DASH for 3D Networked Virtual Environment

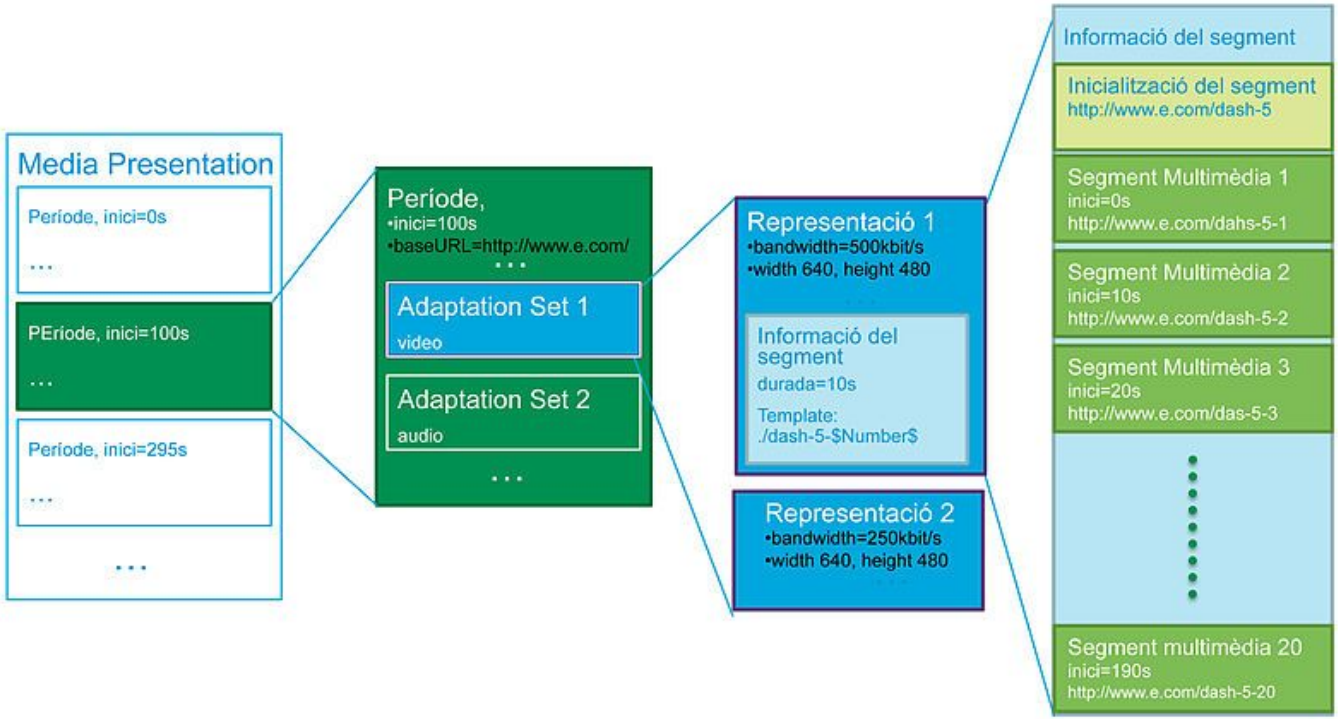
Forgione, T., Carlier, A., Morin, G., Ooi, W. T., Charvillat, V., & Yadav, P. K. (2018, October)
In *2018 ACM Multimedia Conference (MM'18), October*.

Introduction

- DASH is a widely deployed standard for streaming video content

- Can DASH be used for adaptive streaming of 3D content for Networked virtual environment(NEV)?

General MPD File

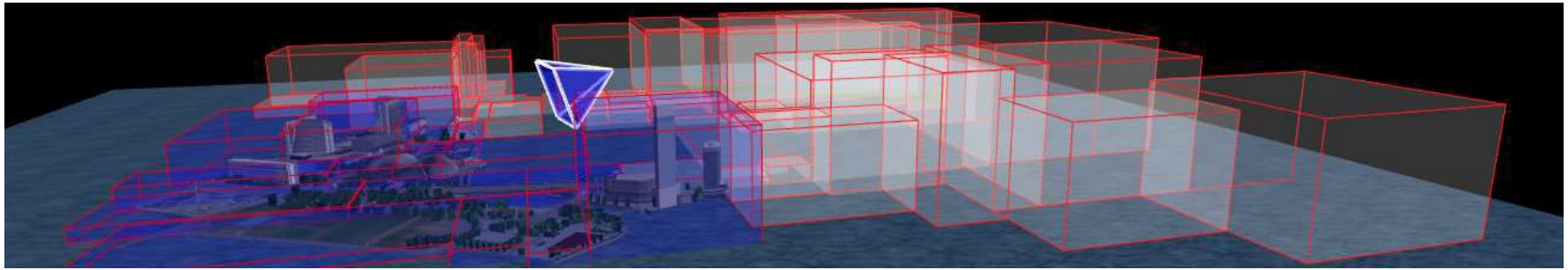


Three Chanllenge

- What are the metadata needs to provide along with 3D data ?
- How to organize the soup of 3D data into DASH adaptation sets ?
- How to help clients choose what to download, and at which resolution?

Mapping NVE into DASH

- Geometry adaptation sets
  - Use a space partitioning tree to organize the faces into cells
  - Create a separate adaptation set for large faces(e.g., the sky or ground)
  - Geometry information is spread on adaptation sets based on spatial coherence, allowing the client to download the relevant faces selectively



Mapping NVE into DASH

- Segments
  - Group the faces into segments to allow random access to the content in a geometry adaptation set
  - First sorting the faces in an adaptation set by their area in descending order
  - Then place each successive N faces into a segment

```
<AdaptationSet>
    <SupplementalProperty value="-8834.11230,2201.58853,
        -0.16950, 174.81540,-1344.47740,4767.83367" />
 <BaseURL>as1/</BaseURL>
 <Representation>
  <BaseURL>repr1/</BaseURL>
  <SegmentList>
   <SegmentURL area="2540342.3" size="120K" media="s0.obj" />
   <SegmentURL area="1124.4" size="162K" media="s1.obj" />
   <SegmentURL area="412.6" size="173K" media="s2.obj" />
   <SegmentURL area="270.3" size="147K" media="s3.obj" />
  </SegmentList>
 </Representation>
</AdaptationSet>
```

Mapping NVE into DASH

- Texture adaptation sets
  - Handle textures using adaptation sets but separate from geometry
  - Each texture file is contained in a different adaptation set with multiple resolutions
  - Add an attribute describing the average color of the texture
  - Use this attribute to render a face for which the corresponding texture has not been loaded yet

```
<AdaptationSet area="198632.73912" average="178,176,173"
      mimeType="image/png">
    <BaseURL>textures/MFLOOR07.PNG/</BaseURL>
    <Representation>
        <BaseURL>64x64/</BaseURL>
        <SegmentList>
            <SegmentURL size="7K" mse="57.6" media="t.png" />
        </SegmentList>
    </Representation>
    <Representation>
        <BaseURL>128x128/</BaseURL>
        <SegmentList>
            <SegmentURL size="27K" mse="0.0" media="t.png" />
        </SegmentList>
    </Representation>
</AdaptationSet>
```

Mapping NVE into DASH

Mapping NVE into DASH

- Material
  - A text file that describes all materials used in entire 3D model
  - Such as specular parameters and a path of each face to a texture file

Dash 3D Client

- A DASH-based NVE client need to estimate the usefulness of a given segment to download
- ***Utility*** is a function of a segment and dynamically computed online by the client from parameters in the MPD file

Segment Utility

- Utility for geometry segments

$$\mathcal{U}\left(s^G, v(t_i)\right) = \frac{\mathcal{A}_{3D}(s^G)}{\mathcal{D}(v(t_i), AS^G)^2}$$

- Utility for texture segments

$$\mathcal{U}\left(s^T, v(t_i)\right) = psnr(s^T) \sum_{k \in K} \frac{\mathcal{A}_{3D}(s_k^G \cap \Delta(T, t_i))}{\mathcal{A}_{3D}(s_k^G)} \mathcal{U}\left(s_k^G, v(t_i)\right)$$

DASH Adaptation Logic

- Due to transmission delay, the segment selected from viewpoint $v(t_i)$ will be delivered at time $t_{i+1} = t_{i+1(s)}$ depending on the segment size and network conditions:

$$t_{i+1}(s) = t_i + \frac{\texttt{size}(s)}{\widehat{BW_i}} + \widehat{\tau_i}$$

- A better solution is to download a segment that is expected to be the most useful in the future. With a temporal horizon χ, we can optimize the cumulated utility over $[t_{i+1(s)}, t_i + \chi]$:

$$s_i^* = \underset{s \in \mathcal{S} \setminus \mathcal{B}_i \cap \mathcal{FC}}{\operatorname{argmax}} \int_{t_{i+1}(s)}^{t_i+\chi} \mathcal{U}(s, \hat{v}(t_i))dt$$

DASH Adaptation Logic

- Alternative greedy heuristic selecting the segment that optimizes an utility variation during downloading (between t_i and t_{i+1}):

$$s_i^{\text{GREEDY}} = \underset{s \in \mathcal{S} \setminus \mathcal{B}_i \cap \mathcal{FC}}{\operatorname{argmax}} \frac{\mathcal{U}\left(s, \hat{v}(t_{i+1}(s))\right)}{t_{i+1}(s) - t_i}$$

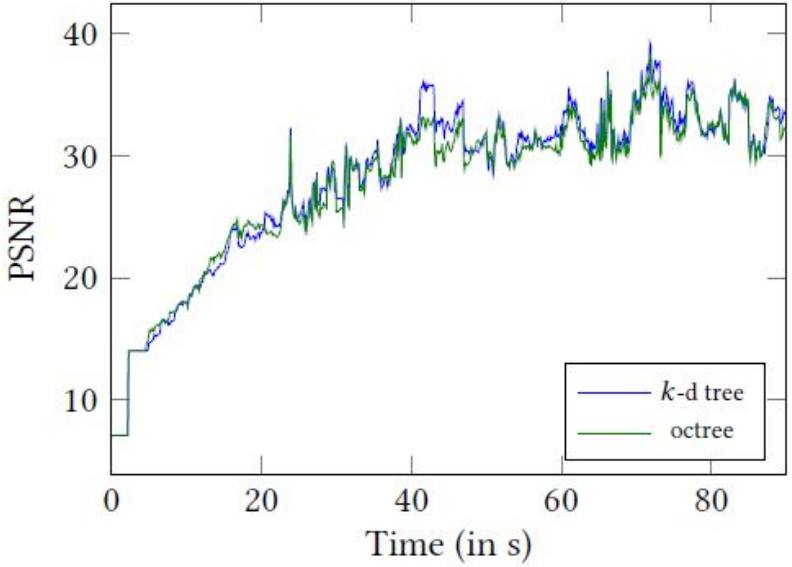
Evaluation

- Prepared a model which has 387,551 vertices and 552,118 faces, and partition the geometry into a k-d tree until the leafs have less than 10000 faces
- Collected six realistic users navigation traces that can be replayed in experiments

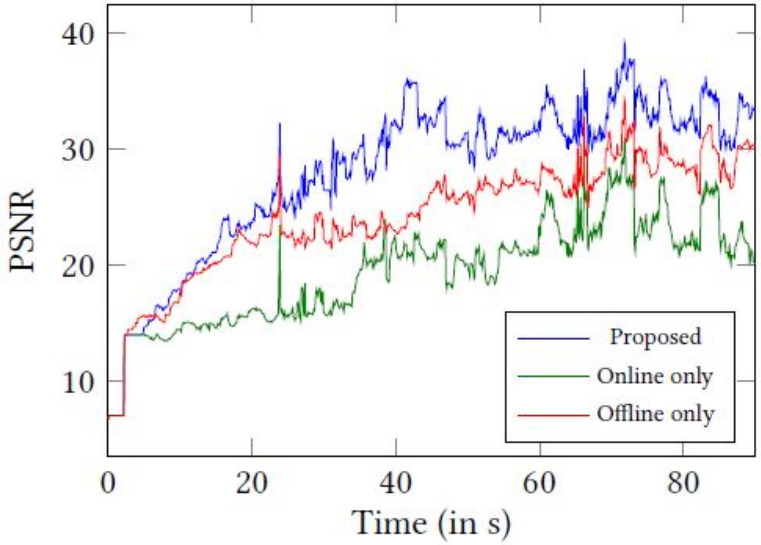
Parameters	Values
Content preparation	Octree, *k*-d tree
Utility	Offline, Online, Proposed
Streaming policy	Greedy, Proposed
Grouping of Segments	Sorted based on area, Unsorted
Bandwidth	2.5 Mbps, 5 Mbps, 10 Mbps

Table 2: Different parameters in our experiments

Evaluation

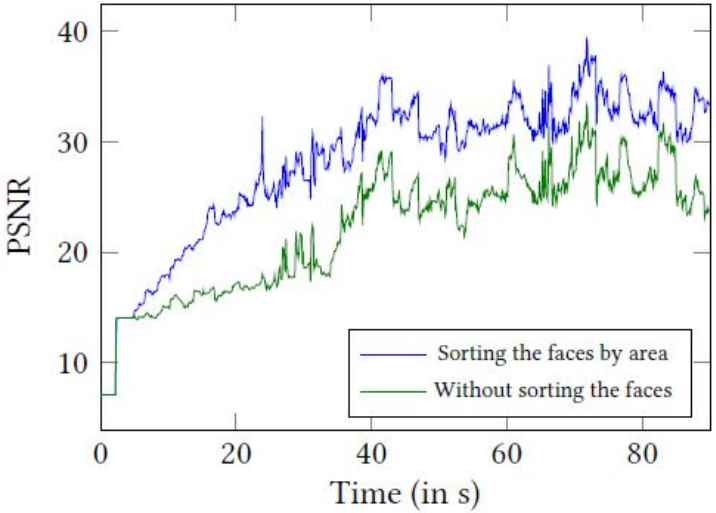


Impact of the space-partitioning tree on the rendering quality with a 5Mbps bandwidth.

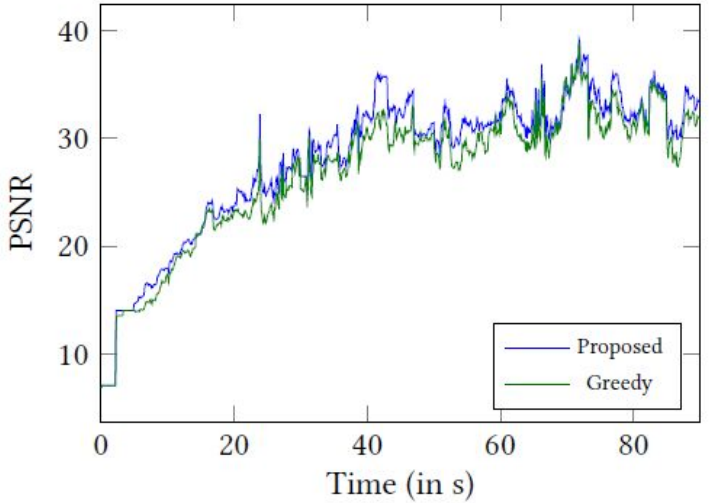


Impact of the segment utility metric on the rendering quality with a 5Mbps bandwidth.
Offline:$A_{3D}(s^G)$, Online:$1/D(v(ti),A_{SG})^2$,
Proposed:$(A_{3D}(s^G)/D(v(ti),A_{SG})^2)$

Evaluation



Impact of creating the segments of an adaptation set based on decreasing 3D area of faces with a 5 Mbps bandwidth.



Impact of the streaming policy (greedy vs. proposed) with a 5 Mbps bandwidth.

Evaluation

Resolutions	2.5 Mbps	5 Mbps	10 Mbps
1	5.7% vs 1.4%	6.3% vs 1.4%	6.17% vs 1.4%
2	10.9% vs 8.6%	13.3% vs 7.8%	14.0% vs 8.3%
3	15.3% vs 28.6%	20.1% vs 24.3%	20.9% vs 22.5%
4	14.6% vs 18.4%	14.4% vs 25.2%	14.2% vs 24.1%
5	11.4% vs 0.3%	11.1% vs 5.9%	11.5% vs 13.9%

Percentages of downloaded bytes for textures from each resolution, for the greedy streaming policy (left) and for our proposed scheme (right).

Limitation

- Did not take account the geometry resolution
- The relationship between geometry and texture is arbitrary
- It is only for mesh model

Conclusion

- DASH can be used for NVE
- The metadata being precomputed offline is sufficient for client to make intelligent decisions about what to download
- The balance between geometry and texture is a viable way to increase visual quality