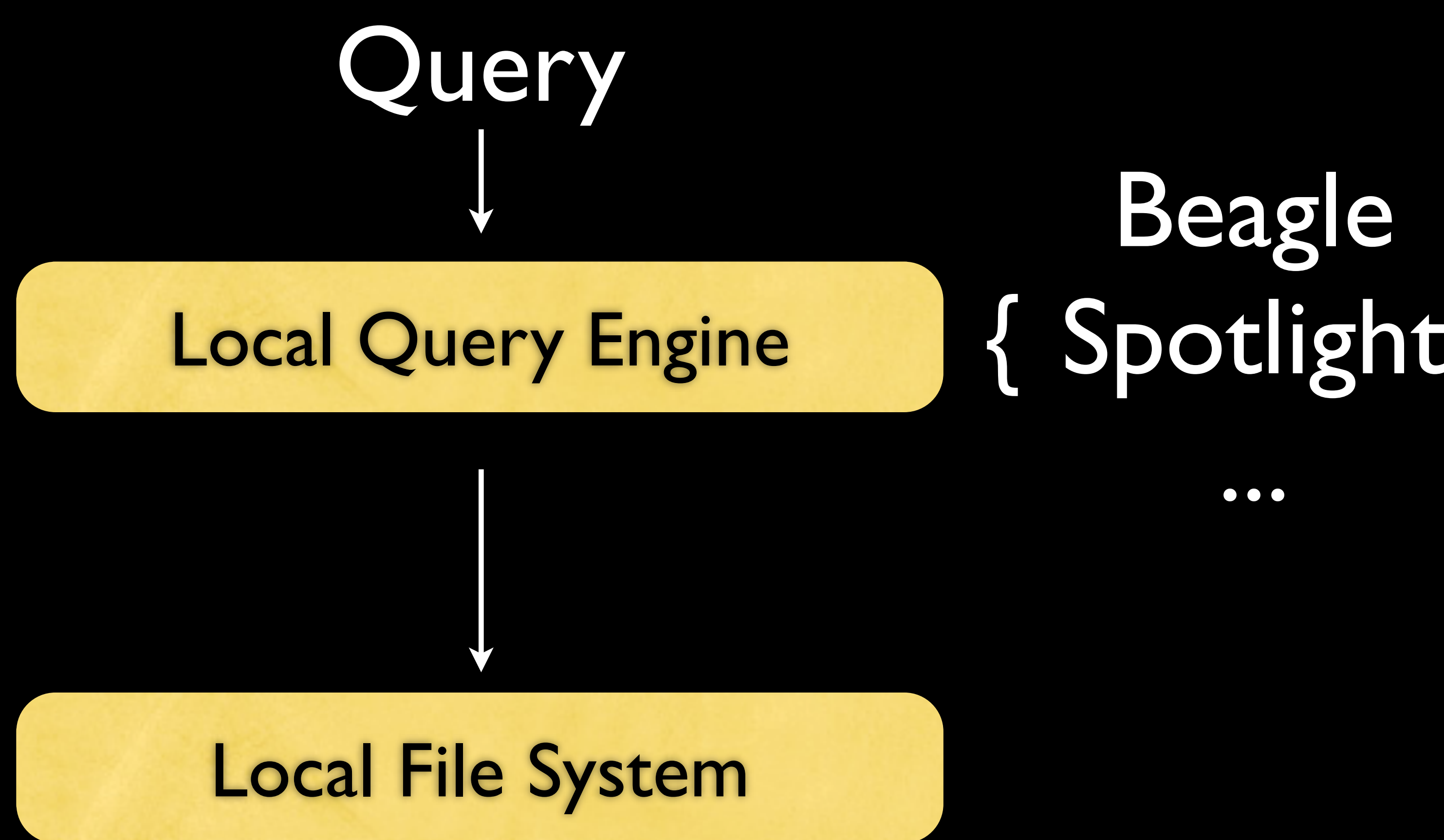
HomeViews


Application
Query
View
Query Parser
Capability Validation
Query Executor
Local Query Engine
Distributed Query Engine
Peers
Local File System

Query Processing



Query Processing



Capabilities

- Associated with a view
- Self-authenticating unit of permission
- Capabilities unforgeable with high probability
- Used in global naming and routing

Capabilities

global view ID	password	IP hint
...	...	...

Hash of node's MAC + Locally-unique view ID

128-bit random password

Capabilities

- Capability provides location information
- IP hint hopefully points to node where capability is defined
- When IP Hint fails, use a DHT (Chord) to locate the content

Capabilities

- Capability could be represented as a simple "token"
- Distributing capabilities is annoying, but relatively easy
- Revocation is easy

Capabilities

- Is this an appropriate model for sharing personal digital data?
- Do capabilities provide sufficient security?

Queries

- Extension of SQL
- Fundamental difference: relations replaced with capabilities
- Define a capability by creating a view
- Base capability provides access to all files

Queries: Views

```
CREATE VIEW MyFiles2007 AS
SELECT * FROM BaseCapability
WHERE createDate > '2007-01-01'
AND createDate < '2008-01-01'
```

⟶ New capability: `Jeremy2007`

Queries: Restrictions

```
RESTRICT Jeremy2007
RIGHTS SELECT
```

→ New capability: `Jeremy2007_RO`

Queries: Selections

```
SELECT * FROM Jeremy2007_RO
WHERE fileType = ‘image’
```

Queries: Complex Views

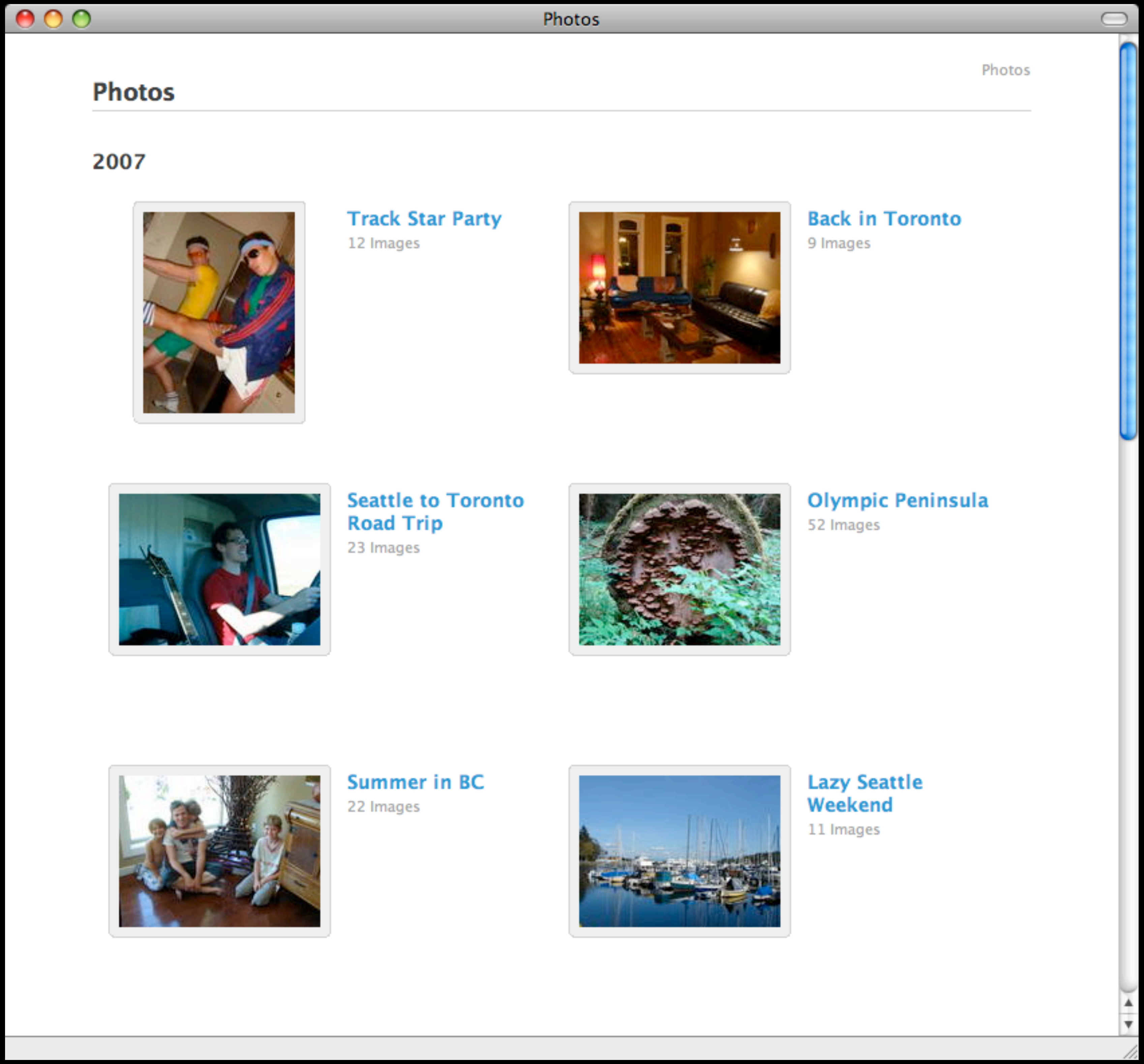
```
CREATE VIEW MyFriendMovies2007 AS

SELECT * FROM Jeremy2007_RO
WHERE fileType = ‘movie’
UNION
SELECT * FROM Rob2007_RO
WHERE fileType = ‘movie’
```

⟶ New capability: `FriendMovies2007`

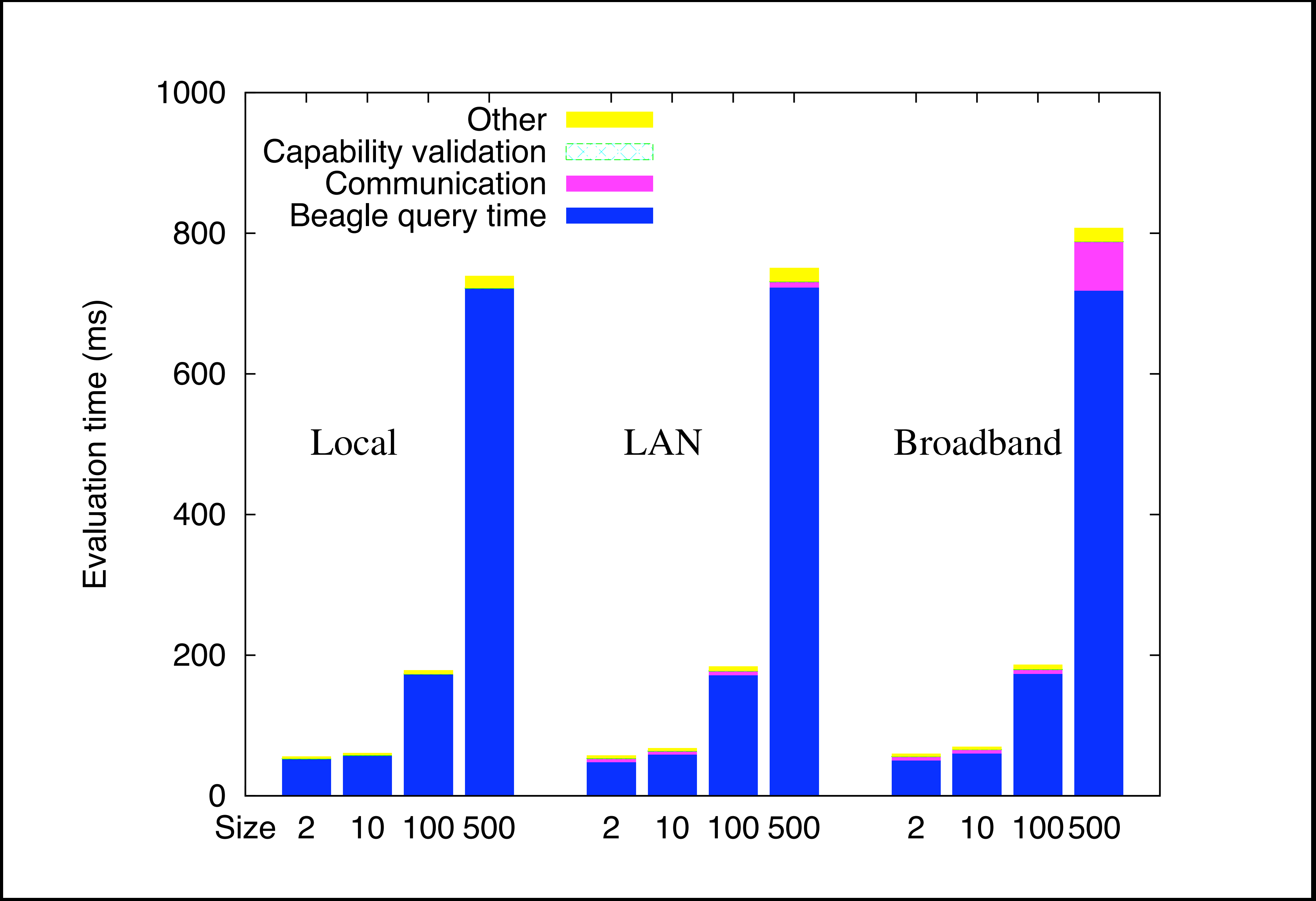
Evaluation: Applications

- Simple read-only filesystem
- Photo gallery

Evaluation: Applications

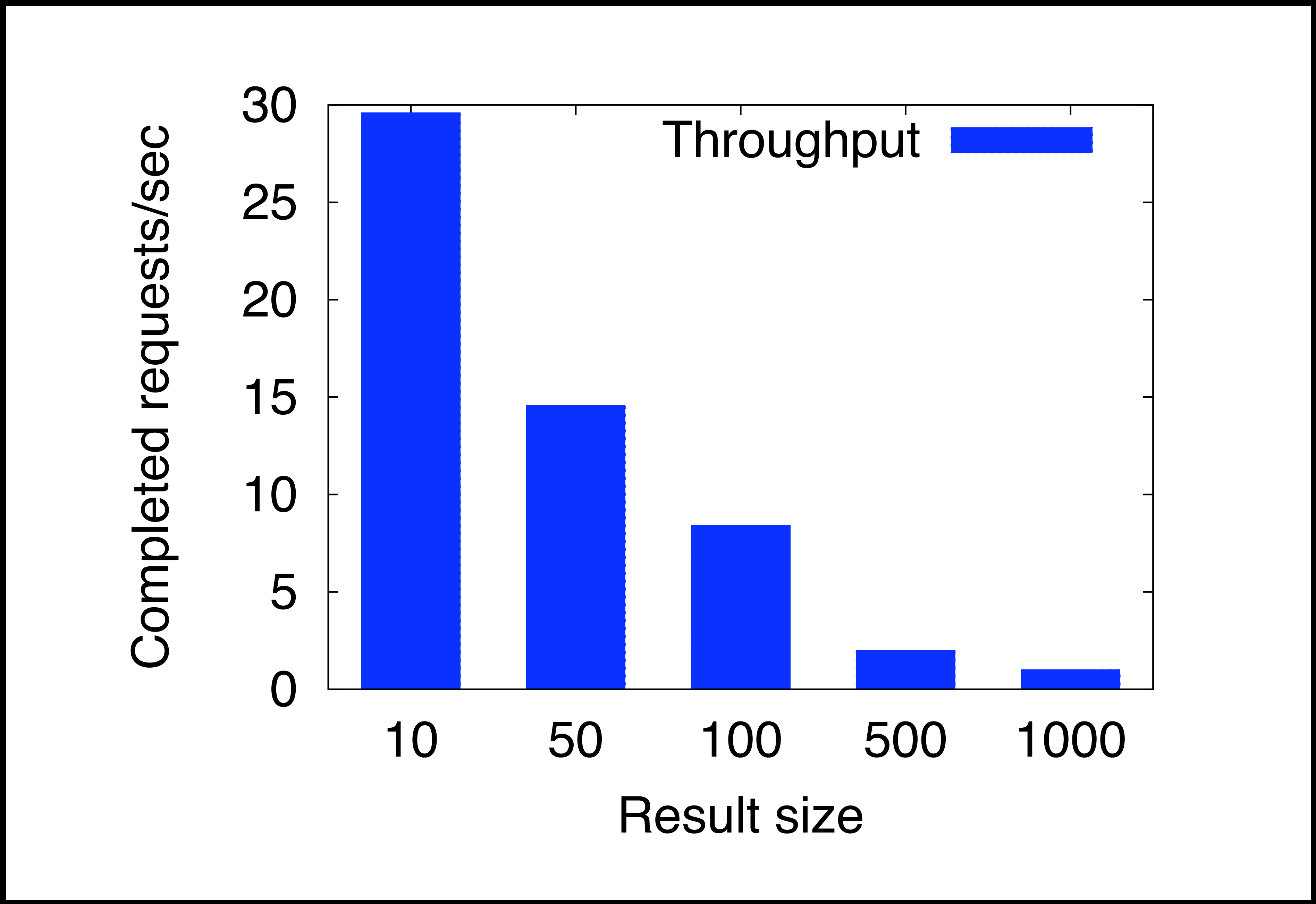


Evaluation: Numbers



Result Size

Evaluation: Numbers



Applicability to OSNs

- Middleware platform for OSN applications to share data between users: ie. PeerSpective, Facebook
- Capability management could be hidden

Applicability to OSNs

- Danger: lack of explicit sharing action and dynamic data

Conclusions

- P2P not always appropriate model for sharing personal digital data
- Complex views offer opportunities for poor performance
- We still don’t have an ideal solution