Report on topic:

"Resuming failed file transfer between squid proxy server and client machine"

According to the documentation, A *range request* comes from a client that wants only some subset of an HTTP response, that could be handy in our case when we want to achieve auto resume of download process on proxy server, which request was initiated by client. They are sometimes used to resume a failed transfer of a large file. Squid isn't yet able to cache partial responses and thus must make a decision when forwarding a range request: either remove the `Range` header or leave it in.

If Squid leaves the `Range` header in, the origin server sends only the subset that the client wants, and the client receives the response immediately. However, this partial response isn't cached.

On the other hand, if Squid removes the header before forwarding, it receives the entire response, which may be cached. Squid is then responsible for ensuring that the client receives only the subset it needs. The origin server may send a lot of data the client doesn't want. Depending on the speed of your connection, the client may be forced to wait a long time until its range is available.

If the beginning of the requested range is larger than the *range_offset_limit* value, Squid forwards the `Range` header and doesn't cache the response. Setting *range_offset_limit* to `0` causes Squid to always forward the `Range` header (the default). Setting it to `-1` causes Squid to never forward the header.

Squid proxy server can help resume downloading data after a client's connection is interrupted while connected through it. This process is based on caching functionality and handling partial content requests. When a client downloads a file through Squid and suddenly loses the connection, Squid caches the downloaded data.

Upon the client's next connection, Squid can provide a portion of the cached data, allowing the client to continue downloading from where it was interrupted.

Squid Configuration: Make sure your Squid is configured to handle partial content requests and has a sufficiently large range_offset_limit setting to cache the necessary amount of data for resuming the download.

Handling Ranges: Squid automatically processes ranges of data requested by clients. When a client sends a request with a Range header, Squid responds with a portion of cached data corresponding to that range.

Client Support: The client downloading the data also needs to support resuming downloads after interruptions. For example, when using the curl command, the client must correctly handle range requests and resume the download.

Cache Expiry: The Squid cache can store data for a limited time. If the client doesn't reconnect and resume downloading for an extended period, the cached data might be removed based on cache lifespan rules. Remember that Squid generally handles resuming downloads after connection interruptions well. However, the success of this process also depends on caching settings, correct client configuration, and proper handling of range requests on the server-side.