Graph Based File Prediction

Relation graph of file access

The relation graph is aiming to represent the relations of file access in a certain time period, we assume such information can help us improve the prediction of future file access.
Description of the graph:

1. Nodes:
   Each file is represented as a node in the relation graph. Nodes have a property called placement, which could either be SSD or HDD.
2. Edges:
   When two files have been accessed together, then there will be an edge between two nodes. By together, we mean in a short time period, which we call edge adding window.
   Edges will expire after a certain time period, which we call edge removal window.
   Edges have weights, that is, two nodes can be accessed together more than once, the weight of an edge represent how many time two nodes have been accessed together.
3. Connections:
   The sum of weights of edges with all neighbors of a node.
   The more connections a node has, the file are related to more other files.
4. Ratio of connections:
   The sum of weights of edges with neighbors in SSD / The sum of weights of edges with all neighbors (connections of a node).
   This ratio reflect the degree of a file's relation with files in SSD.

The prediction experiments settings

We are using

1. ratio of connections
2. connections
3. accumulated access frequencies

to do the prediction of three kinds of traces:

1. SEER Traces:
   I/O traces taken at the system-call level (excluding reads and writes but including opens, closes) of computers used for software development by CS researchers, some with very heavy activity.
2. LASR Traces:
   I/O traces taken at the system-call level of computers used for software development by CS researchers.
3. ALCF Traces:
   Application characterization data collected using the Darshan characterization tool

The benchmarks we used for the experiments:

1. Performance (number of accesses from SSD):
   Based on the prediction results, we recorded the number of accesses from SSD as the main performance benchmark for each experiment.

2. Overhead (number of file movements):
   Based on the prediction results, we recorded the the number of files being moved as the main overhead benchmark for each experiment.

Brief results

1. SEER Traces:
   Performances of using all 1, 2 and 3 are good.
   Overhead of 1 and 3 are acceptable. Overhead of using 2 is too much.
2. LASR Traces:
   Performances of all three are not very good, but 2 and 3 are better than 1.
   Overhead of 1 and 3 are acceptable. Overhead of using 2 is too much.
3. ALCF Traces:
   Performances of all three are not very good, but 2 and 3 are better than 1.
   Overhead of 1 and 3 are acceptable. Overhead of using 2 is too much.