A Deterministic Algorithm for Summarizing Asynchronous Streams over a Sliding Window

Costas Busch
Rensselaer Polytechnic Institute

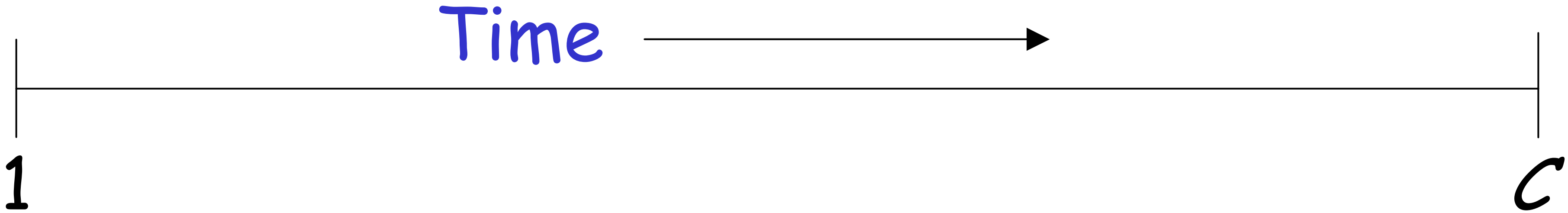
Srikanta Tirthapura
Iowa State University

Outline of Talk

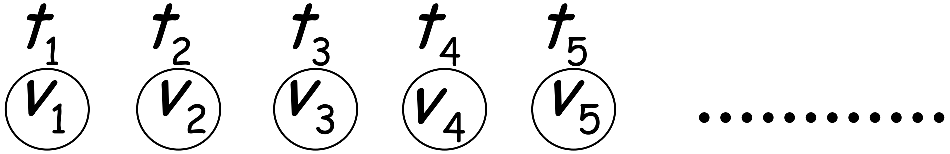
Introduction

Algorithm

Analysis

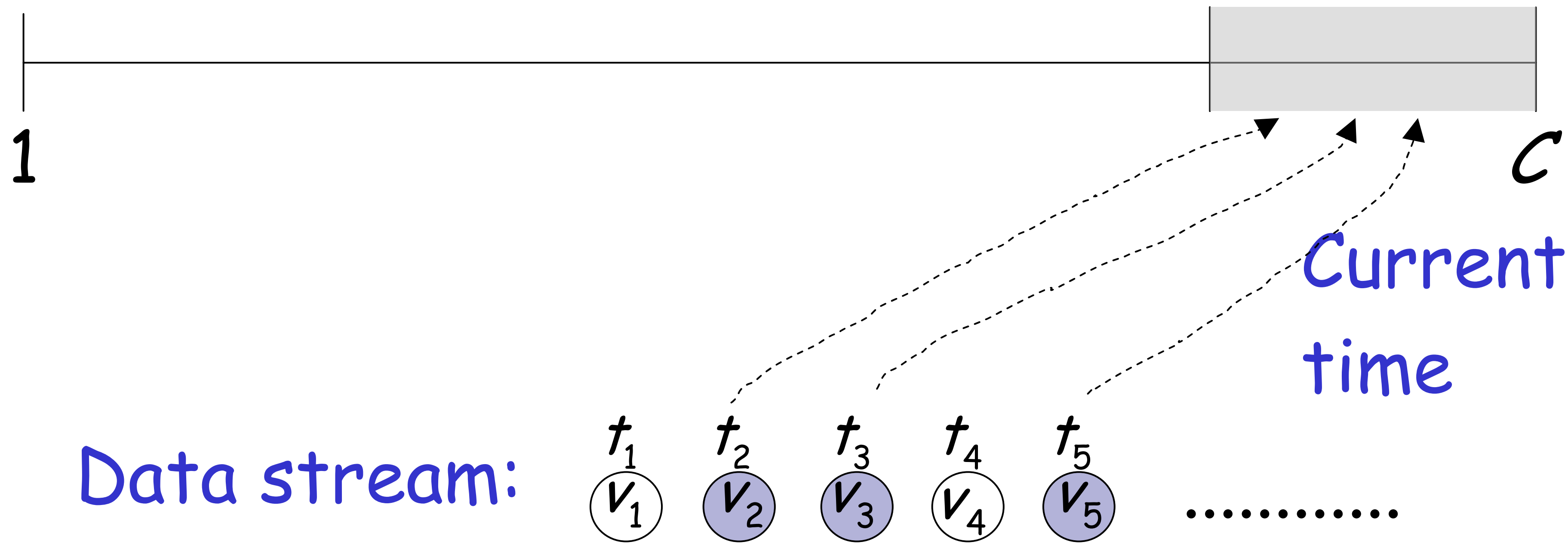


Data stream:



For simplicity assume unit valued elements

Most recent time window of duration W



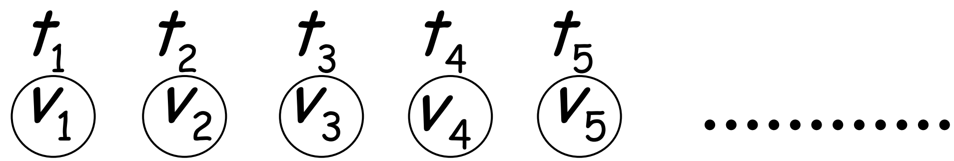
Goal: Compute the sum of elements with time stamps in time window $[C-W, C]$

$$\sum_{C-W \le t_i \le C} v_i$$

Example I: All packets on a network link, maintain the number of different ip sources in the last one hour

Example II: Large database, continuously maintain averages and frequency moments

Data stream:



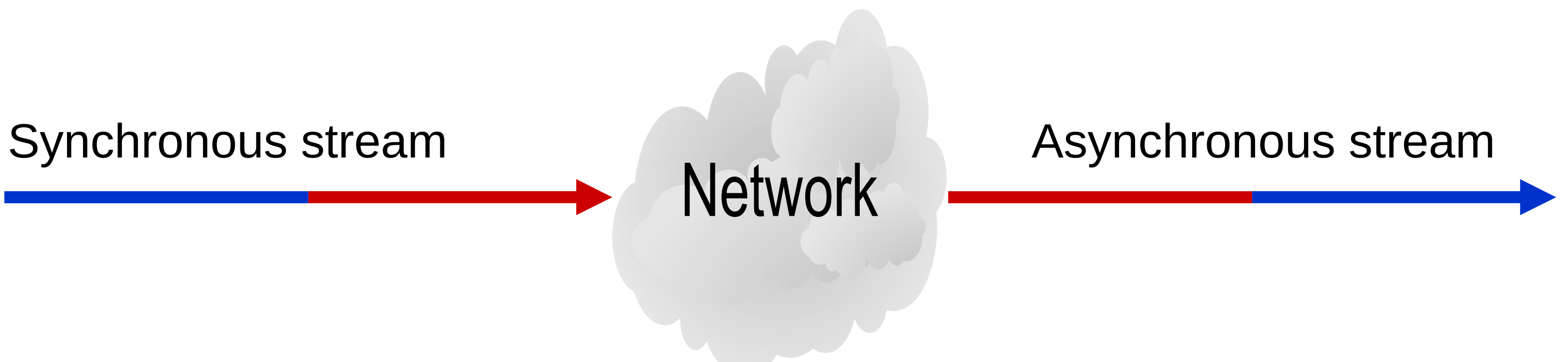
Synchronous stream

t_i: In ascending order

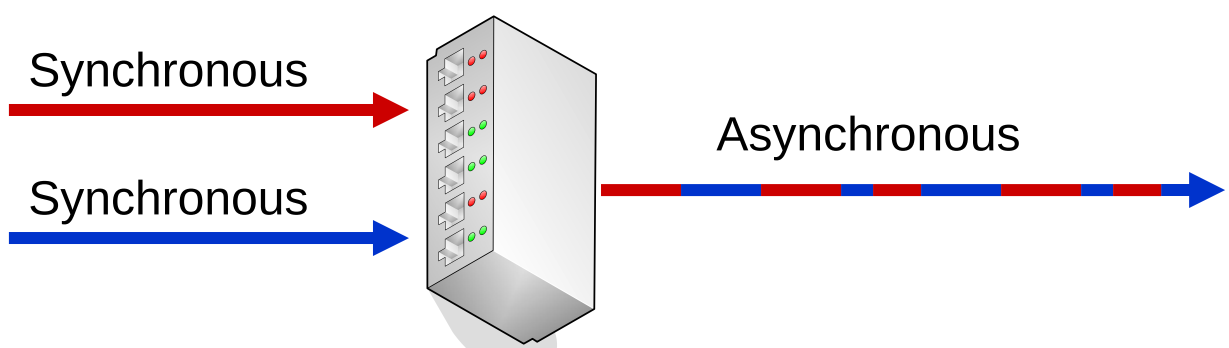
Asynchronous stream

t_i: No order guaranteed

Why Asynchronous Data Streams?



Network delay & multi-path routing



Merge w/o control

Processing Requirements:

- One pass processing
- Small workspace: poly-logarithmic in the size of data
- Fast processing time per element
- Approximate answers are ok

Our results:

A deterministic data aggregation algorithm

Time: $$O\left(\log B + \frac{\log W}{\varepsilon}\right)$$

Space: $$O\left(\log B \log W \frac{\log W + \log B}{\varepsilon}\right)$$

Relative Error: $$\varepsilon = \frac{|X - S|}{S}$$

Previous Work:

[Datar, Gionis, Indyk, Motwani.
SIAM *Journal on Computing*, 2002]

Deterministic, Synchronous

Merging buckets

[Tirthapura, Xu, Busch, PODC, 2006]

Randomized, Asynchronous

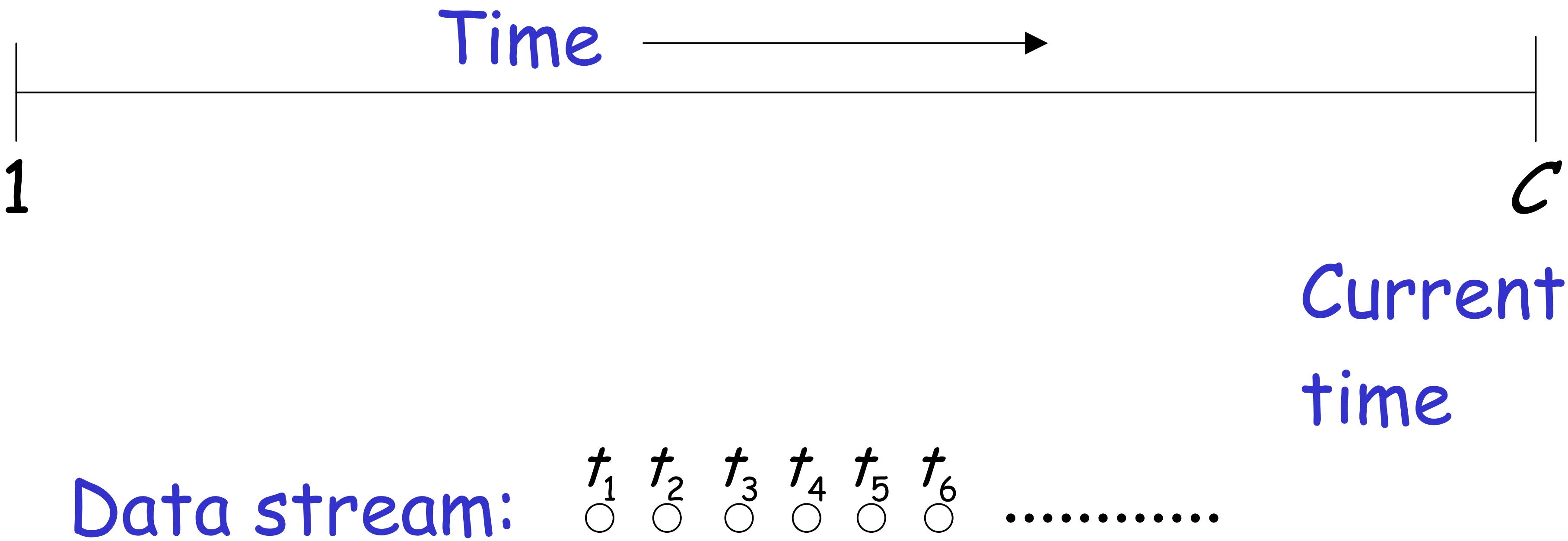
Random sampling

Outline of Talk

Introduction

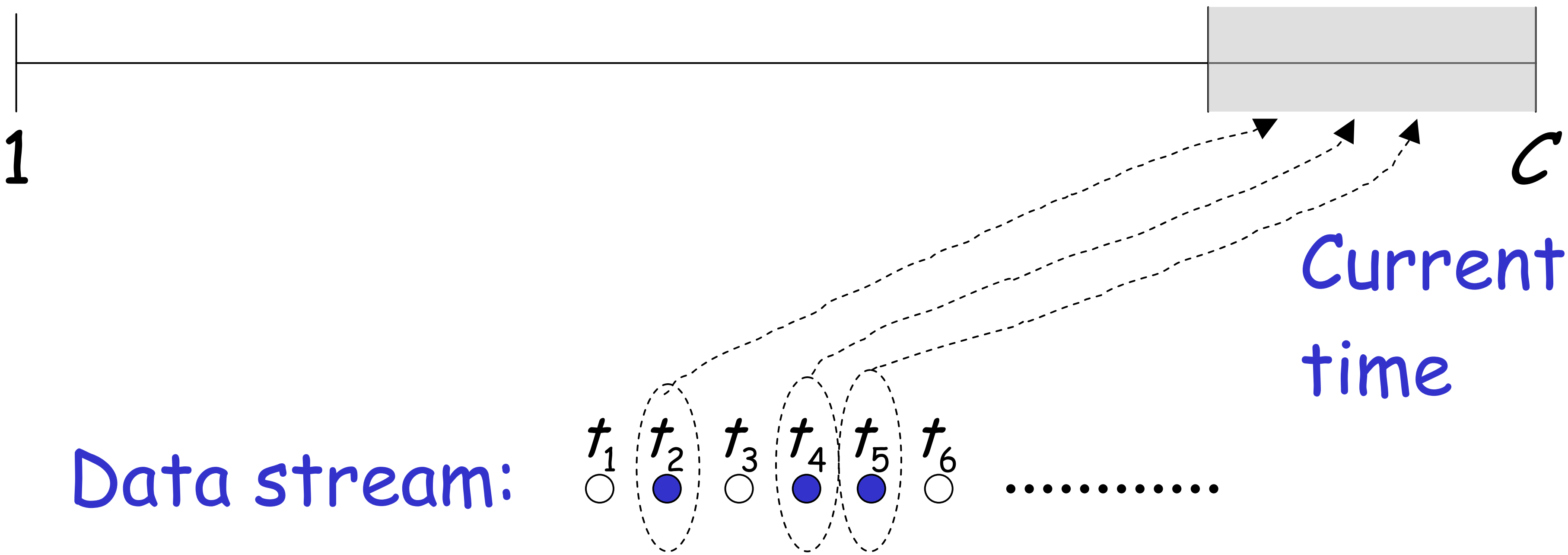
Algorithm

Analysis

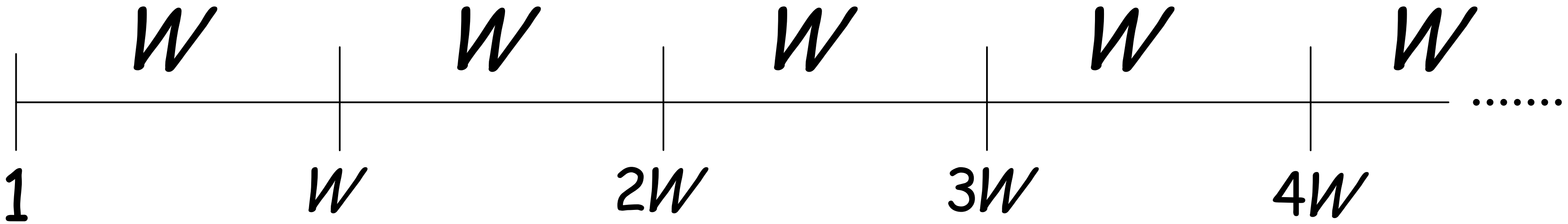


For simplicity assume unit valued elements

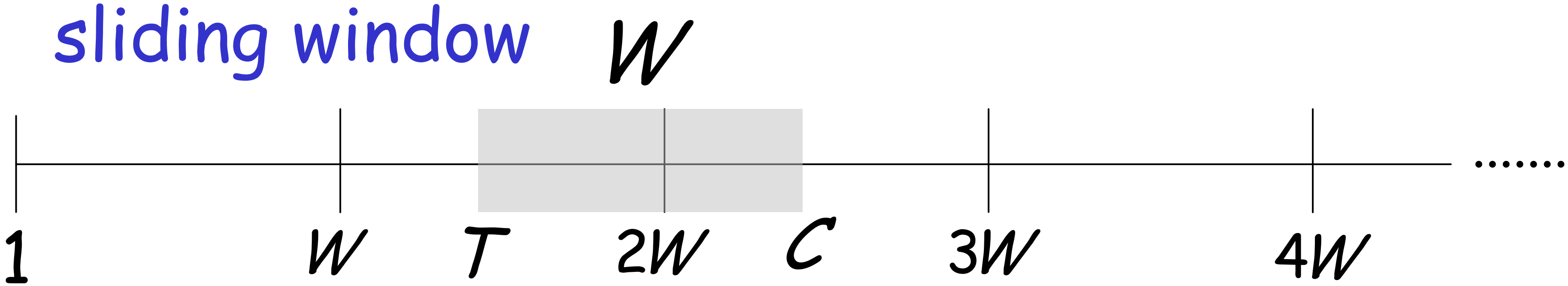
Most recent time window of duration W



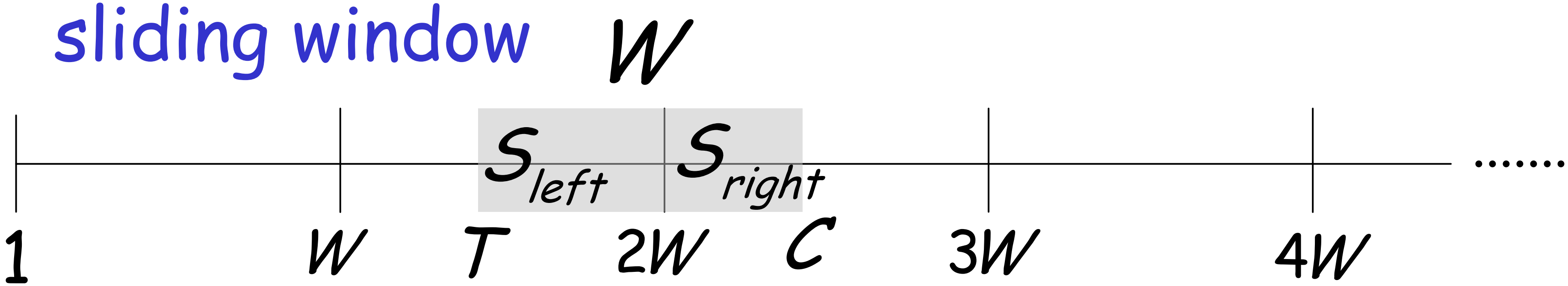
Goal: Compute the sum of elements with time stamps in time window $[C - W, C]$



Divide time into periods of duration W



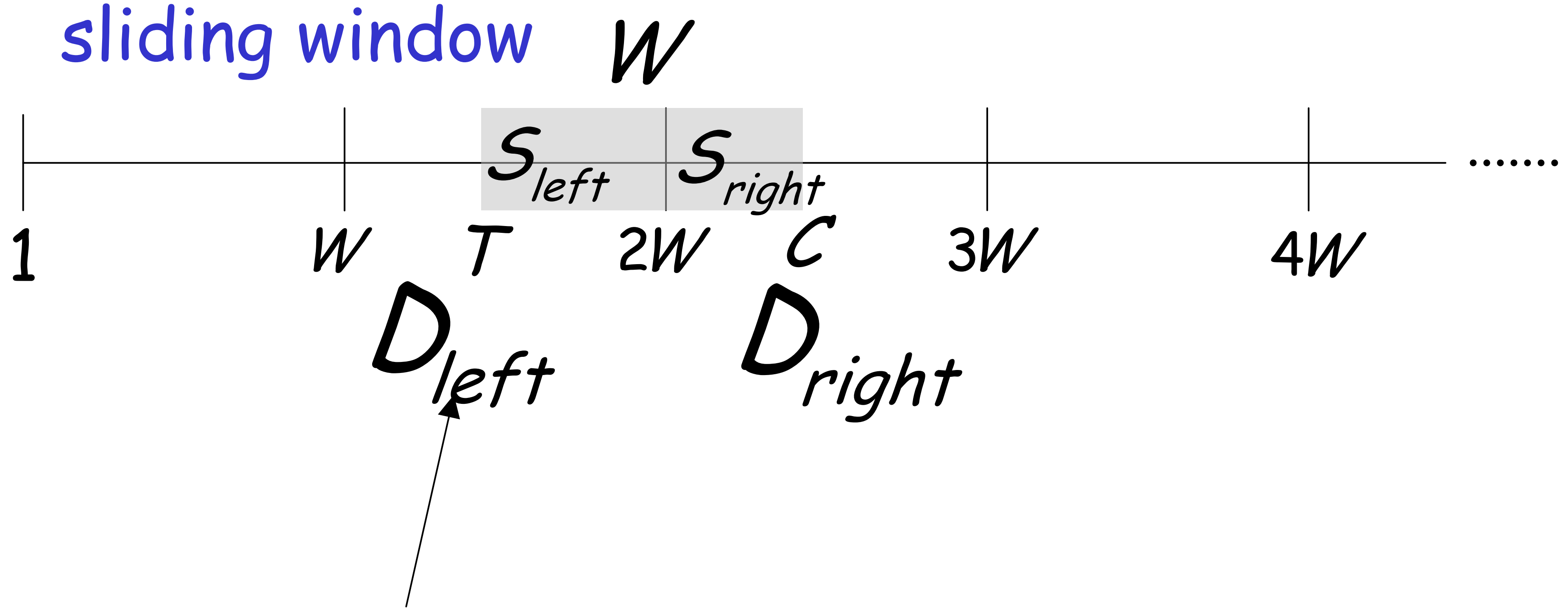
The sliding window may span at most two time periods sliding window



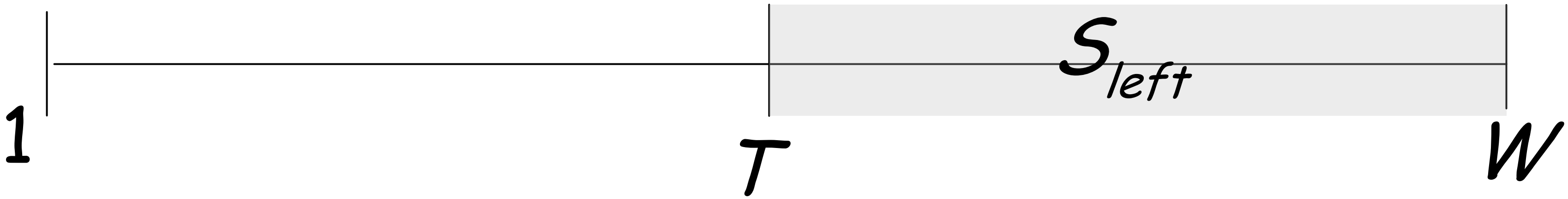
$$S = S_1 + S_2$$

Sum can be written as two sub-sums

In two time periods



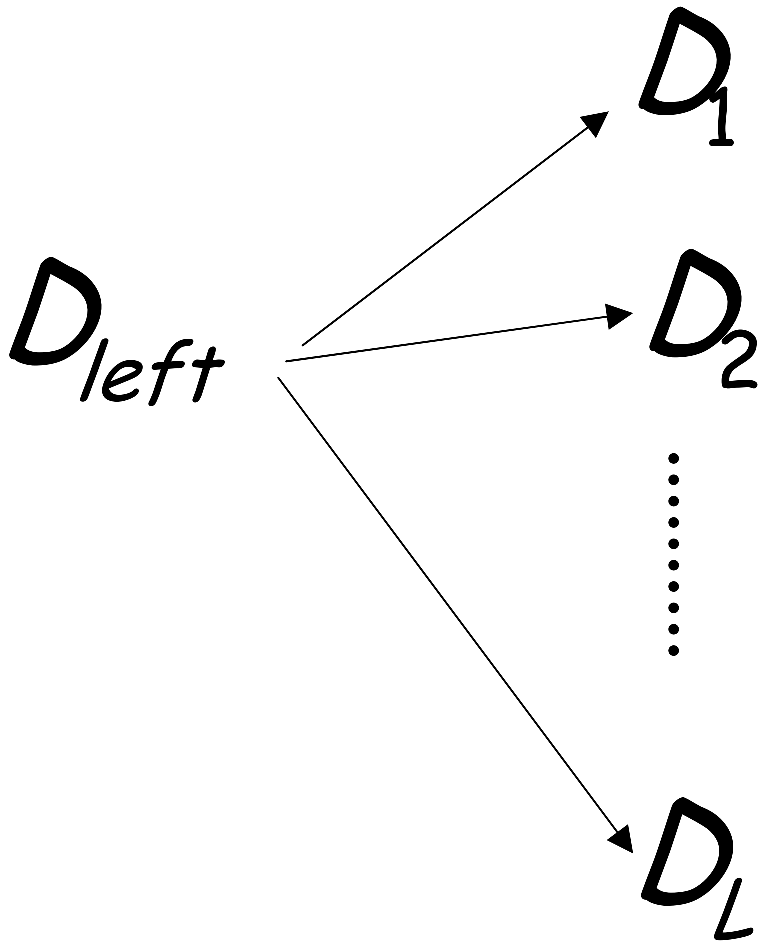
Data structure that maintains an estimate of S_{left} In left time period



D_{left}

Without loss of Generality,
Consider data structure D_{left}
in time period $[1, W]$

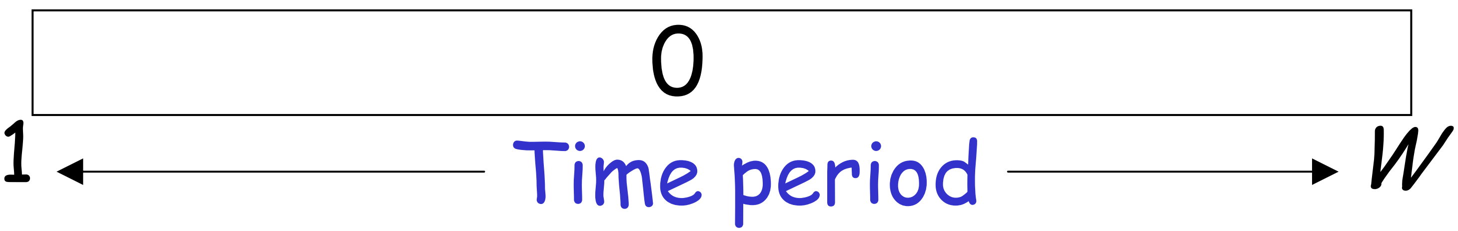
Data structure consists of various levels



2^L is an upper bound of the sum in a period

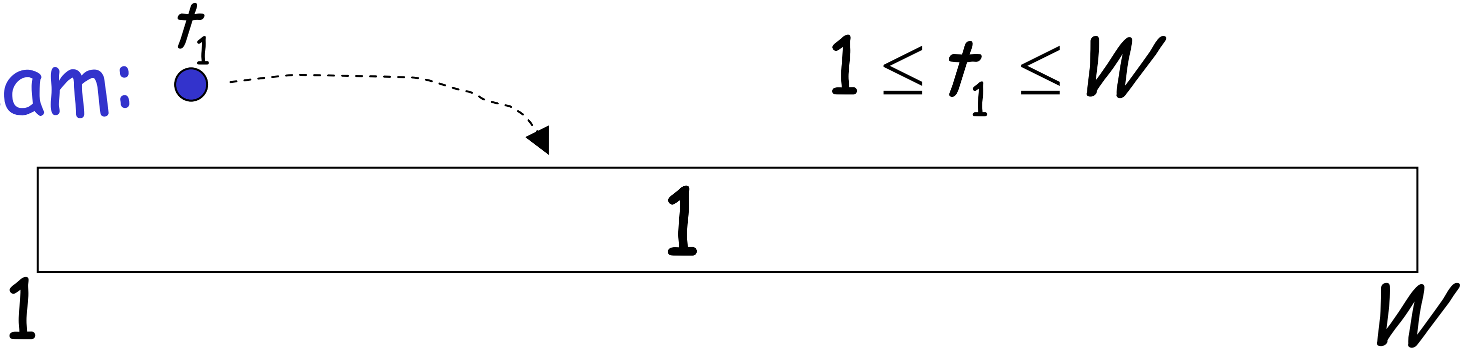
Consider level D_i

Bucket at Level $i+1$



Counts up to 2^{i+1} elements

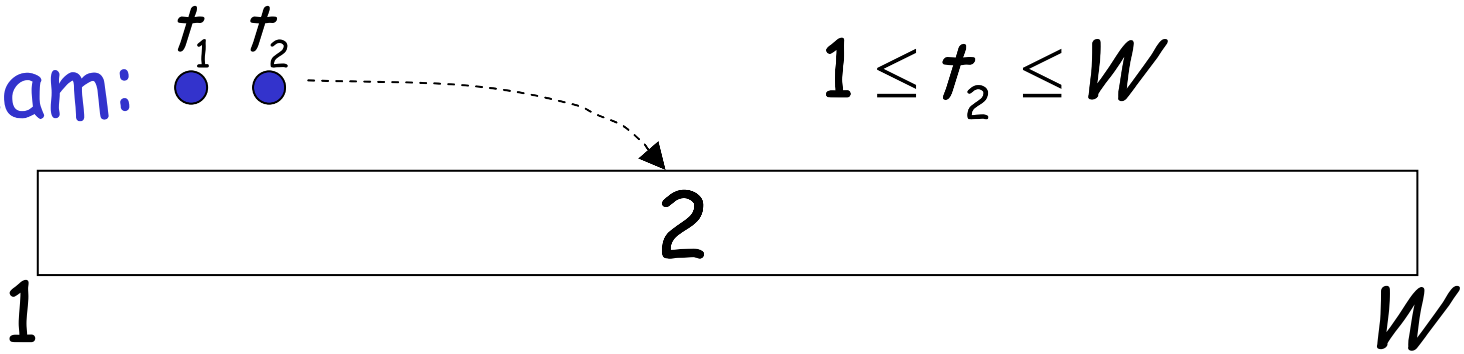
Stream:



Increase counter value

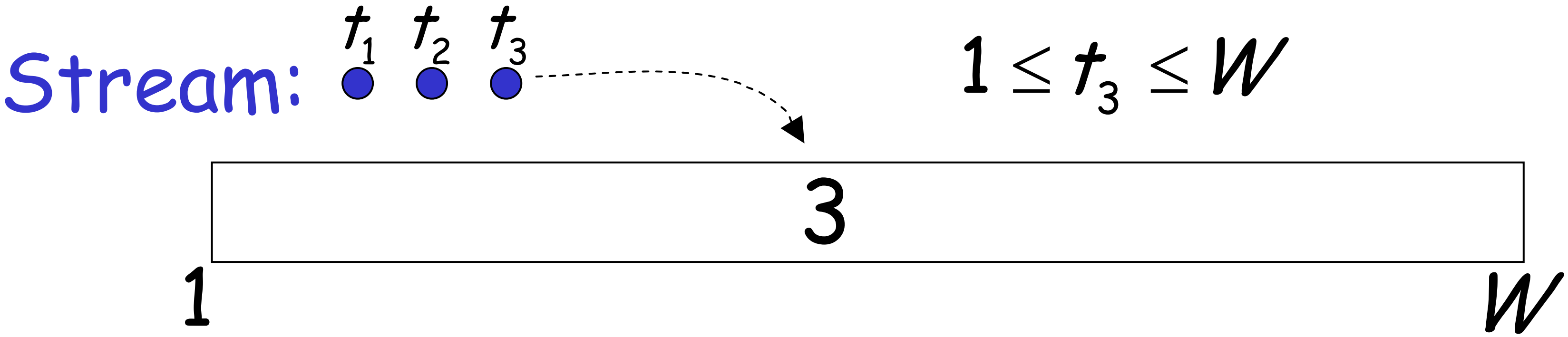
Stream:

$1 \le t_2 \le W$

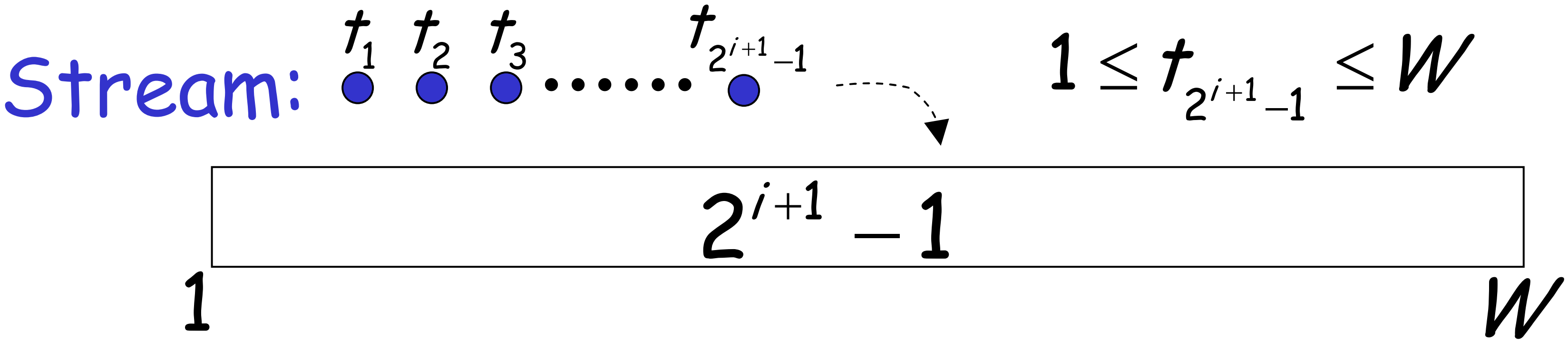


Increase counter value

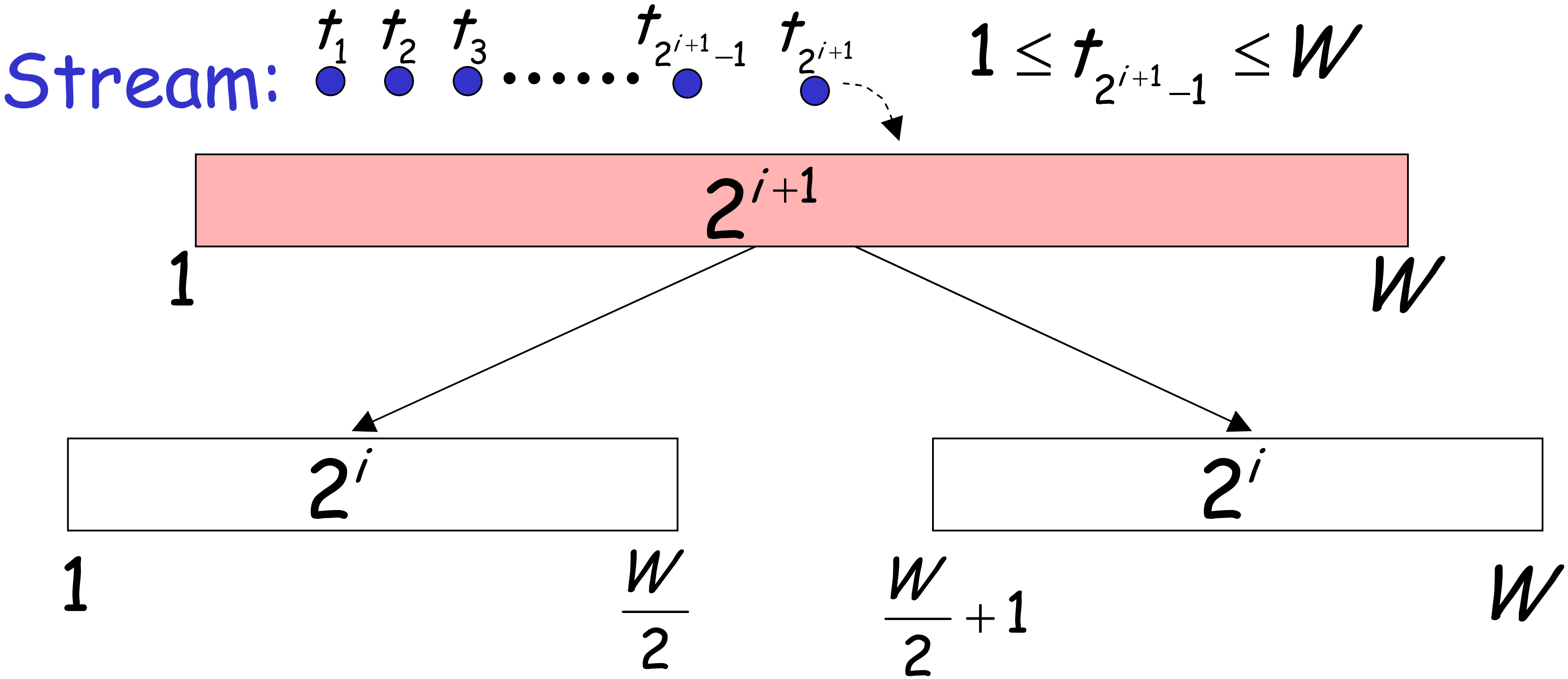
$1 \le t_3 \le W$



Increase counter value



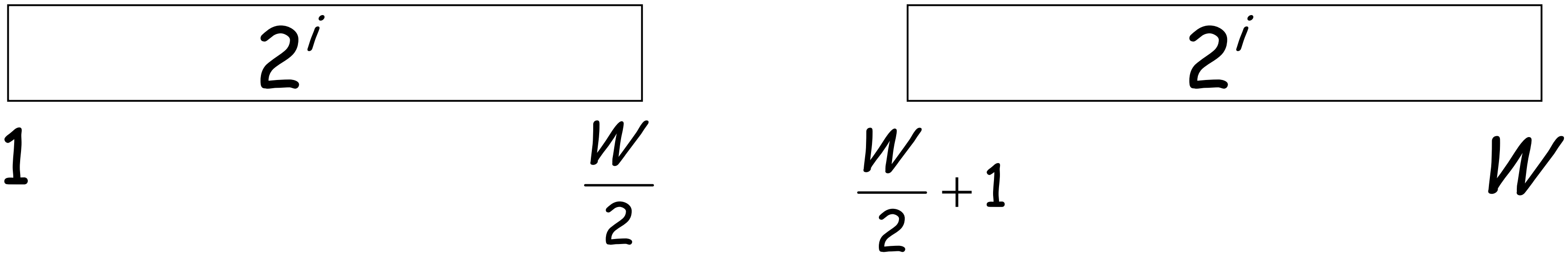
Increase counter value



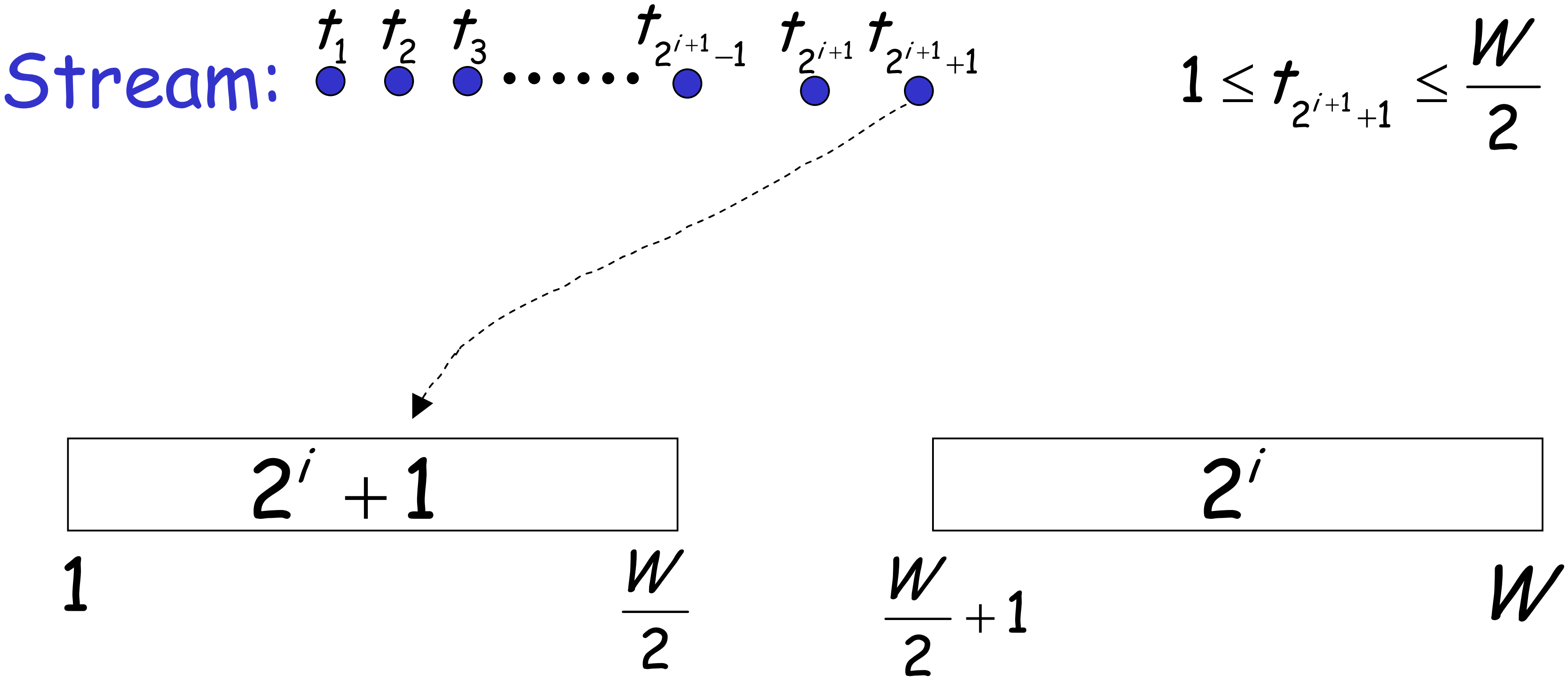
Split bucket

Counter threshold of 2^{i+1} reached

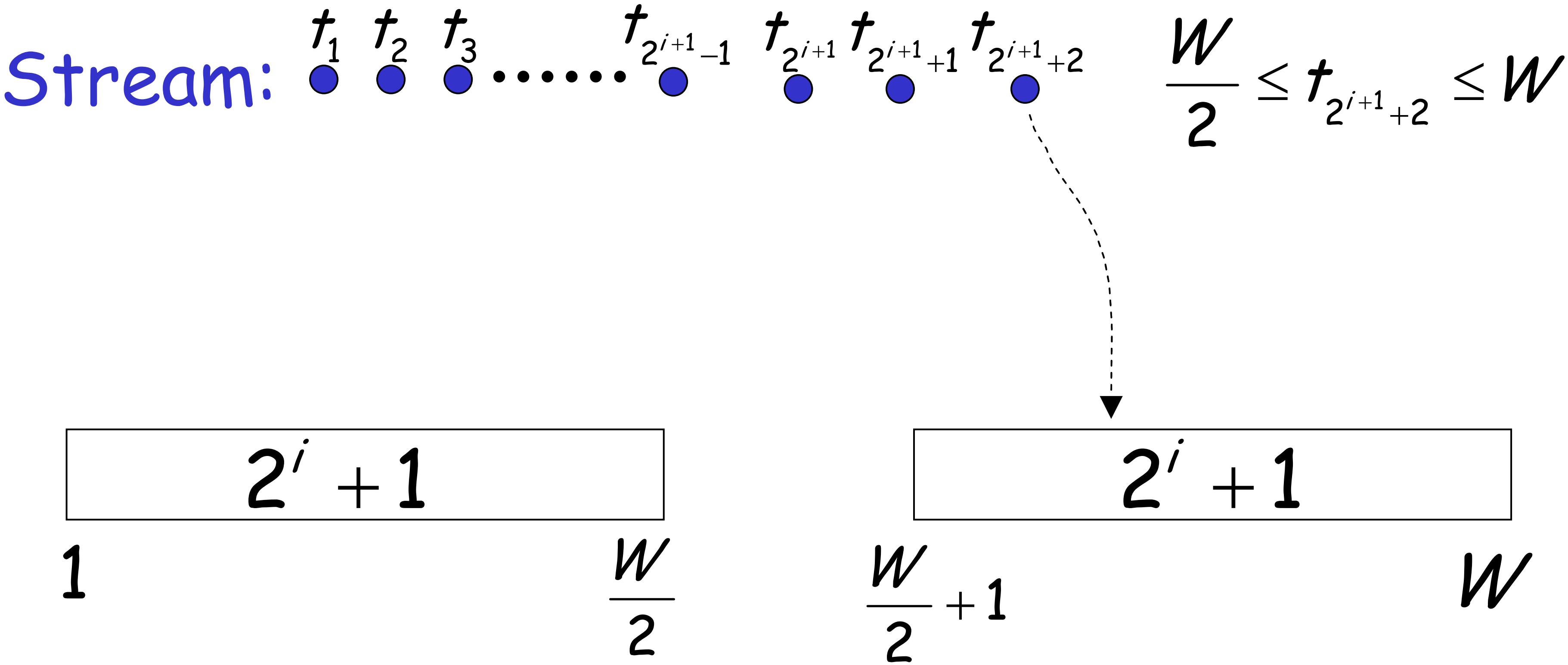
Stream: $t_1 \; t_2 \; t_3 \; \ldots\ldots \; t_{2^{i+1}-1} \; t_{2^{i+1}}$ $\quad 1 \le t_{2^{i+1}-1} \le W$



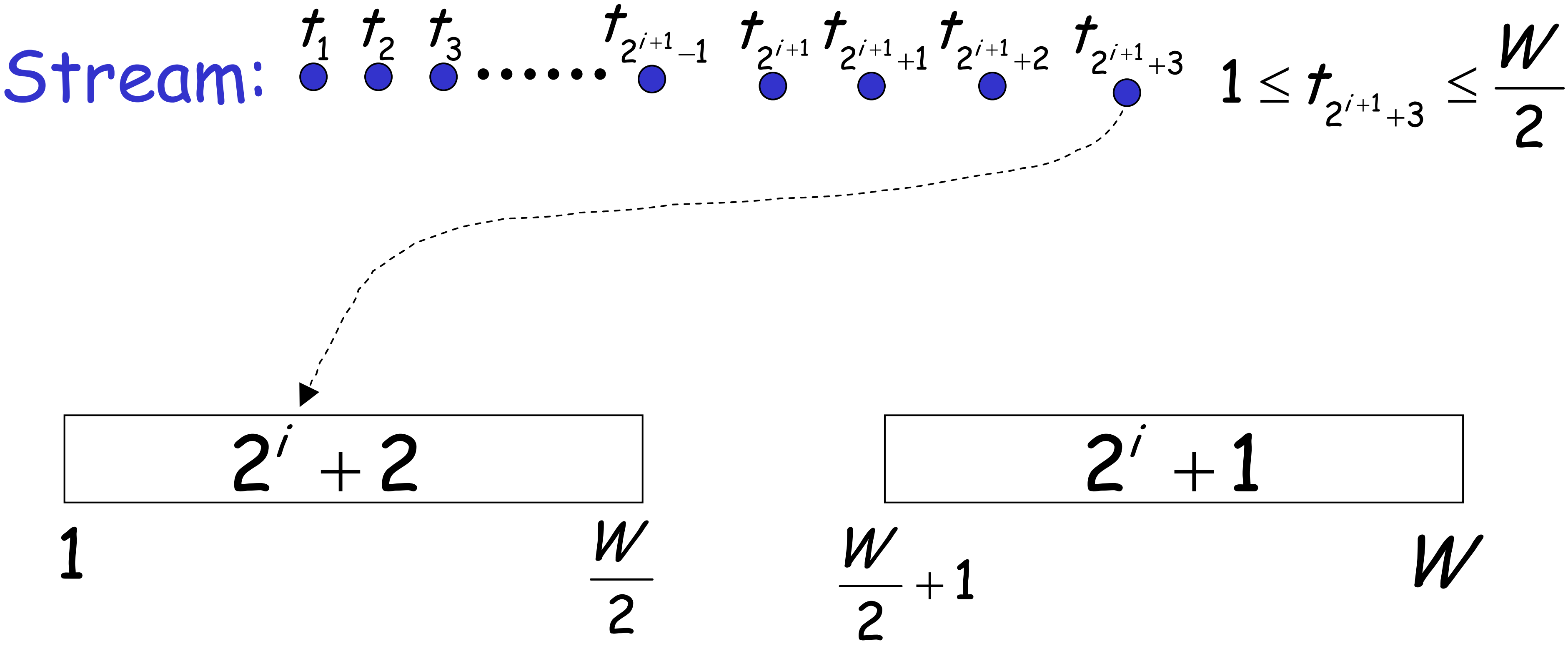
New buckets have threshold also 2^{i+1}



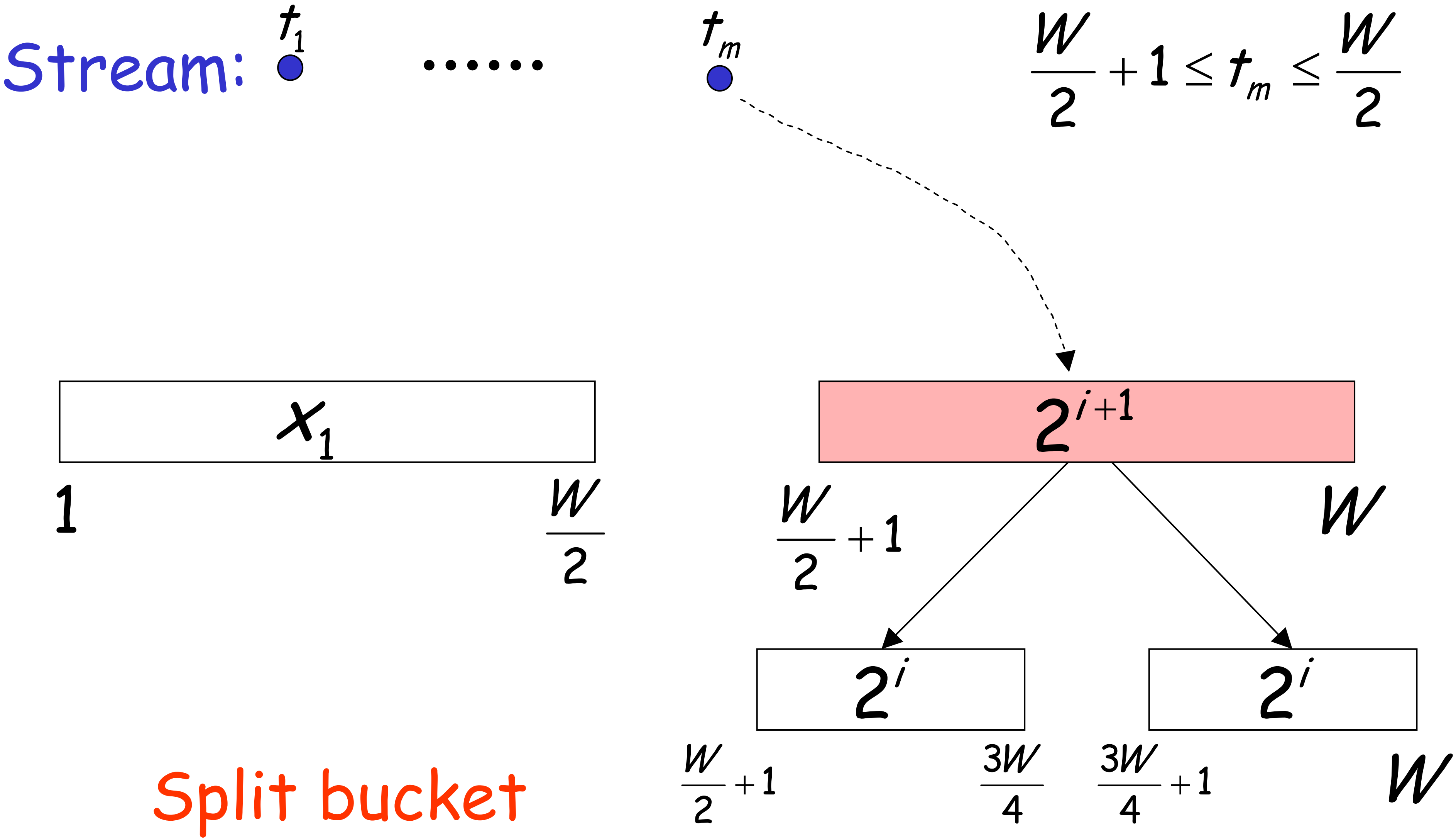
Increase appropriate bucket



Increase appropriate bucket

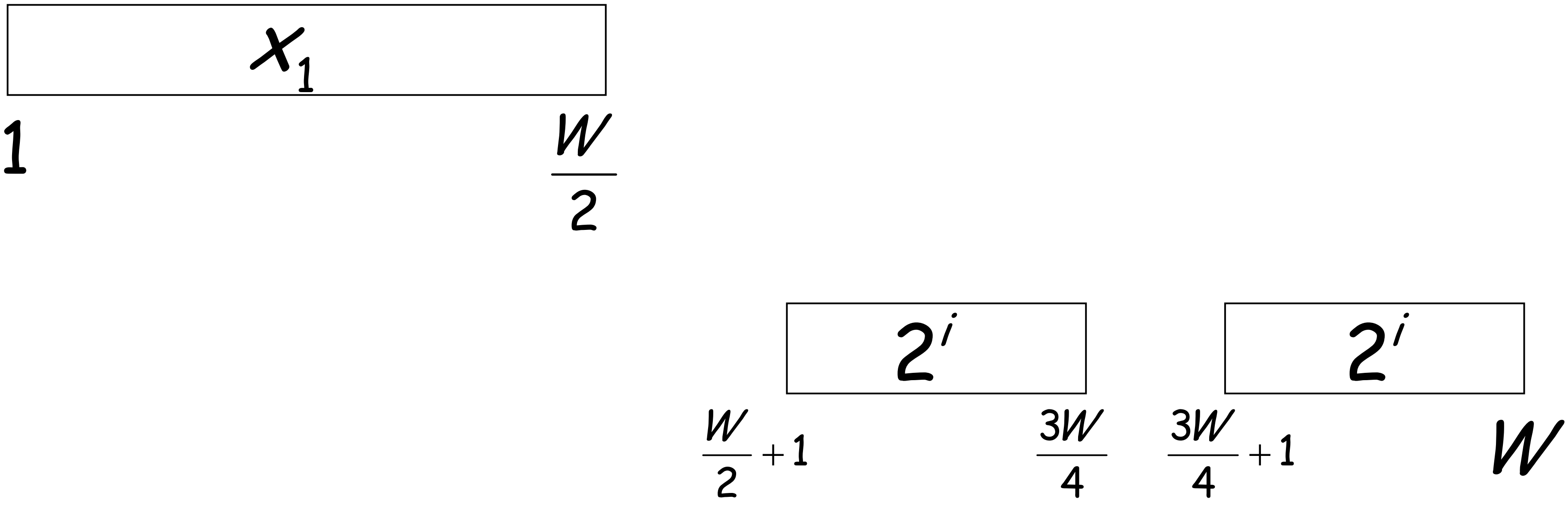


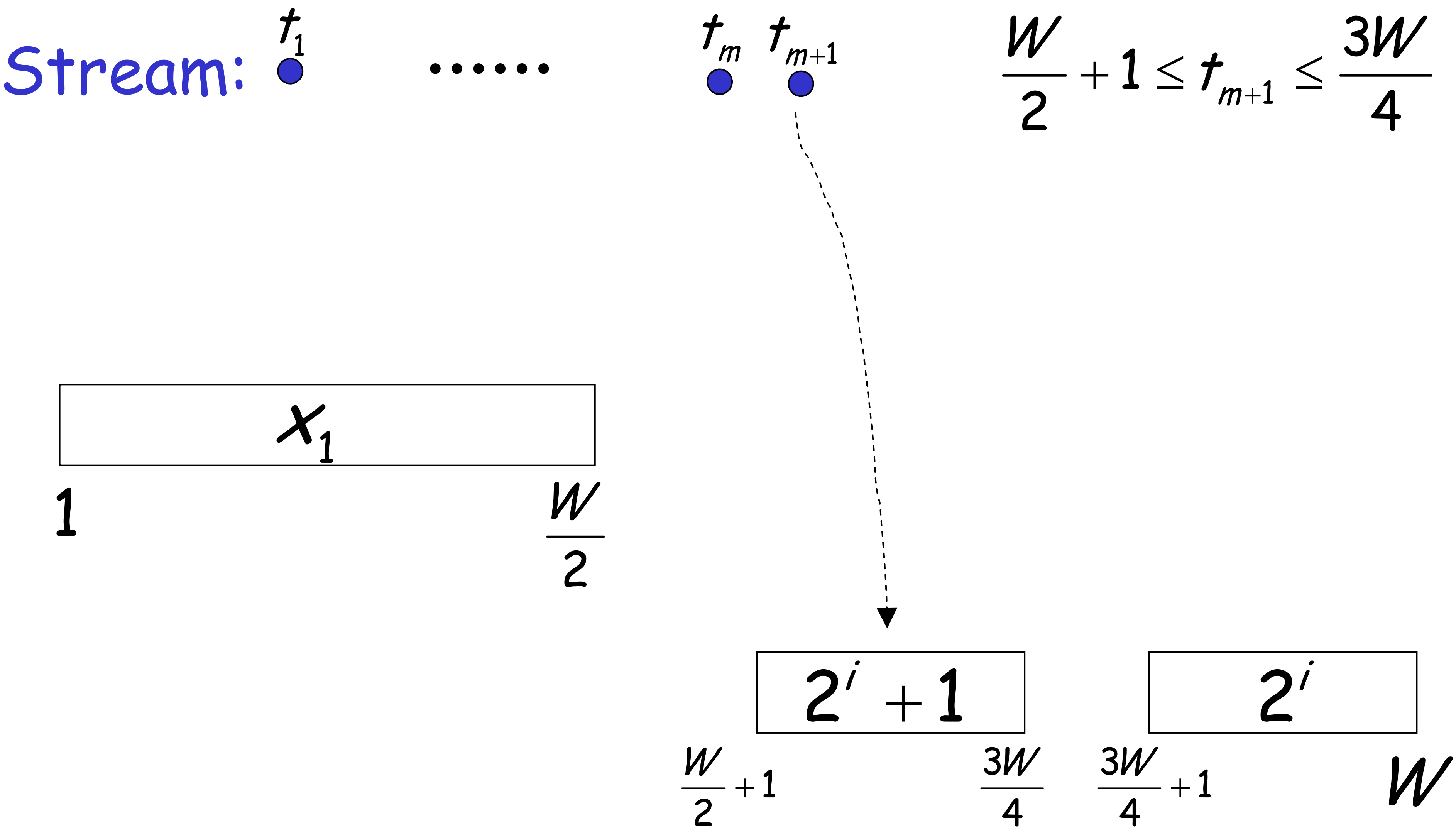
Increase appropriate bucket



Split bucket

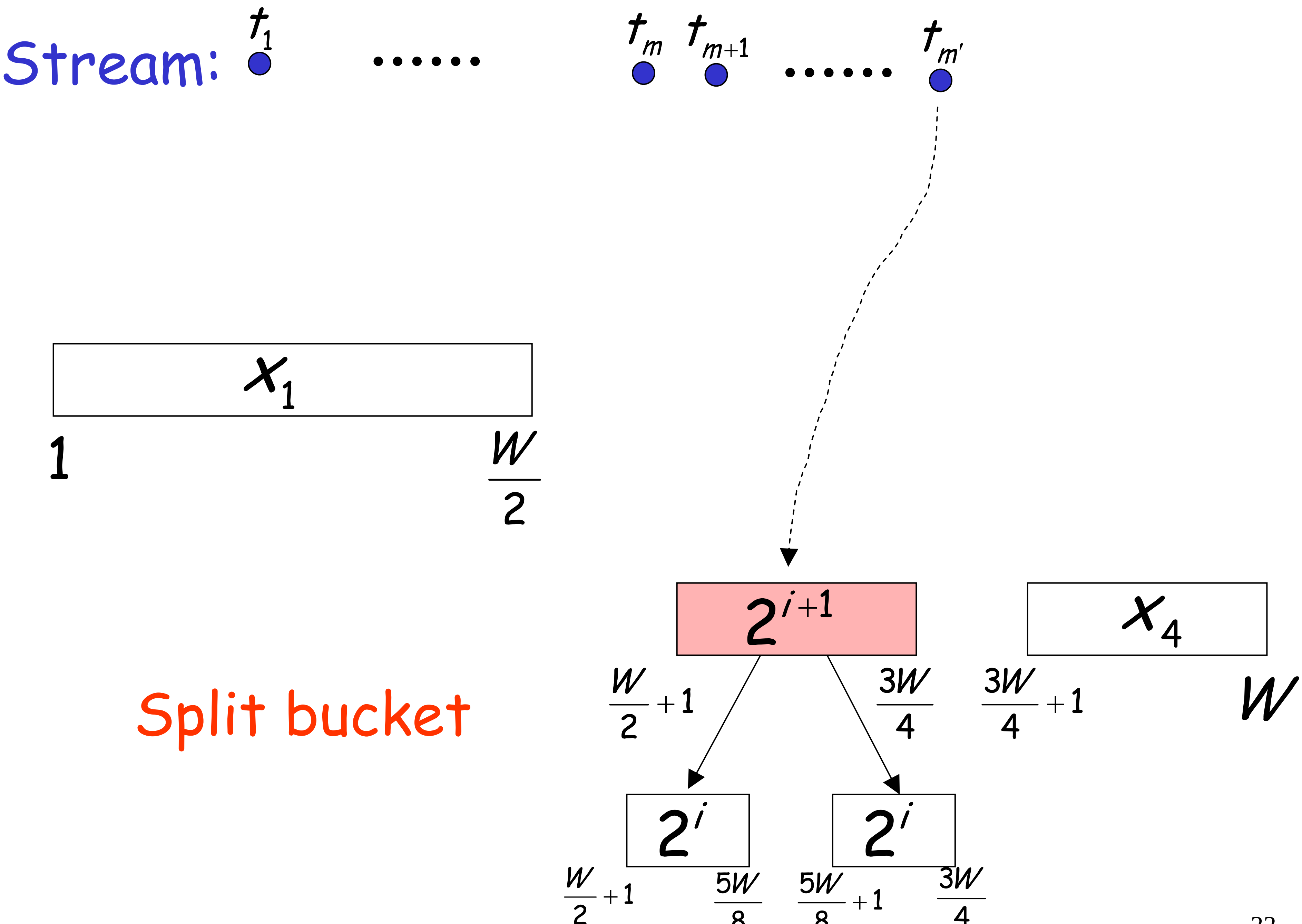
Stream: t_1 …… t_m

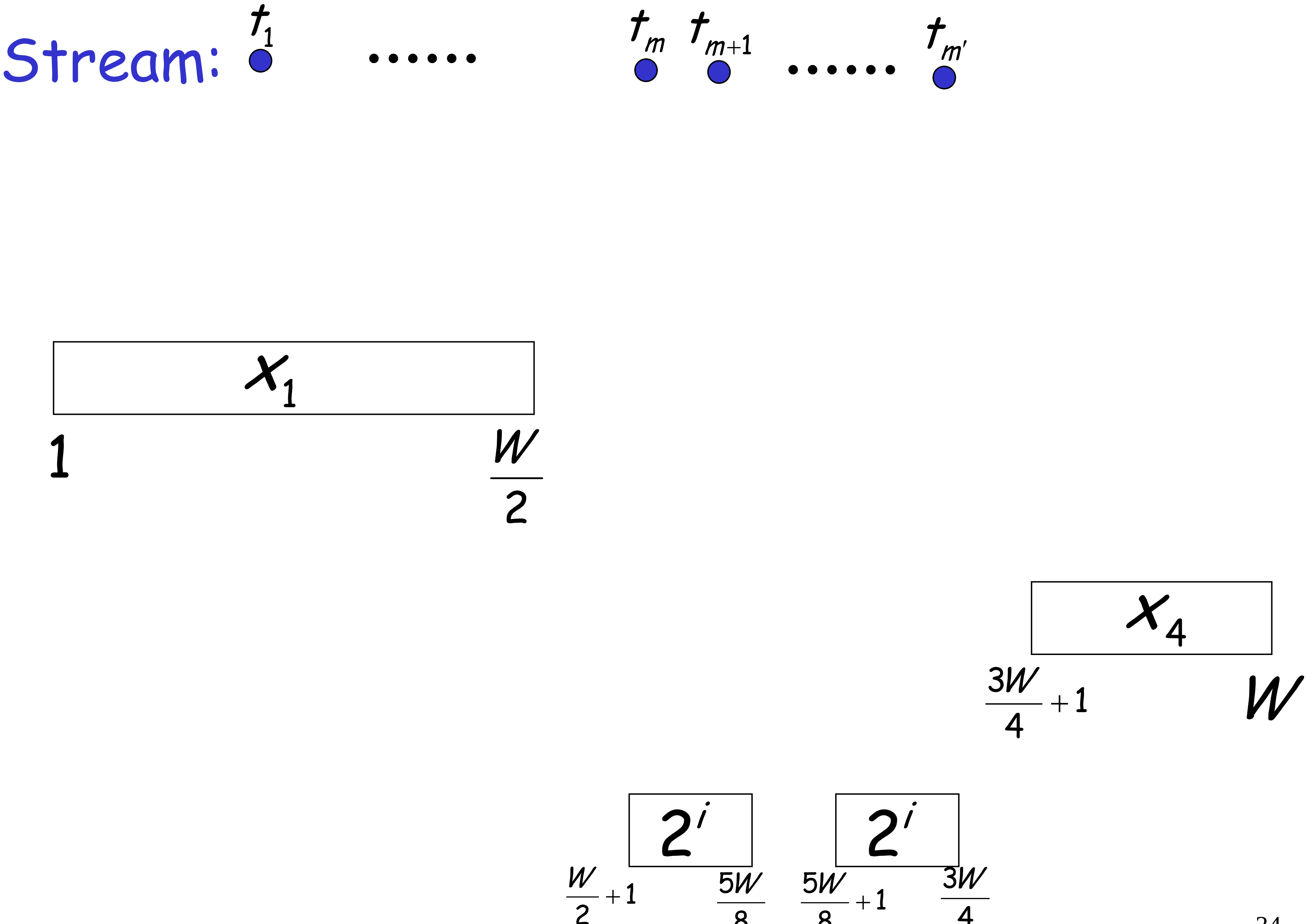




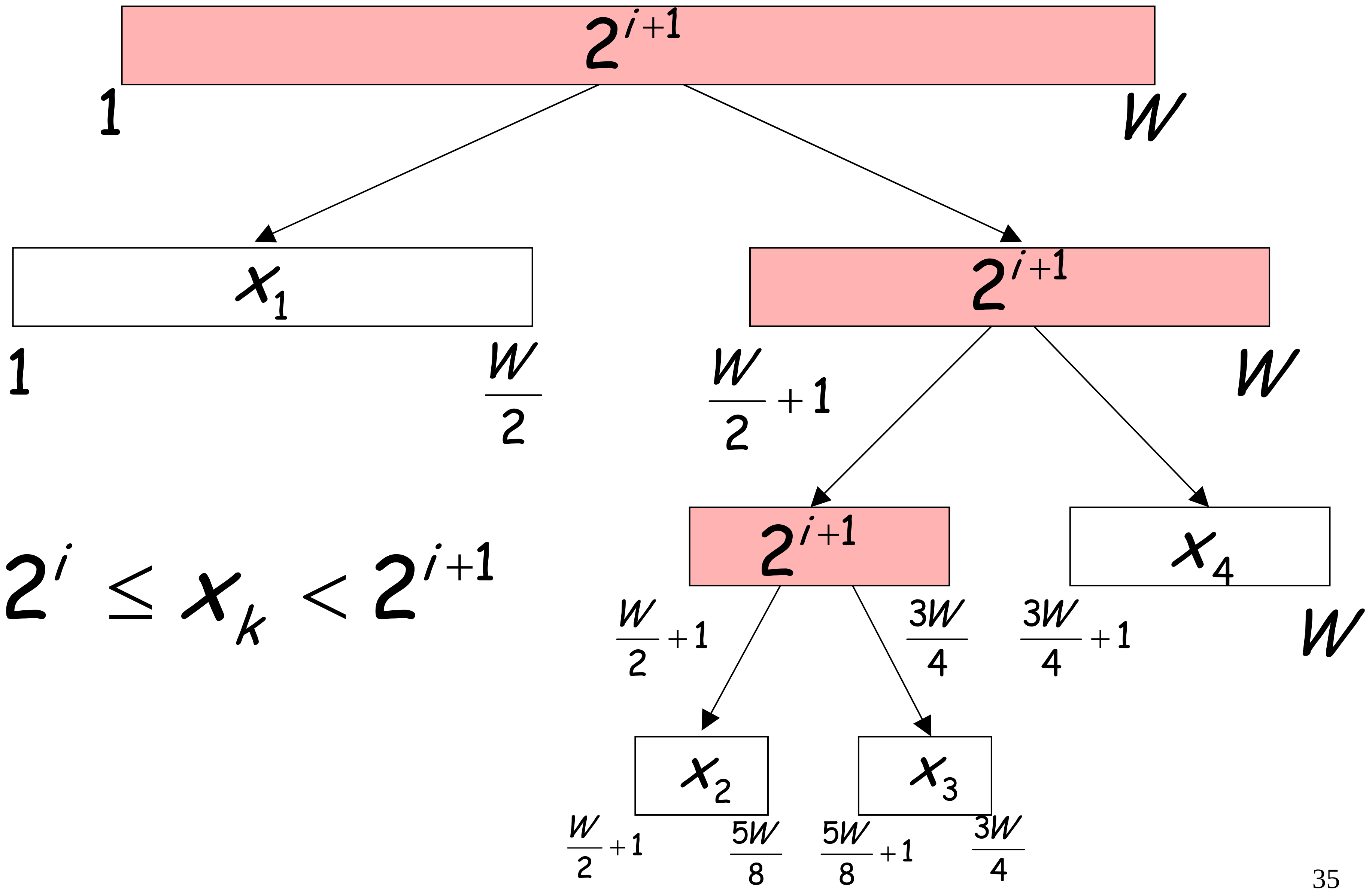
$$\frac{W}{2}+1 \le t_{m+1} \le \frac{3W}{4}$$

Increase appropriate bucket

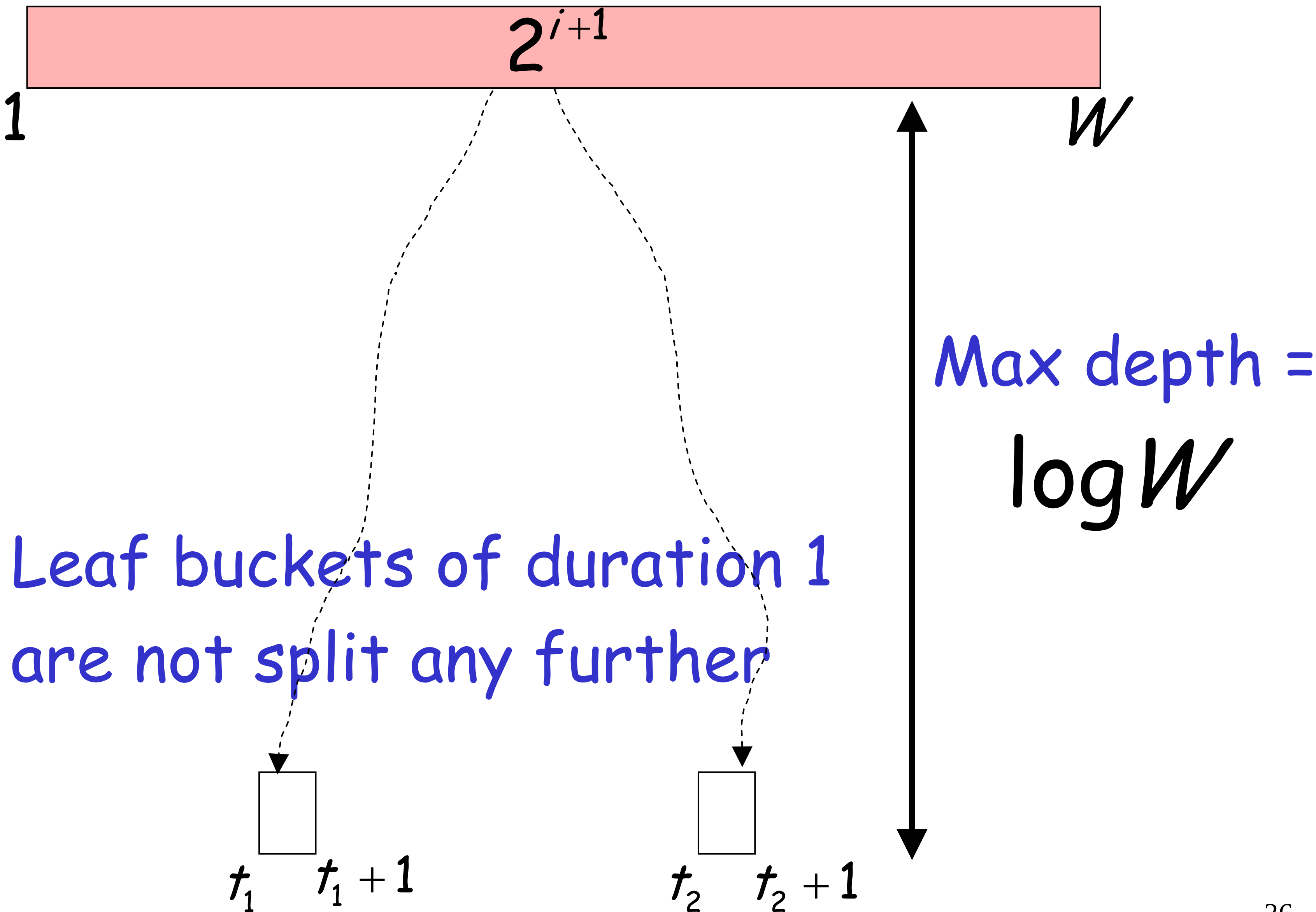

Stream:
t_1
t_m t_{m+1}
$t_{m'}$
x_1
1
$\frac{W}{2}$
2^{i+1}
x_4
$\frac{W}{2}+1$
$\frac{3W}{4}$
$\frac{3W}{4}+1$
W
Split bucket
2^i
2^i
$\frac{W}{2}+1$
$\frac{5W}{8}$
$\frac{5W}{8}+1$
$\frac{3W}{4}$

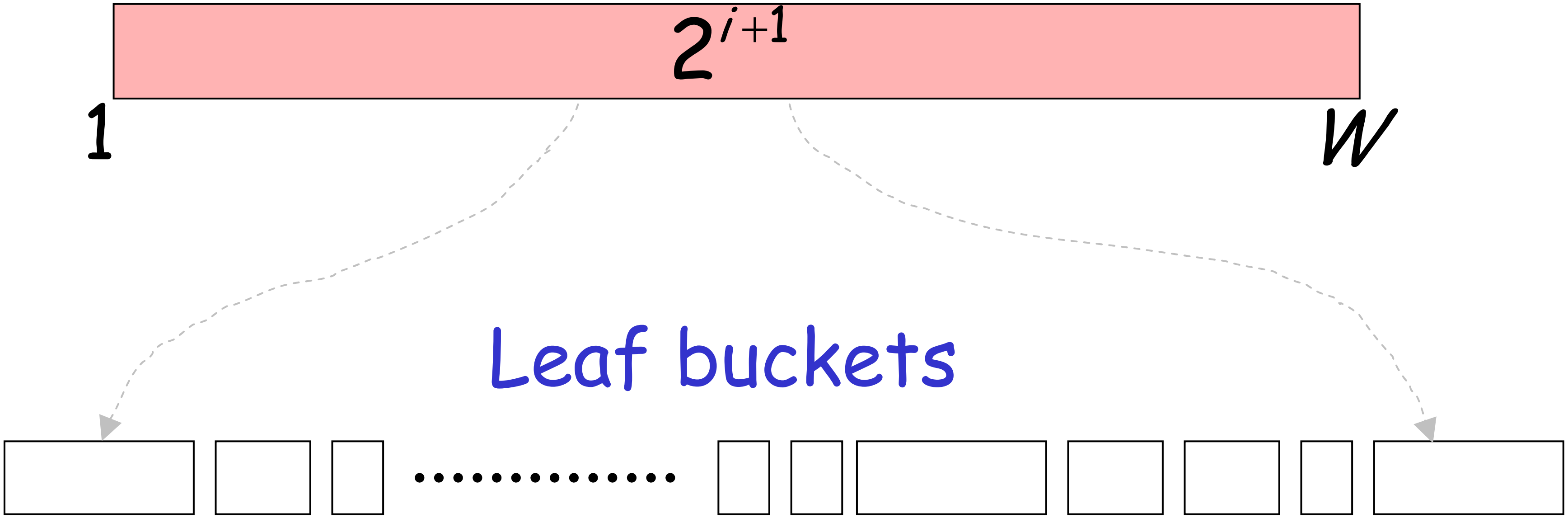

Stream:
t_1
t_m
t_{m+1}
$t_{m'}$
x_1
1
$\frac{W}{2}$
x_4
$\frac{3W}{4}+1$
W
2^i
2^i
$\frac{W}{2}+1$
$\frac{5W}{8}$
$\frac{5W}{8}+1$
$\frac{3W}{4}$

Splitting Tree

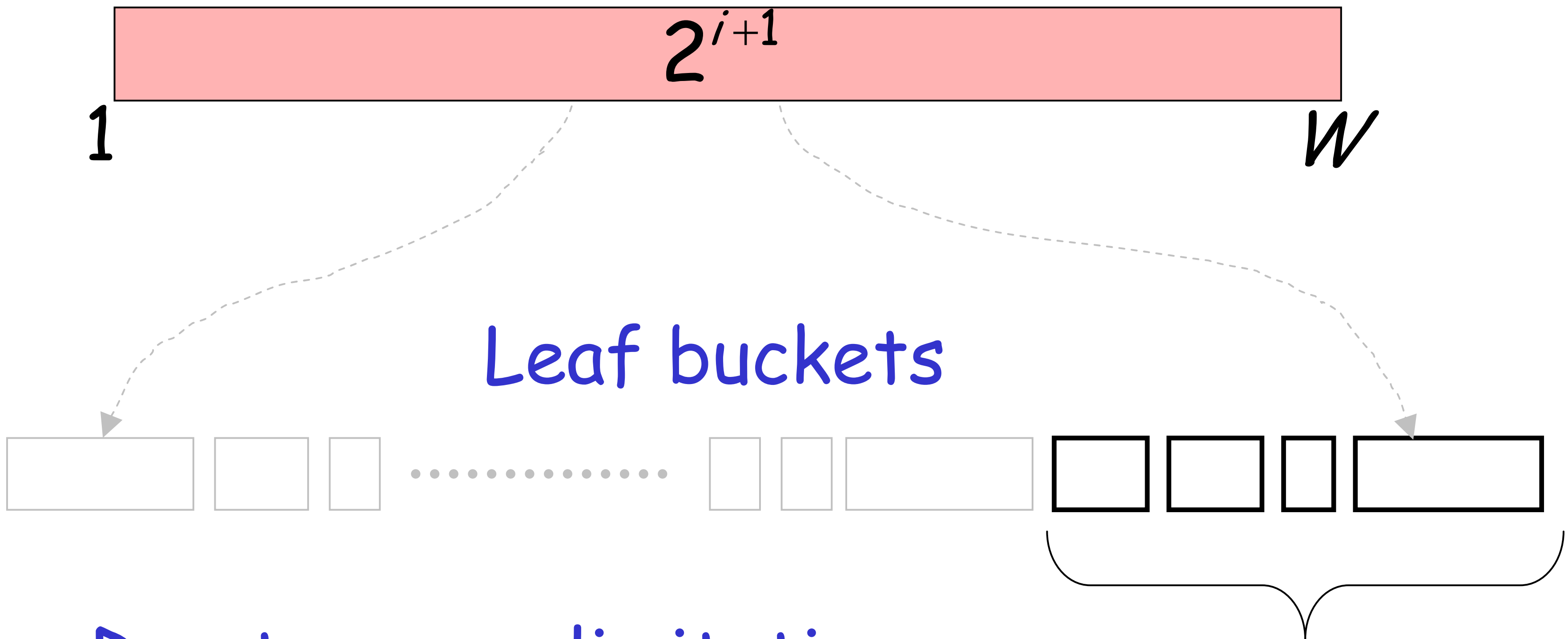


$$2^i \le x_k < 2^{i+1}$$


2^{i+1}
1
W
Max depth =
$\log W$
Leaf buckets of duration 1
are not split any further
t_1
$t_1 + 1$
t_2
$t_2 + 1$

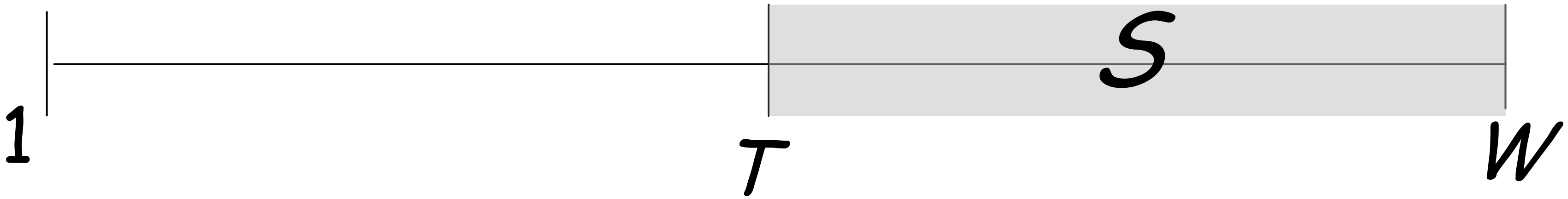


The initial bucket may be split into many buckets

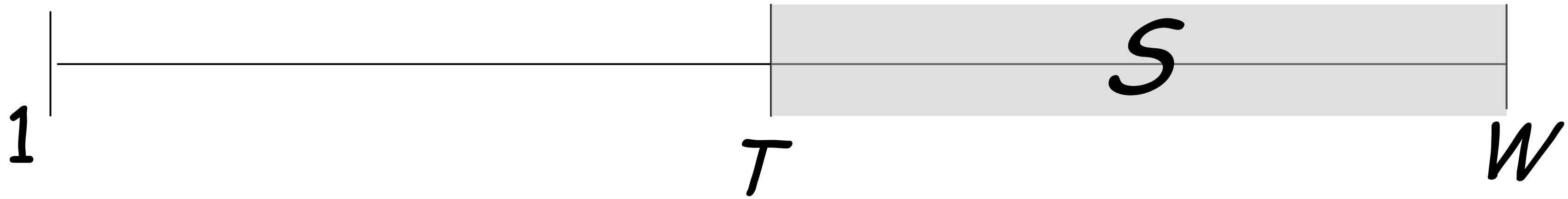


Due to space limitations we only keep the last buckets

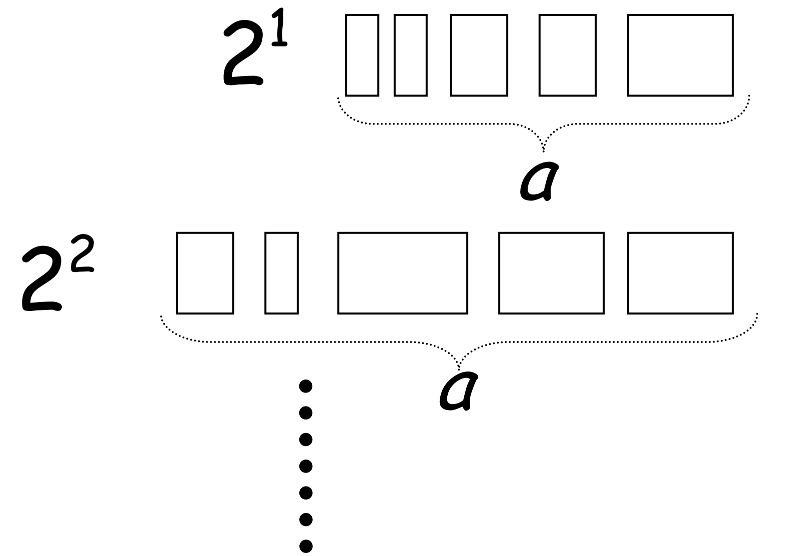
$$a = \frac{2+\varepsilon}{\varepsilon}\log W$$

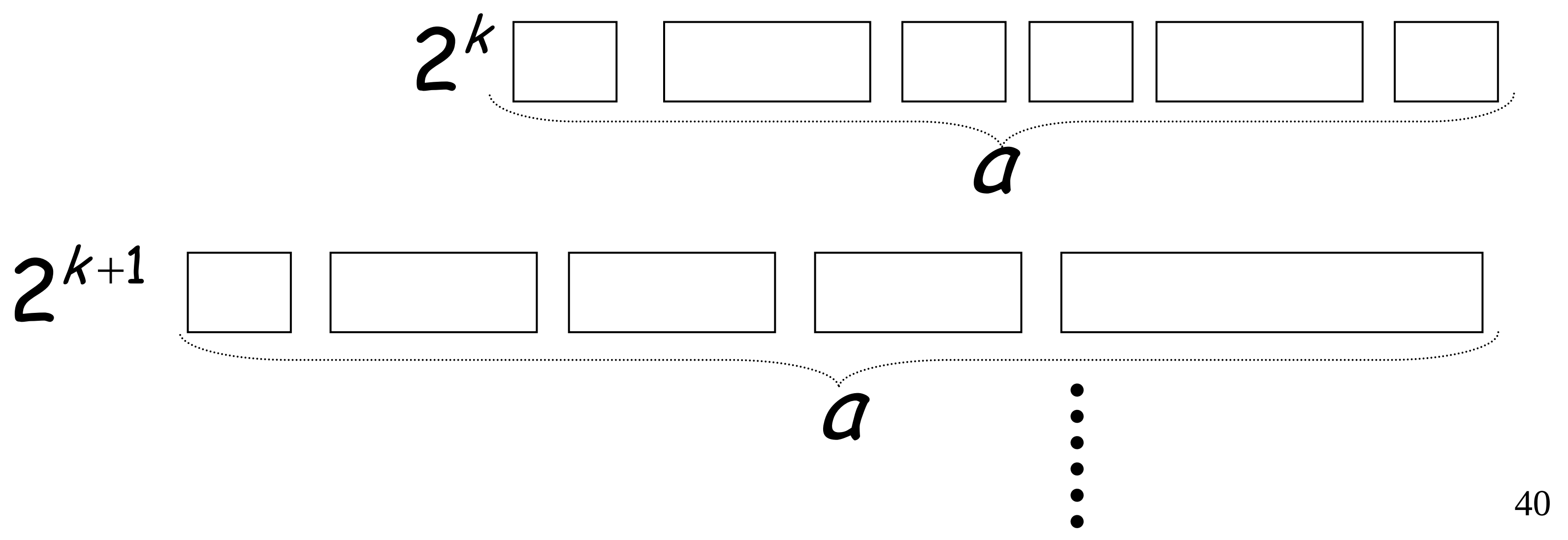


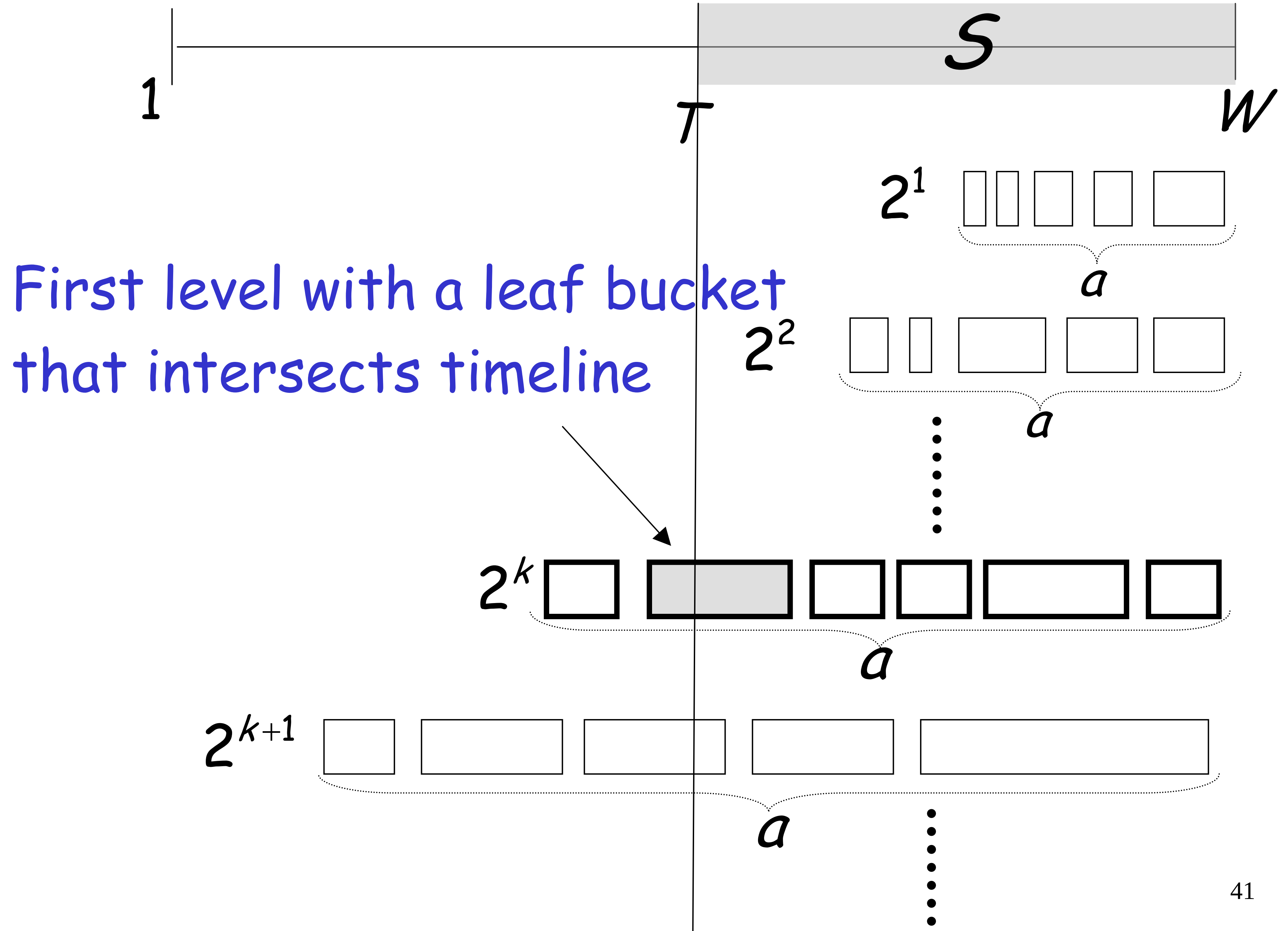
Suppose we want to find the sum S of elements in time period $[T, W]$



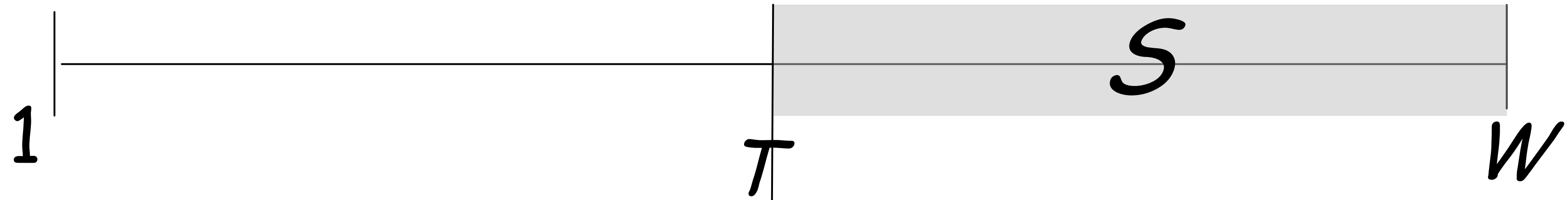
Consider various levels of splitting threshold





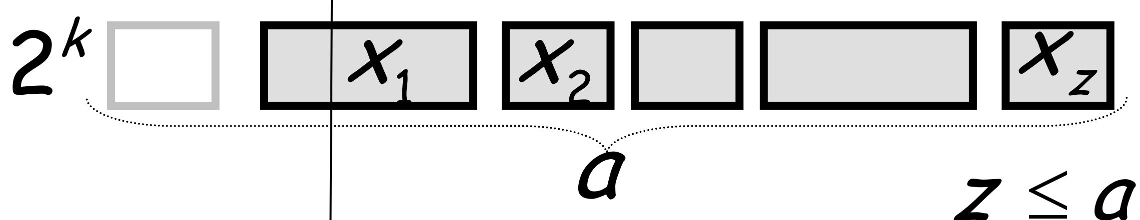


S
1
T
W
2^1
a
First level with a leaf bucket that intersects timeline
2^2
a
2^k
a
2^{k+1}
a



Estimate of S:

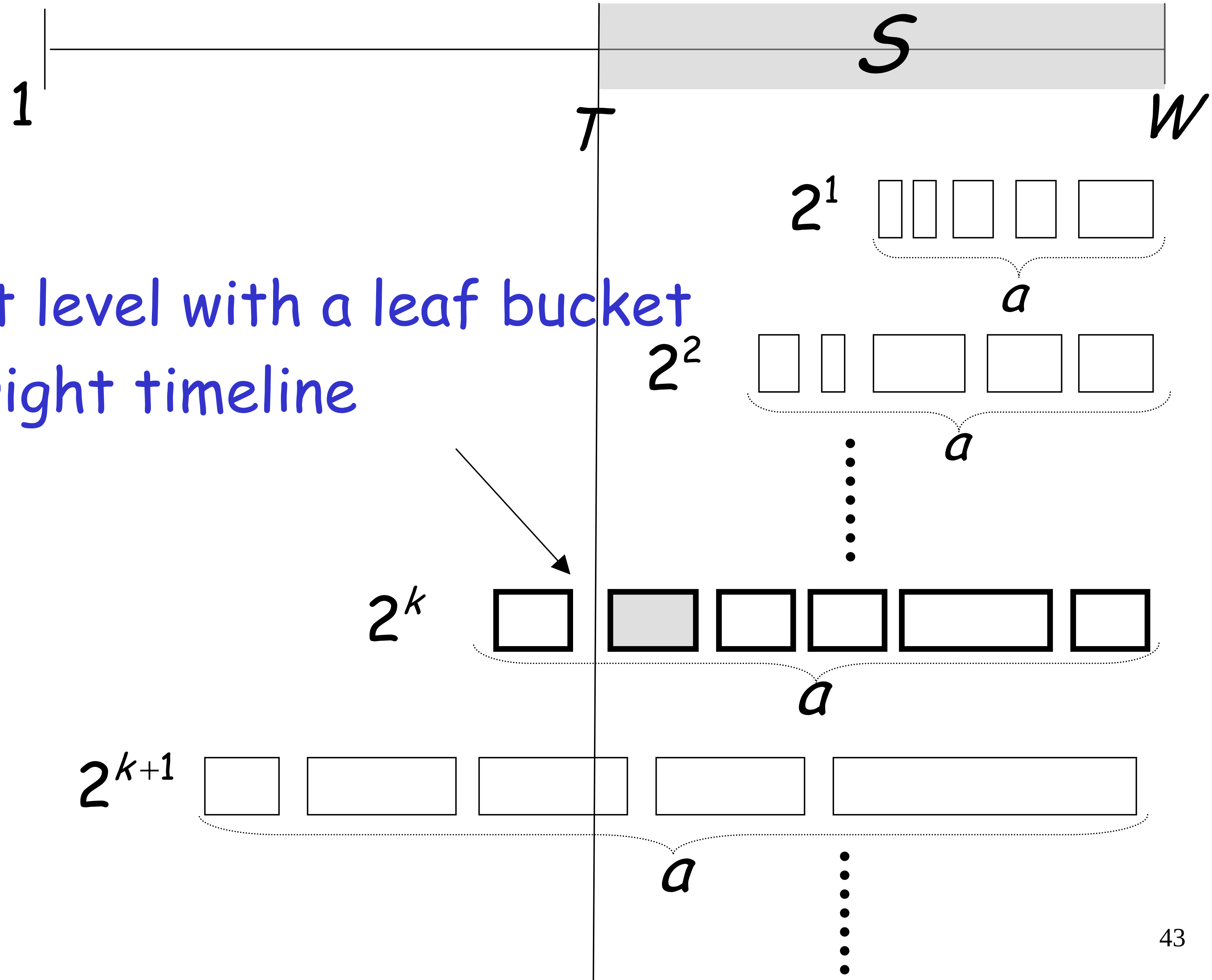
$$X = x_1 + x_2 + \cdots + x_z$$



Consider buckets on right of timeline

OR

First level with a leaf bucket
On right timeline

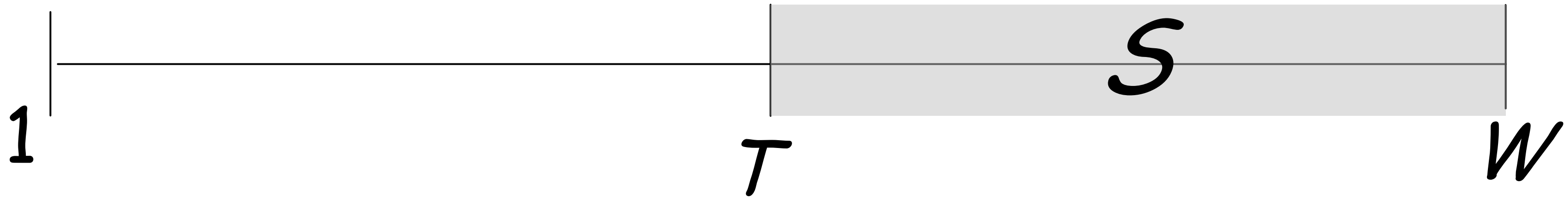


Outline of Talk

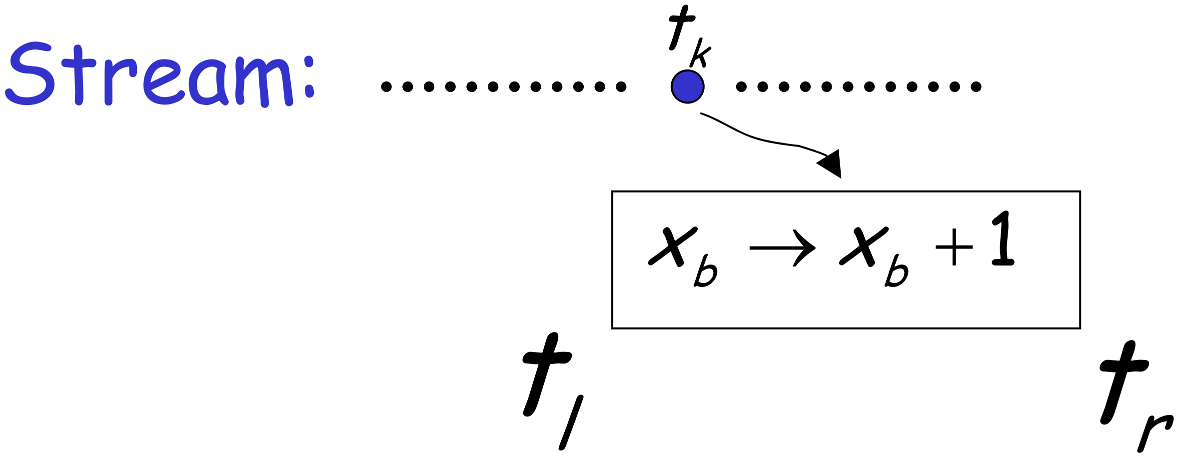
Introduction

Algorithm

Analysis

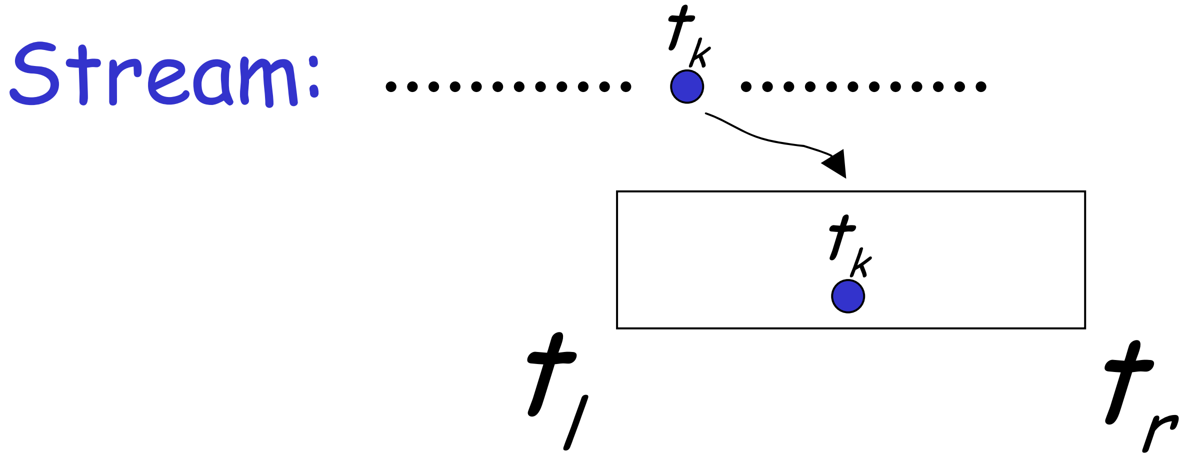


Suppose that we use level 2^{i+1} in order to compute the estimate



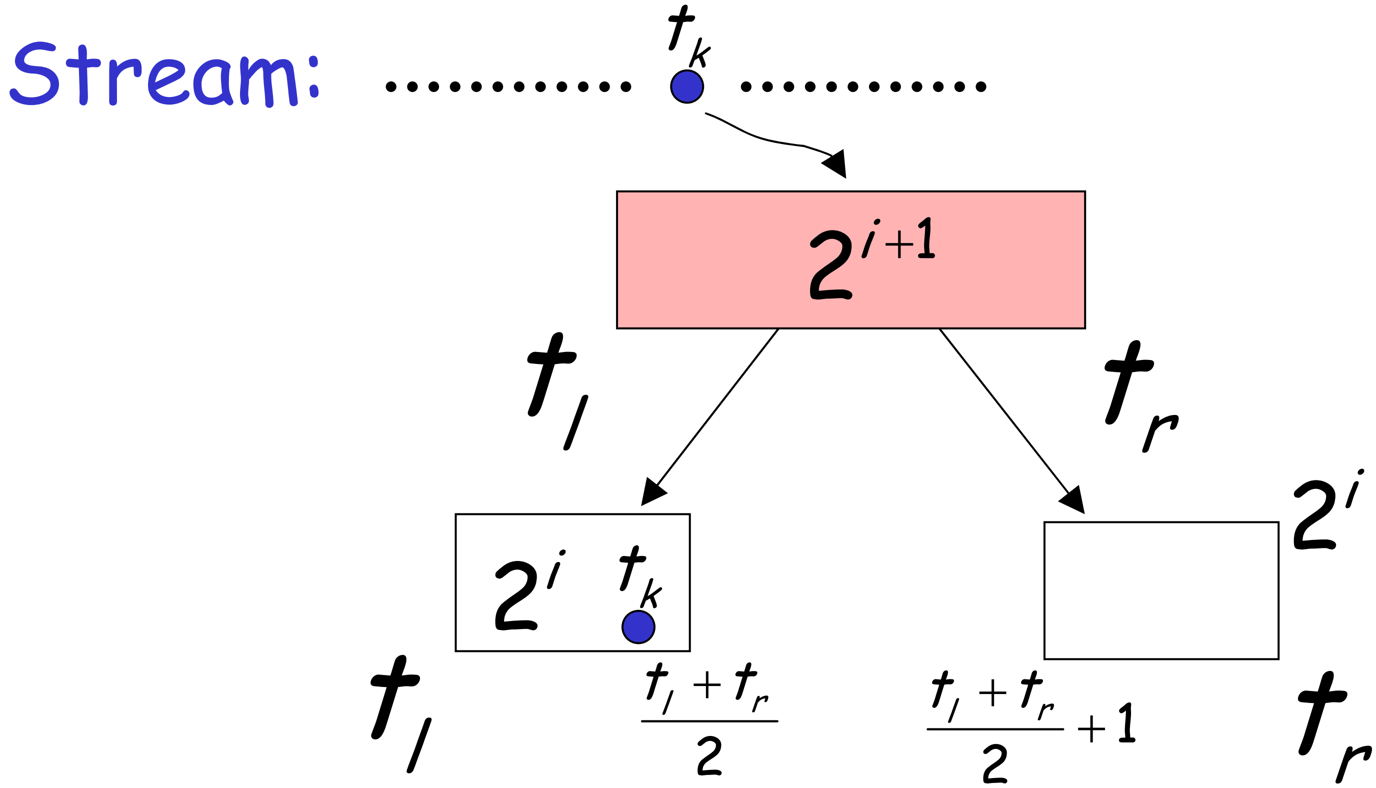
Consider splitting threshold level 2^{i+1}

A data element is counted in the appropriate bucket

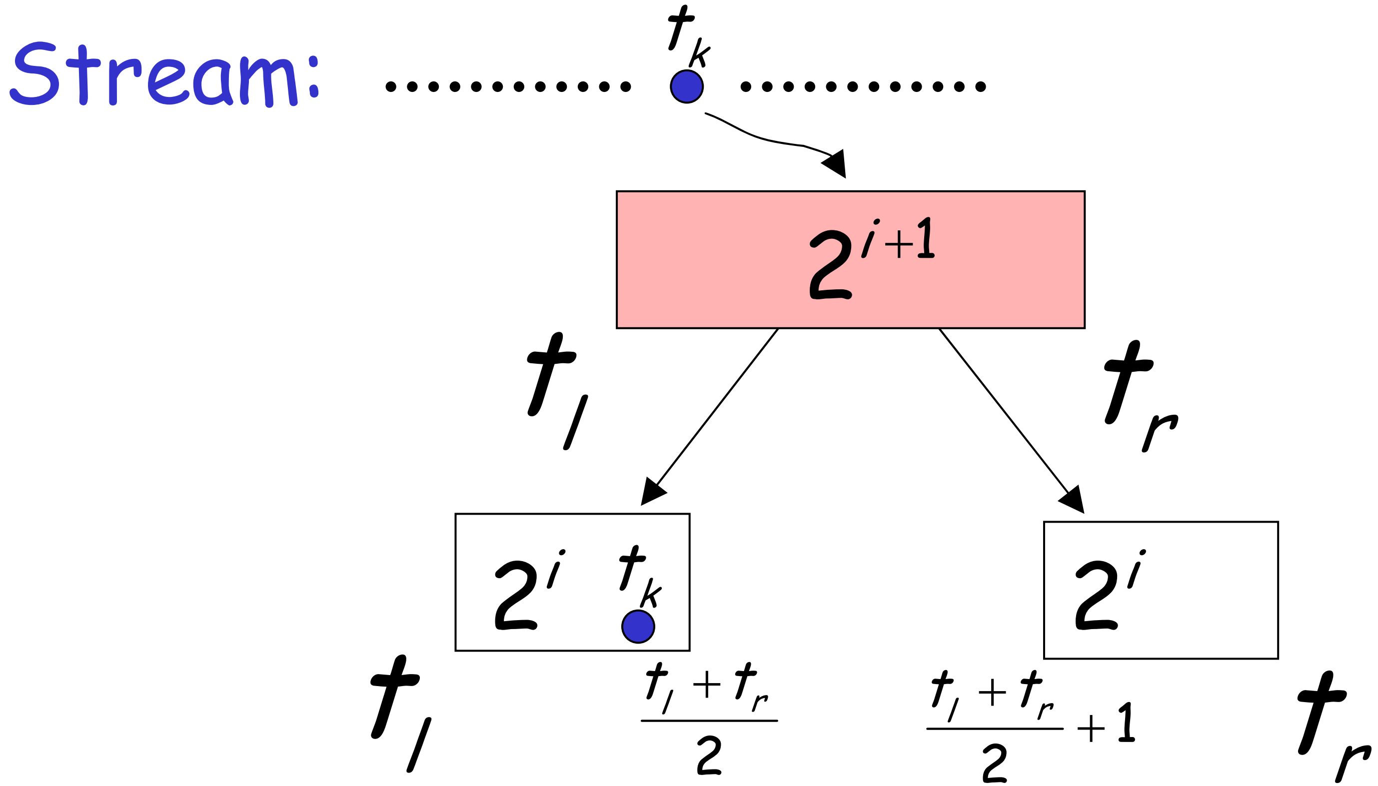


$t_l \leq t_k \leq t_r$

We can assume that the element is placed in the respective bucket



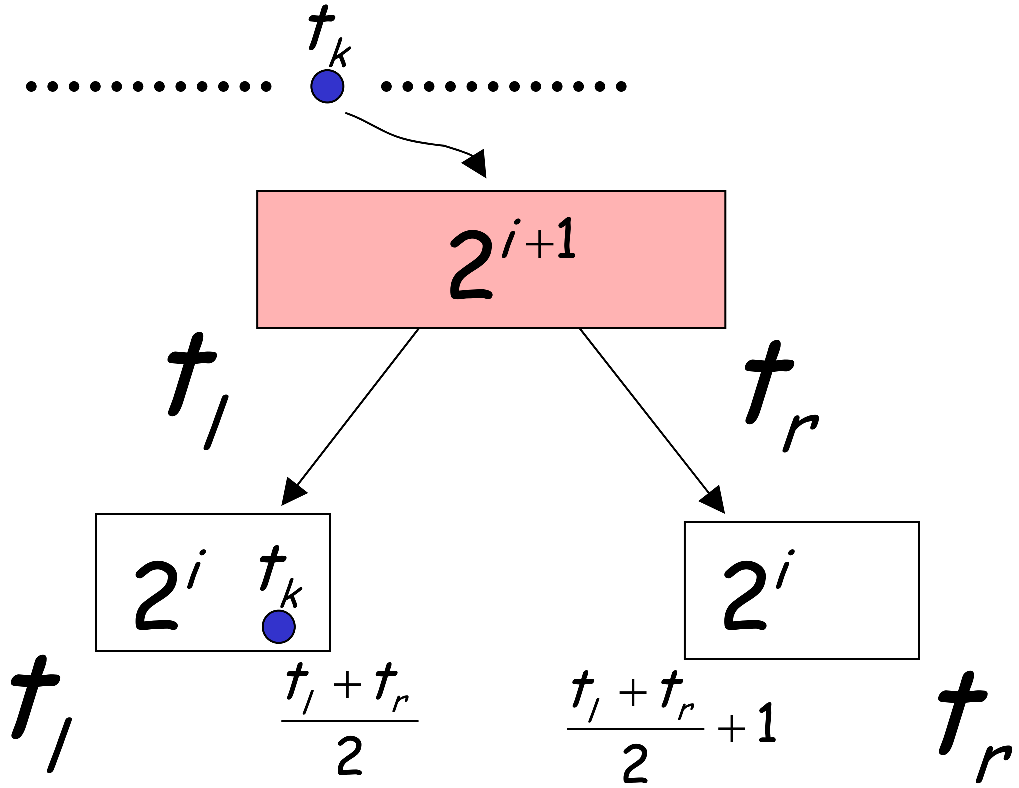
We can assume that when bucket splits the element is placed in an arbitrary child bucket



If: $t_l \le t_k \le \frac{t_l + t_r}{2}$ GOOD!

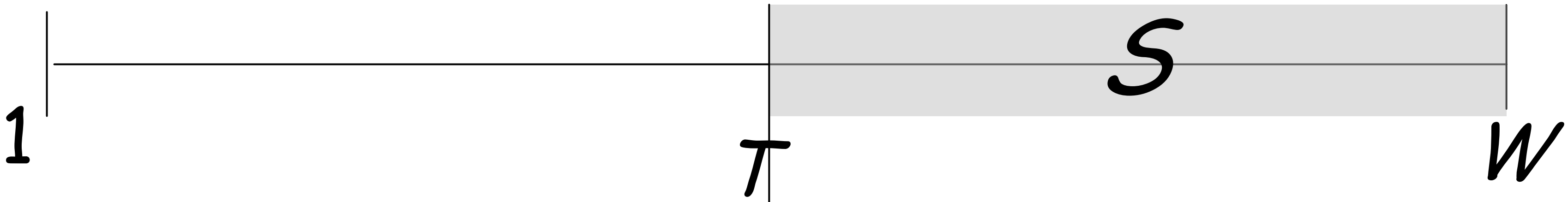
Element counted in correct bucket

Stream:

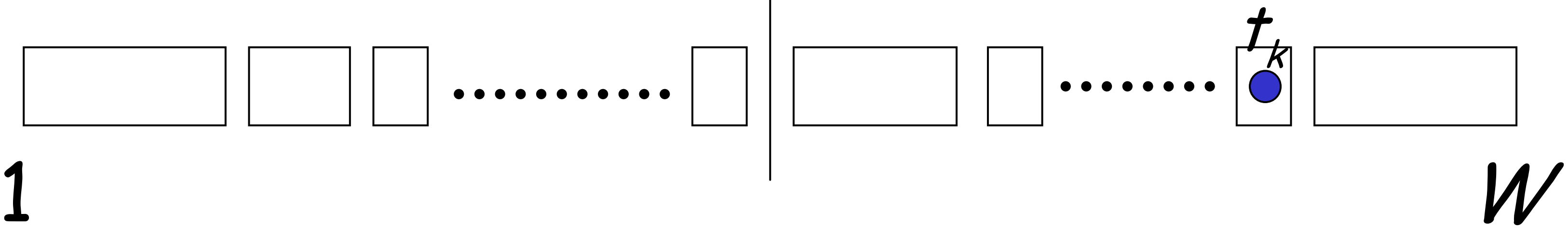


If: $\frac{t_l + t_r}{2} + 1 \leq t_k \leq t_r$ BAD!

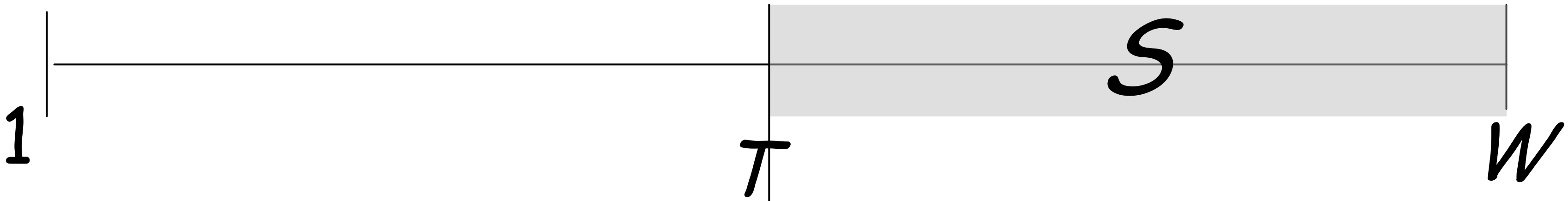
Element counted in wrong bucket



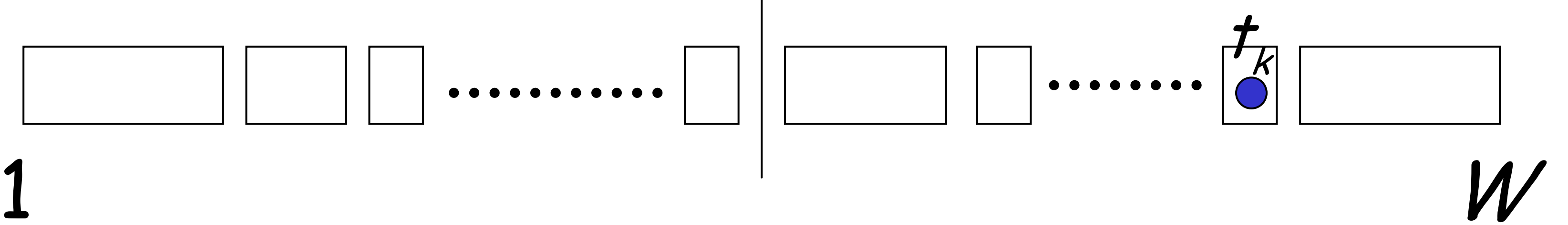
Consider Leaf Buckets



If $T \leq t_k \leq W$ GOOD!



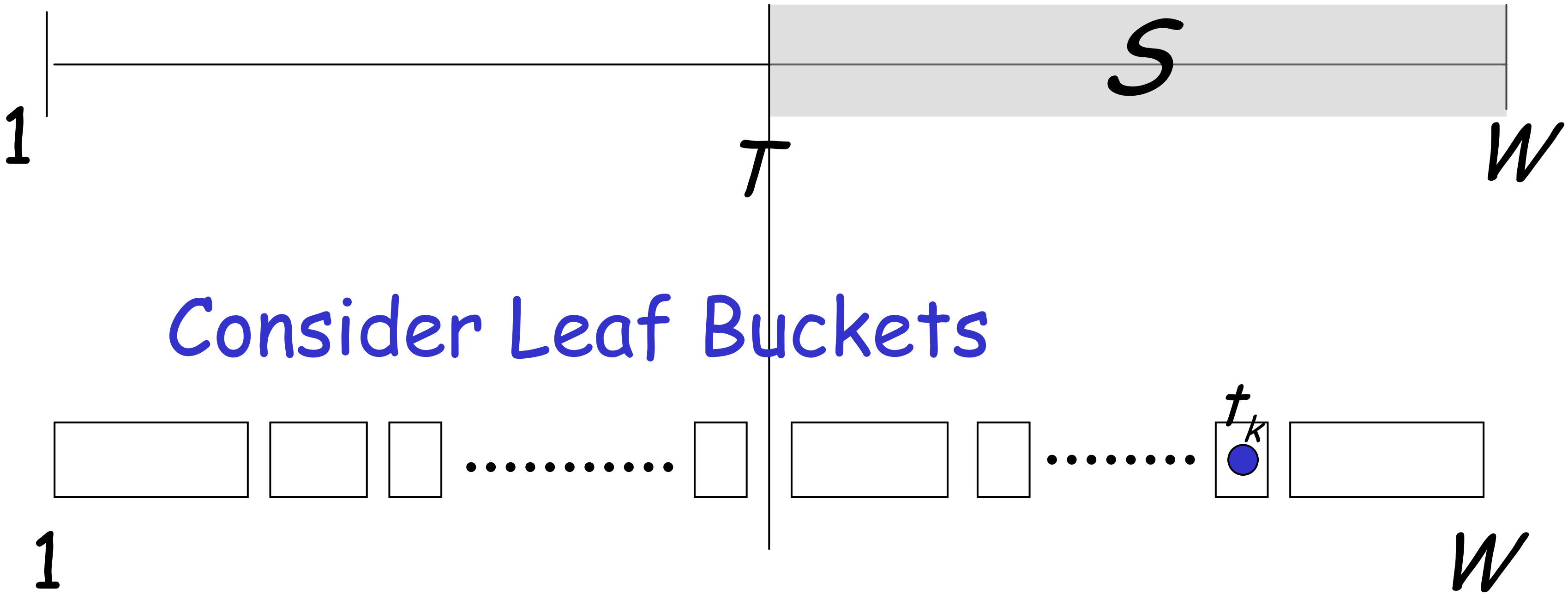
Consider Leaf Buckets



If $t_k < T$ BAD!

Element counted in wrong bucket

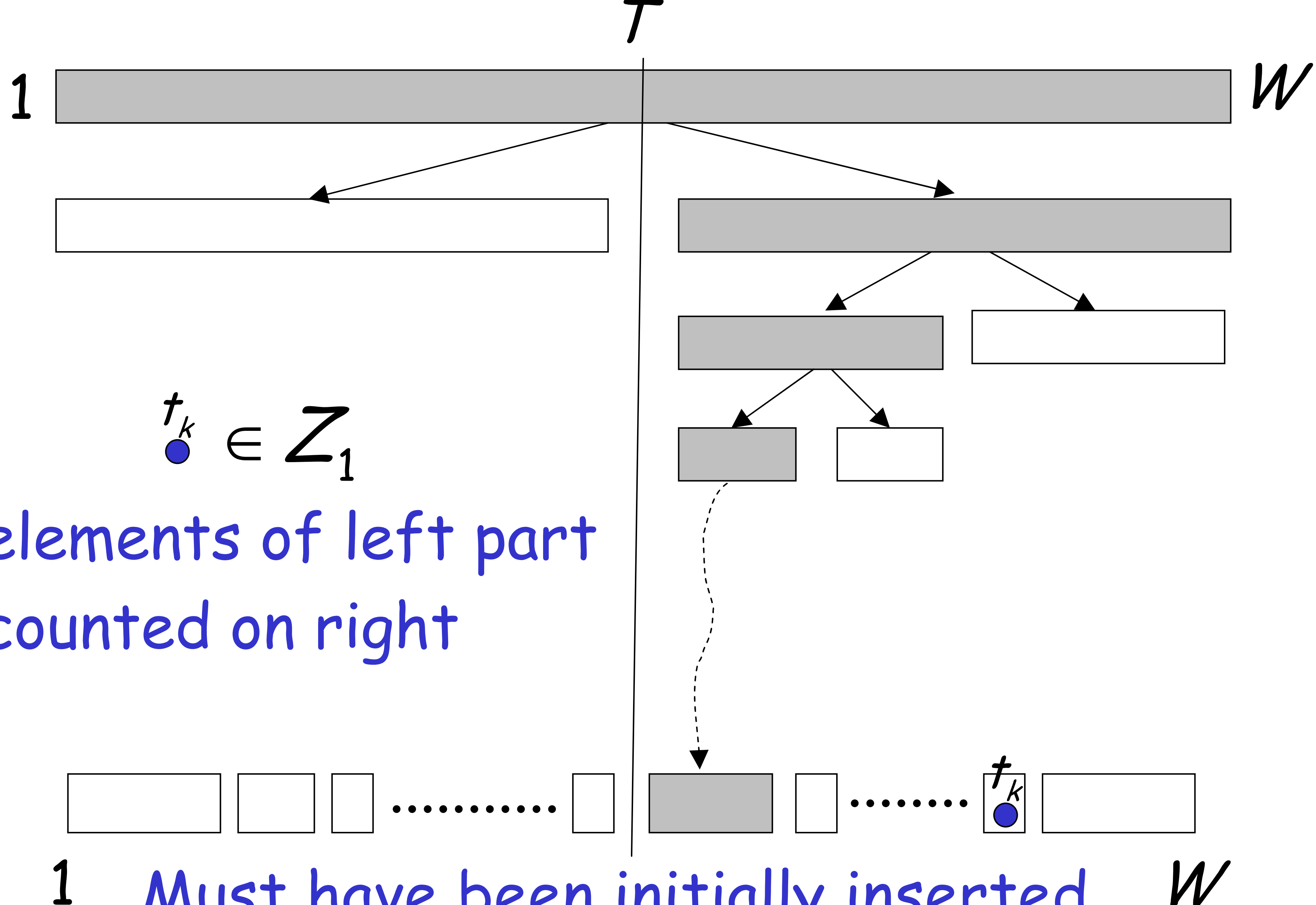
Consider Leaf Buckets



$$X = S - |Z_1| + |Z_2|$$

Z_1 :elements of left part counted on right

Z_2 :elements of right part counted on left



elements of left part counted on right

Must have been initially inserted in one of these buckets

Since tree depth $\leq \log W$

$$|Z_1| = O(2^i \log W)$$

Since tree depth $\leq \log W$

$$|Z_1| = O(2^i \log W)$$

Similarly, we can prove

$$|Z_2| = O(2^i \log W)$$

Therefore: $|X - S| \leq ||Z_1| - |Z_2|| = O(2^i \log W)$

Since $a = \frac{2+\varepsilon}{\varepsilon} \log W$

It can be proven $\varepsilon \cdot S = \Omega(2^i \log W)$

Since $a = \frac{2+\varepsilon}{\varepsilon} \log W$

It can be proven $\varepsilon \cdot S = \Omega(2^i \log W)$

Combined with $|X - S| = O(2^i \log W)$

We obtain relative error : $\frac{|X - S|}{S} \leq \varepsilon$