number of references) follows a geometric distribution with parameter p_i.

If accesses to an object are in fact correlated, then the object's IAT distribution will deviate from the geometric; so to measure correlation we introduce a metric that is very sensitive to this deviation. The *coefficient of variation* (CV) of a distribution is its standard deviation divided by its mean. To form a metric for a given trace, we take an average of IAT-CV over all references in the trace, as described in [4].

For the geometric distribution with parameter p_i, the CV is

$$\sqrt{1 - p_i}\,.$$

In Web reference streams, in which even the most heavily accessed objects have a very low probability of reference (generally much less than 1%), the expected IAT-CV in the case of no temporal correlation is very close to one. Therefore, in Web reference streams, CV values close to unity suggest that reference patterns are close to the IRM (contain little temporal correlation); while values larger than one represent a distribution with large relative variance, and so suggest the presence of strong inter-reference correlations.

	Filtering	Aggregation	Disaggregation
Popularity	Increases entropy	Decreases entropy, especially near servers	Little effect
Correlation	Decreases IAT-CV	Little effect	Decreases IAT-CV especially near clients

Summary of the effects of the different transformations on the two components of temporal locality.

Thus, IAT-CV is an effective metric for capturing temporal correlation. In a manner similar to normalized entropy, the importance of correlation in a reference stream can be measured by its effect on hit ratio in a LRU cache. When a trace is scrambled (removing correlations) the resulting hit ratio tends to decrease; in [4] we find this decrease is strongly correlated with IAT-CV.

Exploring Stream Locality

So far, we have been devoted to decomposing the complexity of Web systems along two axes: the transformations operating on Web streams have been decomposed into aggregation, disaggregation, and filtering; and the locality property of request streams has been decomposed into popularity and correlation. Using this decomposition, we can begin to attack the underlying problem, namely the systemwide engineering of the Web. Organizing the various processes of the system along these two axes yields a set of tractable analyses that can ultimately be recombined into a systemwide view. The idea is to consider how each transformation operates on each source of locality; we organize our findings in the table here. The results in this table are derived from measurements of Web traces, experimental evaluation of transformations of Web traces, and analytic considerations, as described in [4].

Filtering. Filtering by a cache, under commonly used cache replacement policies like LRU and least frequently used (LFU), tends to remove both components of locality. It removes popularity (increases entropy) because highly popular objects are more likely to be found in the cache; and it tends to remove correlation (decreases IAT-CV) because recently referenced objects are more likely to be found in the cache.

Aggregation. Aggregation of streams coming from distinct upstream sources tends to increase locality, especially near servers. This occurs because entropy decreases, while IAT-CV changes very little. Entropy decreases because the popularity of documents in the merged stream is generally more skewed than in the component streams being merged. This surprising effect, observed in empirical traces, occurs because globally popular documents tend to occur in each component stream, while unpopular documents do not. This effect is most pronounced near servers, where the largest number of independent streams are being merged.

Disaggregation. Disaggregation into separate streams tends to reduce locality, but its effect is only pronounced near clients. Disaggregation has little effect on entropy; this seems to be because resulting downstream components tend to maintain skewed object popularity approximately like the original stream. This is consistent with the many studies that have shown that even among objects on a single Web server, Zipf's Law is quite pronounced. Disaggregation does tend to reduce IAT-CV, but this effect is only pronounced when disaggregation occurs close to the client. At most other places in the Web system there is little correlation in either incoming or outgoing streams.

Location in the Web of Streams. Applying these insights to the Web system, we can begin to understand how locality changes as a function of location in the system. Referring back to Figure 1(b), we say that a component is near clients if it receives requests from relatively few clients, and sends requests to relatively many servers; and if the situation is reversed, the component is near servers. We can then ask how