

# BF-based chunk availability compression for PPSP

Lingli Deng: [denglingli@chinamobile.com](mailto:denglingli@chinamobile.com)

Jin Peng: [pengjin@chinamobile.com](mailto:pengjin@chinamobile.com)

Yunfei Zhang: [hishigh@gmail.com](mailto:hishigh@gmail.com)

Rachel Huang: [rachel.huang@huawei.com](mailto:rachel.huang@huawei.com)

IETF 88@Vancouver

# Outline

- Recap: BF scheme for PPSP
- Changes since -02
- Open issues
- Next steps

# Recap: Motivation

- There are frequent bitmap exchanges in PPSP.
  - Uncompressed bitmap is relatively big (of several KBs).
  - They are exchanged frequently (less than several seconds).
  - It sets a limit to the system's efficiency and scalability
- There are efficiency requirements in PPSP PS
  - PPSP.TP.REQ-3: The tracker protocol MUST take the frequency of messages and efficient use of bandwidth into consideration, when communicating chunk availability information.
  - PPSP.PP.REQ-7: The peer protocol MUST take the frequency of messages and efficient use of bandwidth into consideration, when communicating chunk information.
- Existing schemes proposed involves uncontrollable worst case performance.

# Recap: BF compression Scheme

```
-----  
BF(set S, integer m, hash set H)  
1 filter=allocate m bits initialized to 0;  
2 for each element xi in S do  
3   for each hash functions hi in H do  
4     filter[hi(xi)]=1;  
5 return filter;  
-----  
MT(element elm, BF filter, integer m, hash set H)  
1 for each hash functions hi in H do  
2   if (filter[hi(elm)]!=1)  
3     return false;  
4 return true;  
-----  
ST(BF query, BF filter)  
1 temp=query OR filter;  
2 if (temp!=filter)  
3   return false;  
4 return true;  
-----
```

Figure 1 Basic algorithms for BF-bitmaps.

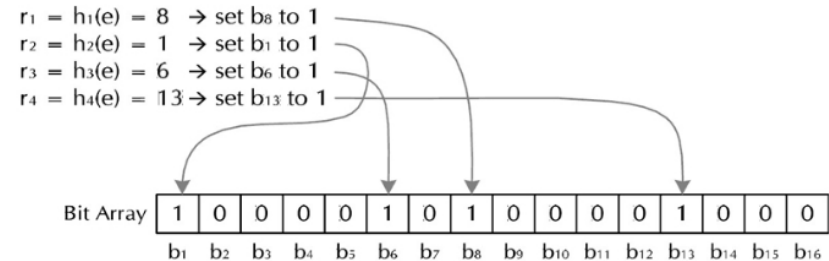


Figure 2 Bloom Filters: an example.

## High Efficiency

Storage/transmission: Bit length: constant.

Processing: Formation/Inquiry/Update: constant.

## Endurable Lose of accuracy

Be controlled by the system configuration of the bit array's length, choice and # of hash functions.

**Example:** a 2GB movie file, divided into 2MB chunks, whose a 1024-bit original chunk bitmap, can be represented by a 128-bit BF-bitmap (using 4 hashes), with only 3% mis-hits rate.

# Recap: Proposals for integration with PPSP family

- RECOMMENDED for PPSP-TP-base/extended
  - Strictly controllable cost for a central entity
    - constant bit-length irrelevant of the chunk-set
    - only replacement or simple bitwise operations needed
  - Certain mis-hits rate COULD be tolerable
    - Tracker serves as an initial broker for neighboring peers
- OPTIONAL for PPSP
  - Peers willing to trade accuracy with cost-efficiency
    - Peers with limited computation/memory resources
    - Peers with huge number of concurrent links, e.g. SNs
  - Certain mis-hits rate MAY be tolerable
    - REQUEST and DATA SHOULD use the original chunk id.

# Changes since -02

- Add Rachel Huang (from Huawei) as co-author
- Changes to BF-based Chunk Availability Exchange
  - Use offline per swarm configuration instead of online notification via tracker protocol for BF parameters
  - Integrate with extended tracker protocol via “StatisticsGroup” element for STAT\_REPORT messages
  - Integrate with peer protocol via a new chunk addressing method

# BF Algorithm Configuration

- Suggestion: per swarm offline configuration
  - BF configuration be stored at the web portal and published to a requesting peer through the web page or MPD file transaction.
- Proposal: add description to the "Installation and Initial Setup" in base TP
  - "In case of a peer or the tracker wishes to exchange further information about the available peers in a flexible way, e.g. the chunk availability information of a specific peer in the same swarm could be represented in a various ways, there should be a way of indication about the specific method/parameters in use, e.g. in the MPD file downloaded by the requesting peer from the web portal."

# Integration with Extended TP

- StatisticsGroup in EXT TP
  - "StatisticsGroup" element in STAT\_REPORT message extended to contain contentmap info
- Proposals to incorporate BF
  - "SegmentInfo" used for the BF-formatted bitmap;
  - "startIndex", "endIndex" and "chunkmapSize" left blank.

Element Name or Attribute Name	Use	Description
StatisticsGroup	0...1	Provides statistic data on peer and content.
Stat	1...N	Groups statistics property data.
@property	M	The property to be reported property values and elements in Table 5 of [I-D.ietf-ppsp-base-tracker-protocol]
Representation	0...N	Describes a component of content.
@id	CM	Unique identifier for this Representation.
SegmentInfo	1...N	Provides segment information by segment range. The chunkmap can be encoded in Base64 [RFC4648].
@startIndex	CM	The index of the first media segment in the chunkmap report for this Representation.
@endIndex	CM	The index of the last media segment in the chunkmap report for this Representation.
@chunkmapSize	CM	Size of chunkmap reported.

Table 3: Semantics of StatisticsGroup.



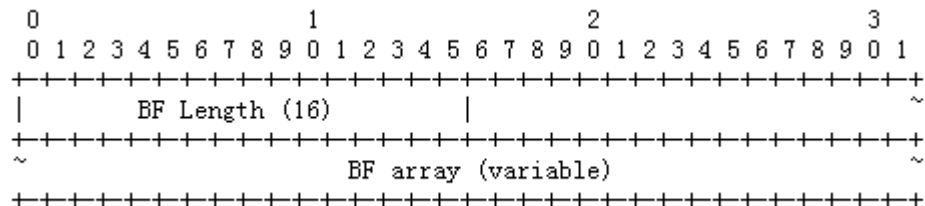
# Integration with PPSP

- Chunk addressing methods
  - Indicated via HANDSHAKE
  - Used in HAVE/REQUEST/ACK
- Proposals for BF scheme

Method	Description
0	32-bit bins
1	64-bit byte ranges
2	32-bit chunk ranges
3	64-bit bins
4	64-bit chunk ranges
5-255	Unassigned

- add a defined value (e.g. 5) from "unassigned" value range for the BF-formatted bitmap;
- use HAVE/REQUEST/ACK message to convey the BF-format array for the overall local chunk bitmap.

- 16-bit length field
- BF array



# Open issue 1: Configuration

- How do we choose which bitmap scheme to use?
- Option1: system wide configuration
  - Adopted by the current **extended tracker** protocol
  - Requires one mandatory choice or configuration indication
- Option2: per-swarm configuration
  - Suggested by the **new BF draft**
  - Requires per-swarm configuration indication scheme to tracker/peer
- Option3: per-peer configuration
  - Adopted by the current **peer protocol**
  - Requires indication/negotiation via HANDSHAKE
  - Requires translation if different schemes are used by communicating peers

# Open issue 1: Configuration(cont')

- Questions to be settled before decision
  - Do we allow different schemes for Tracker/Peer protocol?
    - Yes? Then we need extensions for configuration indication.
  - Do we allow different schemes for peers in one swarm?
    - Yes? Then we need online translation for peer-peer exchange.
  - Do we allow different schemes for tracker and peer protocol transactions in one swarm, respectively?
    - Yes? Then we need online translation for tracker-peer exchange.
- How do we choose which bitmap scheme to use?
  - Per-swarm configuration for both tracker and peer protocols?

# Open issue 2: Peerlist Enhancement

- Shall we extend peerlist to contain bitmaps?
  - Use ContentGroup element in ext CONNECT/FIND request
  - Extend PeerGroup element in ext CONNECT/FIND response
- Pros:
  - Allows for a more precise description of requested chunks
  - Allows for a more efficient neighborhood establishment
- Cons:
  - Seems a bit redundant and Not as accurate as HANDSHAK/HAVE exchange in PPSPP

# Next Steps

- Hope to settle existing issues by this week.
- Any further issues or suggestions?
- Ready for WG adoption?

THANK YOU!