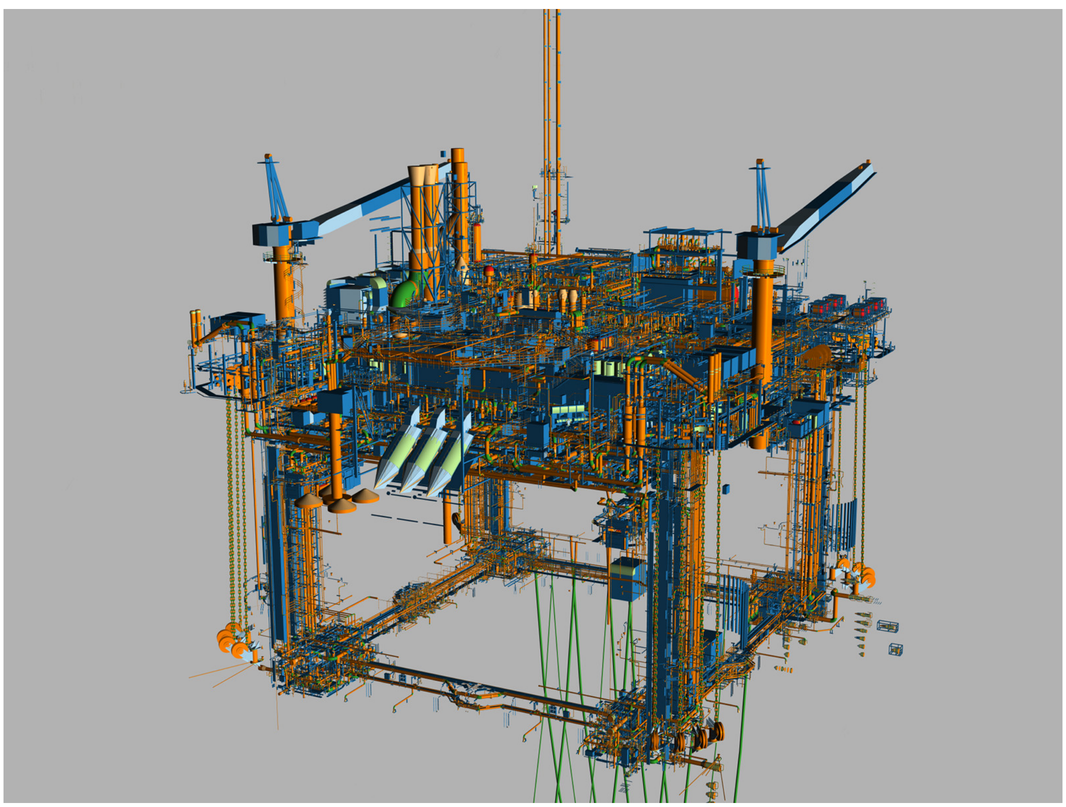
NORSIGD INFO

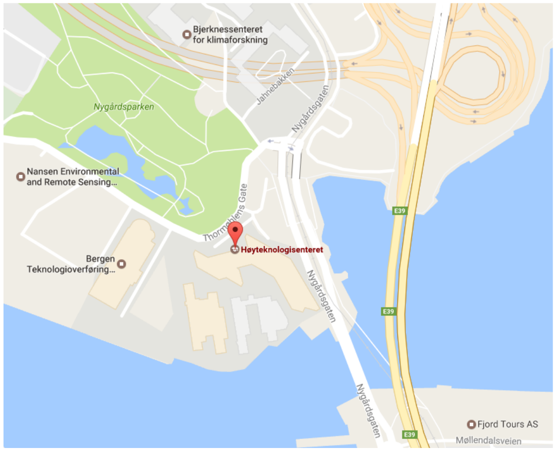
Nummer 1 2017



NORSK SAMARBEID INNEN GRAFISK DATABEHANDLING

ISSN 0803-8317

Notiser



UiB fortsetter med VCF i 2017

v/ Andreas Johnsen Lind, Universitetet i Bergen

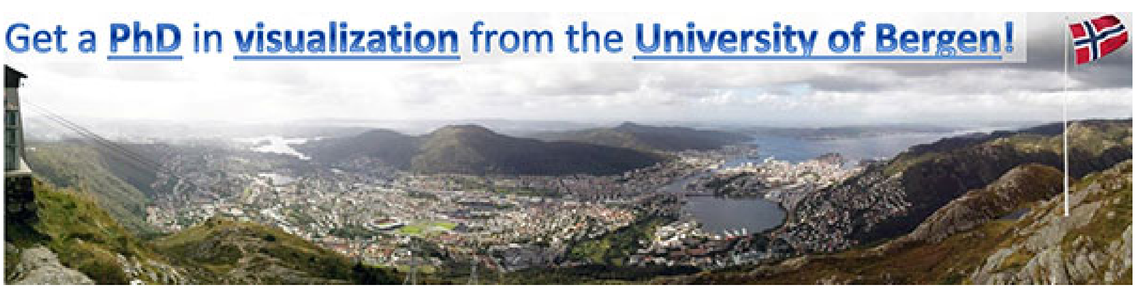
Visualiseringsgruppen ved Universitetet i Bergen (UiB) vil fortsette med sine månedlige foredrag med inviterte foredragsholdere innen visualisering, bildeprosessering, datagrafikk og andre visualiseringsrelaterte emner også i 2017. Disse foredragene som vi har valgt å kalle Visual Computing Forum (VCF), har nå vært et fast innslag hos oss i over fem år. Vi ser fram til mange nye interessante og læringsrike bidrag, ikke bare fra bidragsytere lokalt i Bergen, men også fra norske og internasjonale foredragsholdere. Foredragene har ikke en fast dato hver måned men gjennomføres alltid på fredager klokken 10:15 ved Høyteknologisenteret i Bergen en gang i måneden. Dersom du ønsker å bli informert om kommende VCFer kan du ta kontakt på `vcf.bergen@gmail.com` eller holde deg oppdatert med VCFs nettside `http://www.ii.uib.no/vis/vcf/`.

Ledige Stillinger ved Visualiseringsgruppen på UiB

v/ Andreas Johnsen Lind, Universitetet i Bergen

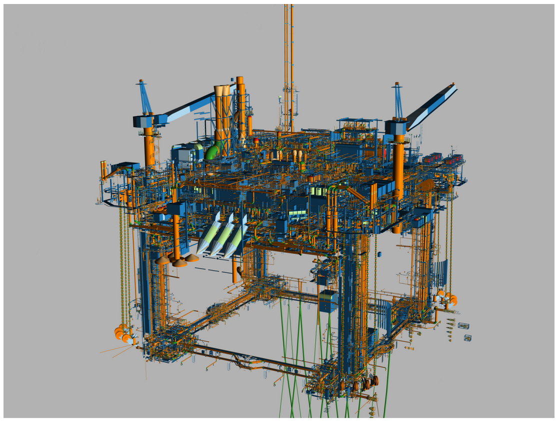
Visualiseringsgruppen ved Universitetet i Bergen (UiB) har flere ledige stillinger i 2017. Institutt for Informatikk ved Universitetet i Bergen har 2 stipendiat-stillinger og 1 post-doc stilling som skal fylles. Frist for å søke på disse stillingene er 1. mars 2017. For mer informasjon, se følgende nettsted: `http://www.ii.uib.no/vis/about/jobs.html` eller ta kontakt med prof. Helwig Hauser (`Helwig.Hauser@uib.no`) eller prof. Stefan Bruckner (`Stefan.Bruckner@uib.no`).

Dessuten er en treårig stilling i *MetaVis*-prosjektet ledig. Prosjektet utforsker visualeringsløsninger med det målet at brukere effektivt skal kunne velge riktig visualiseringsløsning for sine problemstillinger. For mer informasjon, se det tidligere nevnte nettstedet eller ta kontakt med prof. Bruckner. Instituttet ønsker også å ansette inntil to professorer eller førsteamanuenser med ekspertise innen maskinlæring. Dette er ment som det første steget i en prosess for å bygge en sterk maskinlæringsgruppe ved instituttet. Igjen, se det tidligere refererte nettstedet for mer informasjon.



VRVis Conference Calendar: `http://confcal.vrvis.at/`

Eurographics Events: `http://www.eg.org/events`



Om forsiden
Skjermbilde laget med OpenSceneGraph som viser Kristin-plattformen. Bilde hører til artikkelen ved Tore Hammervoll om “Efficient rendering of parametric CAD data using geometry instancing” fra side 4 i dette heftet.

Hilsen fra styret

Kjære medlemmer,

De teknologiske forandringene i vårt fagfelt skjer tilsynelatende så fort at vi ikke rekker å fortelle hverandre om dem. Jeg husker en tid da vi diskuterte heftig da nye bilder eller animasjoner fra de daværende stjernene på datagrafikkhimmelen ble vist. Vi diskuterte hvordan de kunne ha laget disse bildene og hvilke eventuelle “feil” de hadde, med lure-skygger og liksom-speil.

Hvordan er det idag når nyvinninger kommer? Jo, man nikker anerkjennende i noen sekunder i retning skjermen der nyvinningen blir vist. Så venner man seg til dette – som om det hadde vært slik hele tiden.

Man glemmer at bak alle de store fremskrittene ligger det hardt arbeid – små og stadige forbedringer. Denne utgaven av NORSIGD Info forsøker å motvirke dette. Vi presenterer to faglige artikler og en konferanserapport, samt notiser og stillingsannonser fra Universitetet i Bergen. Vi ønsker våre lesere god fornøyelse.

Selv om året 2017 allerede er godt i gang ønsker styret i NORSIGD dere alle et godt datagrafikkår!

Hilsen,

Wolfgang Leister

NORSIGD Info
– medlemsblad for NORSIGD

Utgitt av: NORSIGD
Ansvarlig: Wolfgang Leister
Norsk Regnesentral
Postboks 114 Blindern
0314 OSLO

ISSN: 0803-8317

Utgivelser: 2017: 1/17 2/17

Layout: Wolfgang Leister
$\LaTeX 2_\epsilon$

Innhold

Efficient rendering of parametric CAD data using geometry instancing

Tore Hammervoll (Bergen University College)

This paper presents the results of profiling geometry instancing using several different techniques for storing per-instance data, each of them tested with different combinations of number of vertices and number of instances. The profiling is done to identify what number of vertices and number of instances are optimal when using geometry instancing, and the performance is compared to the performance of batching. This paper will show that using geometry instancing when rendering massive parametric CAD models can lead to huge reduction in memory usage, while sacrificing very little in performance, and even improving the performance in certain cases.

High level graphics APIs like OpenGL and DirectX (version 11 and older) perform some processing using the CPU every time the application tells it to render something to the screen. Telling the graphics API to draw something, issuing a draw call, may be quite fast, but for real-time graphics applications there is a very limited amount of time available to perform such tasks. This sets a limitation on the total number of draw calls that can be issued per frame. Figure 1a shows a planetary ring with 3 000 identical rocks being rendered by issuing 3 000 draw calls, while Figure 1b shows how 150 000 identical rocks can be rendered by issuing just one instanced draw call. In both cases, 55 frames are rendered per second.

A common technique to reduce the number of draw calls is to merge several small meshes together into one bigger mesh that can be rendered in one draw call, a technique called batching [6]. When batching is used, multiple small meshes are positioned, rotated, and scaled (transformed) relative to each other, and then merged together to form one big mesh. This means that batching uses a large amount of memory. Even if an object is reused multiple times, like the rock model in Figure 1, each version needs its own copy of the mesh data so that it can be transformed and then merged to form a bigger batch.

Like batching, geometry instancing aims to reduce the number of draw calls, but does so by using an instanced draw call that takes an additional parameter specifying the number of instances of the mesh that should be rendered. For this to work, all data that is unique to each instance, for example transformation matrix or a material ID, has to be stored on the GPU, or generated by the GPU in some way. This data is needed by the GPU during rendering in order for each instance to appear different in some way when rendered to the screen. As only one version of each instanced mesh is needed in memory, and the per instance memory requirement is constant, the memory usage is far smaller than it is for batching. Additionally, an instance can be dynamic by changing just the instance data, which incurs very little overhead.

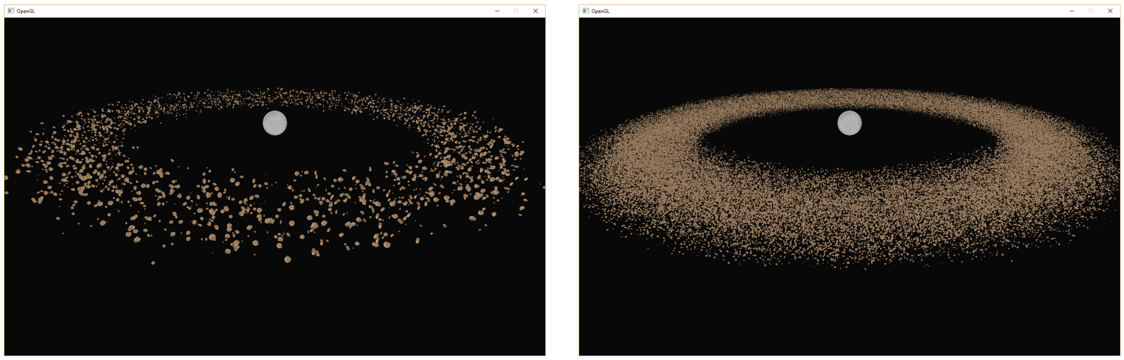
This paper will present the results of profiling different instance data storage techniques available with OpenGL 4.5, compared with using batching and one draw call per instance. The purpose is to find what effect the amount of vertices in the mesh, and the amount of instances of that mesh, has on the performance of the different techniques. The performance and memory usage of geometry instancing will also be compared to batching when rendering parametric CAD data, where a massive model is constructed using simple geometric primitives such boxes, cylinders, and toruses.

Related Work

A common use case for geometry instancing is for rendering the trees in a forest, or individual leaves of grass on a field. Deussen et al. [2] present a system for realistic rendering of plant ecosystems, where geometry instancing is utilized to reuse both individual parts of a plant and whole plants. Bao et al. [1] present a framework for rendering large scale forest scenes by rendering trees in multiple levels of detail using geometry instancing.

An algorithm for automatically creating instances from objects organized hierarchically in a scene graph proposed by Schultz and Schumann [5], work by identifying group nodes that are instances of another group node. Another paper by the same authors describe methods for efficiently storing a scene using instancing concepts [4].

Geometry instancing has also been shown to be a promising technique for rendering massive CAD models. Santos and Filho [3] present an efficient approach to reduce the memory usage without sacrificing too much rendering performance by using geometry instancing. The instances in the scene are identified using a shape matching algorithm, and with a



(a) Without instancing: 3000 rocks rendered at 55 fps   (b) With instancing: 150000 rocks rendered at 55 fps

Figure 1: Planet with a ring rendered with and without geometry instancing

moderate error threshold the reductions in memory needed to represent the scene is considerable.

Profiling

The profiling has been performed by rendering a number of instances of a mesh with a number of vertices, such that the total number of vertices processed each frame is constant. For each combination of amount of vertices and amount of instances, 1000 frames have been rendered and the mean frame time calculated. This has been done with five different instance data storage techniques, instanced vertex attribute, texture, buffer texture, shader storage buffer object (SSBO), and uniform buffer object (UBO). In addition, this profiling has been performed using one draw call per instance, where the instance data, a transformation matrix, is set using a uniform variable between each draw call, and using batching, where the mesh is copied and transformed using the transformation matrix for each instance, and then merged together to form one big batch.

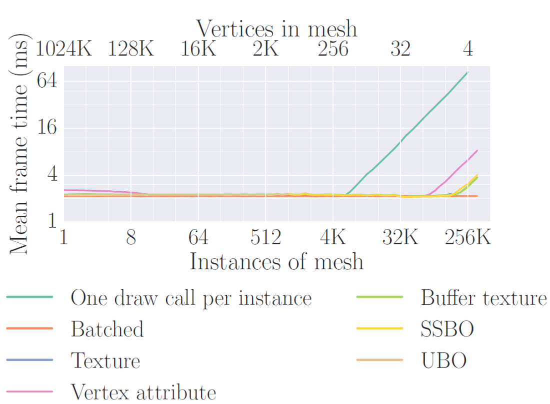


Figure 2: Mean frame time of combinations of vertex amount and instance amount. 1024K vertices rendered in total.

Figure 2 shows the mean frame time of profiling with static instance data. As can be seen, there is almost no change in frame time between the different instance data storage techniques and batching, and between the different combinations of amount of vertices and amount of instances.

Using one draw call per instance the frame time is far higher than it is for the other techniques, if the amount of instances is high. This is to be expected, as this technique is limited by the overhead of issuing such a large number of draw calls per frame. For the geometry instancing techniques there is also an increase in frame time with certain combinations, but this occurs only when there are very few vertices in the mesh.

		Frame time	
Technique	Vertices	Static	Dynamic
One draw call per instance	16384	2.15 ms	2.14 ms
	1024	2.21 ms	2.18 ms
	64	5.49 ms	5.38 ms
Batching	16384	2.16 ms	7.98 ms
	1024	2.16 ms	8.19 ms
	64	2.14 ms	8.02 ms
Geometry instancing	16384	2.24 ms	2.32 ms
	1024	2.30 ms	2.36 ms
	64	2.23 ms	2.27 ms

Table 1: *Mean frame time of 16K, 1024, and 64 vertices in the mesh, with different percentages of instances being dynamic. 1024K vertices rendered in total.*

Even if the instance data, a transformation matrix, is updated every frame for some or all of the instances, there is almost no difference in mean frame time. Table 1 compares the frame time with all of the instance data being static and dynamic, for the combinations of 16K vertices and 64 instances, 1024 vertices and 1024 instances, and 64 vertices and 16K instances.

If using batching, there is a huge difference in frame time between having static or dynamic instance data. This is simply due to the amount of time it takes to update the vertex data of the batch. With all of the instances being dynamic, every vertex in the batch has to be updated, and in this case there are 1024K vertices in total.

Using geometry instancing however, there is barely any difference in frame time between static and dynamic instance data.

Parametric CAD data

A parametric CAD model of the Statoil Kristin offshore platform has been rendered using the batching technique and using geometry instancing. Figure 3 shows a screen capture where each parametric primitive is colored based on their type. Cylinders are orange, boxes are dark blue, pyramids are light blue, etc.

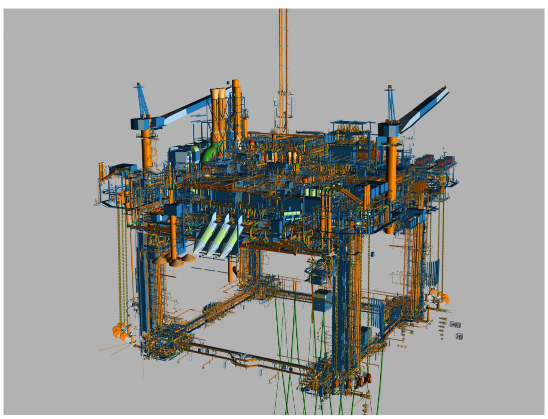


Figure 3: Screen capture of the Kristin platform rendered using OpenSceneGraph. Only simple primitives have been loaded.

The parametric CAD model is loaded in such a way that as much information about a primitive as possible is used to create a transformation matrix, while the mesh is as general as possible. Cylinders for example, have a height and radius parameter which is used to create a scaling matrix to scale a cylinder mesh with height 1 and radius 1. While all cylinders can use the same mesh, this is not the case for every type of primitive. By reusing meshes as much as possible however, this CAD model with 1.1 million primitives could be reduced to just 5789 unique meshes.

The memory usage and frame time of rendering this CAD model is shown in Table 2. Two versions of the CAD model has been rendered with different levels of detail (LOD). With LOD 10, cylinders and other primitives that are circular are constructed using ten segments to approximate a circle. This results in a total of 45 million vertices. With LOD 15, there are a total of 68 million vertices.

Comparing the frame time using batching and instancing shows that the frame time of batching is somewhat lower with the LOD 10 version, while it is far higher with the LOD 15 version. The reason for this is that the GPU used in this case has 2 GB of dedicated memory, which is less than is needed when using batching. This results in some of the data being located on system memory instead of the dedicated GPU memory.

Technique	LOD	Memory	Frame time
Batching	10	1600 MB	47.4 ms
Instancing	10	186 MB	60.6 ms
Batching	15	2362 MB	161.3 ms
Instancing	15	196 MB	90.1 ms

Table 2: *Memory usage and performance of rendering a parametric CAD model using batching and instancing at different levels of detail.*

Conclusion

The profiling shows that all of the instance data storage techniques, as well as batching, perform equally across all combinations of number of vertices and number of instances. Geometry instancing also performs just as well with dynamic instance data as it does with static instance data.

Using geometry instancing when rendering parametric CAD data does results in somewhat higher frame time compared to batching if there is enough dedicated GPU memory. In memory constrained situations however, the huge reduction in memory usage when using geometry instancing leads to far lower frame time as well.

References

[1] G. Bao, H. Li, X. Zhang, and W. Dong. Large-scale forest rendering: Real-time, realistic, and progressive. *Computers & Graphics*, 36(3):140 – 151, 2012. ISSN 0097-8493. doi: 10.1016/j.cag.2012.01.005. Novel Applications of {VR}.

[2] O. Deussen, P. Hanrahan, B. Lintermann, R. Měch, M. Pharr, and P. Prusinkiewicz. Realistic modeling and rendering of plant ecosystems. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, pages 275–286, New York, NY, USA, 1998. ACM. ISBN 0-89791-999-8. doi: 10.1145/280814.280898.

[3] P. I. N. Santos and W. C. Filho. Instanced rendering of massive cad models using shape matching. In *2014 27th SIBGRAPI Conference on Graphics, Patterns and Images*, pages 335–342, Aug 2014. doi: 10.1109/SIBGRAPI.2014.34.

[4] R. Schultz and H. Schumann. Efficient scene descriptions using advanced modelling techniques in the renderman context. In *Proceedings of Spring Conference on Computer Graphics*, 2000.

[5] R. Schultz and H. Schumann. Automatic instancing of hierarchically organized objects. In *Proceedings of the 17th Spring Conference on Computer Graphics*, SCCG '01, pages 63–, Washington, DC, USA, 2001. IEEE Computer Society. ISBN 0-7695-1215-1.

[6] M. Wloka. Batch, batch, batch: What does it really mean? Presentation at game developers conference, 2003.

Acknowledgement. This paper is based on the authors master's thesis titled "Efficient rendering of parametric CAD data using geometry instancing". The subject was suggested by GexCon AS and submitted as part of the joint master's program in software engineering at University of Bergen and Bergen University College. Thanks to Harald Soleim and Atle Geitung for their contributions regarding this paper and the thesis it is based on.

Konferanserapport IEEE VIS 2016

Andreas Johnsen Lind, Universitetet i Bergen

Den årlige IEEE VIS-konferansen om visualisering arrangeres av IEEE Computer Society's Visualization and Graphics Technical Committee (VGTC). Siden den første VIS-konferansen ble holdt i San Francisco i 1990 har den blitt gjennomført årlig i forskjellige byer i USA. Utenom i 2014 da den ble gjennomført i Paris, Frankrike. VIS-konferansen i 2016 ble avholdt i Baltimore, Maryland og chaired av Terry Yoo som til daglig jobber ved National Institutes of Health.

IEEE VIS regnes som visualiseringsfeltets viktigste forum for utveksling av ideer og forskningsresultater. Den ukeslange konferansen samler ledende eksperter i feltet fra hele verden og inkluderer innslag fra akademia, bedrifter, kunstverden og andre med en brennende interesse for visualisering. Fra og med 2016 vil konferansen også ha ett større fokus på sin rolle som et rekrutteringsforum for bedrifter og akademiske organisasjoner med en interesse for søkere med visualiseringskompetanse. 32 bedrifter og institusjoner hadde en tilstedeværelse ved VIS 2016 enten som sponsorer eller for å presentere seg som arbeidsplass/sine produkter.

Artikkelen "Comparative Visualization of Protein Secondary Structures" mottar prisen for beste artikkel i BioVis.

VIS 2016 ble avholdt i Baltimore, Maryland fra søndag 23. oktober t.o.m. fredag 28. oktober. Selve konferansen ble holdt på Hilton Baltimore Hotel som ligger i den såkalte inner harbor delen av byen. VIS er inndelt i 3 hovedkonferanser Visual Analytics Science and Technology (VAST), Information Visualization (InfoVis) og Scientific Visualization (SciVis). I tillegg ble årets IEEE Symposium om Large-Scale Data Analysis (LDAV) og IEEE Symposiet om Visualization for Cyber Security avholdt som del av VIS-uken. Videre ble 12 workshops (inkluderte BioVis, VDS og BELIV), 10 tutorialer, 6 paneler gjennomført i løpet av VIS-uken. En rekke andre arrangementer og presentasjoner inkludert en kunstutstilling ble også gjennomført som del av konferansen.

Visualiseringsgruppen fra Universitetet i Bergen (UiB), Bergen, Norge hadde det høyeste antallet konferanseartikler fra en enkelt institusjon i årets konferanse. Hele 7 artikler hadde en UiB-affiliert forfatter på forfatterlisten,

- A Fractional Cartesian Composition Model for Semi-spatial Comparative Visualization Design, DOI: 10.1109/TVCG.2016.2598870.
- Designing Progressive and Interactive Analytics Processes for High-Dimensional Data Analysis, DOI: 10.1109/TVCG.2016.2598470.
- Vol2velle: Printable Interactive Volume Visu-

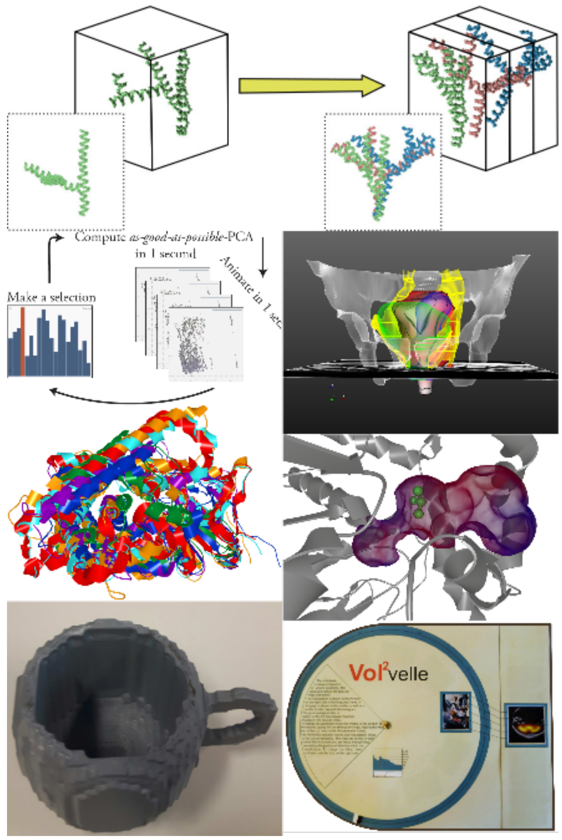
alization, DOI: 10.1109/TVCG.2016.2599211

- PelVis: Atlas-based Surgical Planning for Oncological Pelvic Surgery, DOI: 10.1109/TVCG.2016.2598826.
- Comparing Cross-Sections and 3D Renderings for Surface Matching Tasks using Physical Ground Truths, DOI: 10.1109/TVCG.2016.2598602.
- Comparative Visualization of Protein Secondary Structures **[Beste artikkel BioVis]**.
- Interactive Exploration of Ligand Transportation through Protein Tunnels.

Dette inkluderte Comparative Visualization of Protein Secondary Structures som vant beste artikkel-prisen i BioVis 2016 og hvor tre av de fem forfatterne var affiliert med Universitetet i Bergen.

Det var flere innviterte foredragsholdere til VIS uken i 2016. Blandt dem var professor Ricardo Hausmanns, direktør hos Center for International Development. Han ga et keynote-foredrag om sammenhengen mellom teknologisk spredning og nasjoners velstand. Tesen til Hausmann er at hemmeligheten til økonomisk velstand for en nasjon er den samme som i et slag Scrabble. Desto flere bokstaver, desto flere ord kan man lage. Det samme gjelder for stater, hvor desto flere kapabiliteter, spesielt teknologiske, desto flere produkter kan du lage og desto rikere blir nasjonen. Dette er en interesant ide og professor Ricardo argumenterte godt for den. Capstone-foredraget til konferansen ble gitt av Jean-Luc Doumont, PhD fra kommunikasjonsselskapet Principæ. Foredraget hadde tittelen "Kommunikasjonens Tre Lover". De tre lovene han omtalte var (*1*) Tilpass deg til publikum, (*2*) Maksimer signal- vs. støy-raten og (*3*) Effektiv bruk av redundans for effektiv kommunikasjon. Utover disse tre grunnreglene ga han også flere grunnleggende råd for hvordan å gi gode vitenskapelige presentasjoner.

VIS-uken har alltid vært ett must for personer med en interesse for visualisering, med et akademisk program uten sidestykke i visualiseringsverdenen og gode muligheter for å bygge nettverk. Og den nyligste gjennomførelsen kan ikke sies å ha vært ett unntak. Med ett fantastisk akademisk program, de sedvanlige sosiale arrangementene i regi av diverse forskningsgrupper og ett utvidet fokus på rekruttering så har VIS 2016 hatt ett bredt program med noe for alle som har en interesse for visualisering. For meg personlig var VIS 2016 en stimulerende opplevelse med mulighet for å reflektere over mitt eget arbeid. VIS 2017 går av stabelen 1. Oktober, 2017 i Phoenix, Arizona og abstract deadlinen er satt til Tirsdag, 21. Mars (se http://ieeevis.org/ for mer informasjon). VIS er en konferanse som absolutt anbefales til alle som ønsker å holde seg oppdatert på trendene i visualisering og jeg ser fram til å møte mange av dere i Phoenix i Oktober.



Utvalgte bilder fra UiBs bidrag til VIS 2016.

MERK: I 2018 kommer IEEE VIS tilbake til Europa. Da blir den avholdt i Berlin, Tyskland, så for de som synes at en tur til USA blir for langt kommer det en mulighet til å besøke VIS med en langt kortere flytur enn i år.

The Graphical Expression of the Visitor Engagement Installation (VEI) Profile Revisited

Wolfgang Leister, Peder Aursand, Ingvar Tjøstheim (Norsk Regnesentral)
Syver Lauritzsen, Göran Joryd (Expology)

In a previous issue of NORSIGD Info, we have presented the VEI profile that characterises installations in science centres and museums for the evaluation of the how engaging these exhibits are for visitors. The VEI profile is used by the stakeholders (such as curators and exhibition designers) to specify and evaluate characteristics of exhibits. In its original form, the VEI profile is represented as a spiderweb diagram. While using this graphical representation in practice, we noticed some disadvantages that we try to avoid using a new representation. In the following, we present the principles behind this new graphical representation and how it is used.

Science centres and museums present exhibitions, installations, and educational programmes that are designed to engage visitors for self-education on a subject and to inspire the visitors to learn more. The VEI profile [2] was developed as a structured way of quantifying the characteristics in installations along eight dimensions each of which is given a value between 0 and 5 according to a definition table. The dimensions of the current[1] VEI profile represent the degrees of *competition* (C), *narrative* elements (N), *interaction* (I), *physical* activity (P), *visitor (user) control* (U), *social* aspects (S), *achievements* awareness (A). and *exploration* possibilities (E).

Note that the values in this definition table cannot be associated with good or bad design of an installation, but these show properties of the installation. For instance, some installations will include storytelling as a main ingredient while others will present a fact that the visitors can experiment with (i.e., use a much simpler narrative). For some of the eight dimensions, high or low values might be appealing to some members of the target group while others might find these less attractive. Thus, the values of the VEI profile for an installation should be aligned to the visitor type.

The VEI profile Spiderweb Diagram

The VEI profile can be used to show the characteristics of installations as well as ranges of characteristics that are used when curators specify requirements to the designers of these installations. In the assessment of the VEI profile of an installation, one goes through all eight dimensions and sets the values according to a table that is given in the paper by Leister et al. [2].

The spiderweb-diagram, as shown in Figure 4, has been chosen as representation. Early experiments with the spiderweb diagram showed that the value 0 in some dimensions could cause degenerated diagrams that were difficult to interpret. Thus, we decided to move the value 0 out of the midpoint of the diagram.

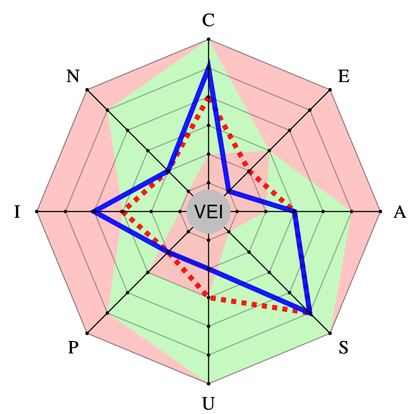


Figure 4: Spiderweb representation of a VEI profile. The green area shows the minimum and maximum values from a specification, the blue line the assessment by the designer, while the red dotted polygon shows the opinion of an evaluation panel.

Further, the slopes of a polygon in the spiderweb diagram show a dependency between two adjacent dimensions that does not have a particular meaning. On the positive side, the area a polygon covers can give an overall impression of the expectations to an installation.

Regarding the visualisation of ranges, when a range only consists of one value, the green area (see the E-dimension in the example of Figure 4) does not have a width; thus, it can happen that the diagrams degenerate, and the green area is not visible if for several consecutive values the upper and lower number for one dimension are equal.

[1] An earlier version of the VEI profile based on six dimensions has been presented in a previous issue of NORSIGD Info [1].

The VEI profile Design Diagram

The new VEI profile Design Diagram shows the dimensions in eight segments of a disc, as shown in Figure 5. In this representation, ranges of values (these are needed for specification purposes) are shown as coloured segment slices. Single values that are assessed using the definition table are shown as hatched segment slices. Measured values or values from an opinion assessment are shown as orange arcs. This arc can be positioned stepless; arcs in the middle of a segment slice represent the value of this segment slice.

This representation avoids the disadvantages of the Spiderweb representation: the slopes between values and possible degeneration of the diagram are avoided. Further, choosing a coloured area, hatches, and a bold line segment as graphical elements, the representation makes a clean impression. Note that we did not use the mid-point of the disk for the level 0 since a triangle shape in the middle would have been a new element in the visualisation.

Both diagram types have as a weakness that the areas covered do not reflect useful information. For example, in a specification of one dimension from 0 to 2 would be visualised with a smaller area than one ranging from 3 to 5. Currently, we have not seen this as a serious issue.

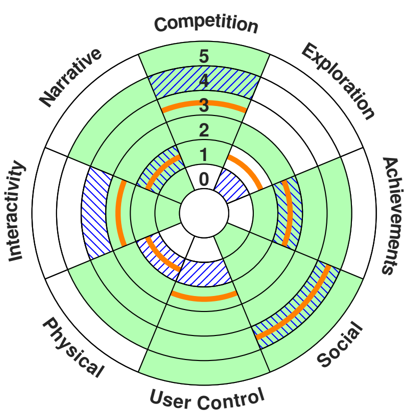


Figure 5: The new representation of a VEI profile. The green segments shows the values from the specification, the blue hatched segments the assessment by the designer, while the orange arcs show the opinion of an evaluation panel.

How is the VEI profile used?

In the design of an installation we anticipate three stakeholder groups: *a*) the science centre, represented by a curator; *b*) the designer, and *c*) the visitor. First, the curator specifies the desired ranges in the VEI Design Diagram. Here, the green segments show the areas that are envisaged by the science centre. Then, the designers design the installation and perform an assessment following the VEI profile table. This is shown with the blue hatches. Note that some of the blue hatches might be outside the specified green areas. If this happens, curator and designer need to discuss the reasons and take decisions what to change in specification or design.

In a third step, the visitors may give their opinion. This opinion is shown as the orange lines. Ideally, these orange lines should be in the fields with the blue hatches. If this is not the case, this could be a hint that the visitor's expectations are in mismatch with the designer's intention. We are currently working on how the visitor's feedback can be given without expecting that the visitor is a specialist. One working hypothesis suggests asking the visitor whether more or less of one dimension would be desirable.

The three steps that we described will be performed in several rounds as an iterative process, covering specific phases for concept, design, detailing, production, fabrication, installation, and post-opening activities.

Conclusion

We posit that the new VEI Design Diagram is a more suitable representation of the VEI profile than previously used spiderweb diagram as it avoids several of the experienced problems. The new diagram is a helpful tool in the design process to foster a basis for discussions between curator and designer, informed by the visitor. The VEI Design Diagram will be part of a novel interactive design- and development process for exhibits and installations [3].

References

[1] W. Leister, I. Tjøstheim, and G. Joryd. How to visualise the quality of installations? *NORSIGD Info*, 1/2015:9–10, 2015.

[2] W. Leister, I. Tjøstheim, G. Joryd, T. Schulz, A. Larssen, and M. de Brisis. Assessing visitor engagement in science centres and museums. *Journal on Advances in Life Sciences*, 8(1&2):49–63, 2016.

[3] W. Leister et al. An evaluation-driven design process for exhibitions. work in progress, 2017.

Acknowledgement. The work presented here has been carried out in the project VISITORENGAGEMENT funded by the Research Council of Norway in the BIA programme, grant number 228737.

Hva er NORSIGD?

NORSIGD – Norsk samarbeid innen grafisk databehandling – ble stiftet 10. januar 1974. NORSIGD er en ikke-kommersiell forening med formål *å fremme bruken av, øke interessen for, og øke kunnskapen om grafisk databehandling i Norge.*

Foreningen er åpen for alle enkeltpersoner, bedrifter og institusjoner som har interesse for grafisk datbehandling. Medlemskontingenten er 1.000 kr per år for institusjoner. Institusjonsmedlemmene er stemmeberettiget på foreningens årsmøte, og kan derigjennom påvirke bruken av foreningens midler.

Personlig medlemskap koster 250 kr per år. Kontingenten er redusert til 150 kr ved samtidig medlemskap i vår europeiske samarbeidsorganisasjon *Eurographics*. Ansatte hos institusjonsmedlemmer innvilges gratis personlig medlemskap.

Medlemsbladet *NORSIGD Info* utkommer 1–3 ganger per år og blir tilgjengelig på NORSIGD sine nettsider. NORSIGD har tilrettelagt informasjon om foreningen på Internett på adressen `http://www.norsigd.no`. Der finnes det også informasjon om GPGS, samt alle utgaver av *NORSIGD Info*.

Interesseområder

NORSIGD er et forum for alle som er opptatt av grafiske brukergrensesnitt og grafisk presentasjon, uavhengig av underliggende teknologi. NORSIGD arrangerer møter og seminarer, deltar i nettverks-aktiviteter og formidler informasjon fra internasjonale fora.

NORSIGD har lang tradisjon for å støtte opp om bruk av datagrafikk. Foreningen bidrar til spredning av informasjon ved å arrangere møter, seminarer og kurs for brukere og utviklere.

GPGS

GPGS er en 2D- og 3D grafisk subrutinepakke. GPGS er maskin- og utstyrsuavhengig. Det vil si at et program utviklet for et operativsystem med f.eks. bruk av plotter, kan flyttes til en annen maskin hvor plotteren er erstattet av en grafisk skjerm uten endringer i de grafiske rutinekallene. Det er definert grensesnitt for bruk av GPGS fra FORTRAN og C.

Det finnes versjoner av GPGS for en rekke forskjellige maskinplattformer, fra stormaskiner til Unix arbeidsstasjoner og PC. GPGS har drivere for over femti forskjellige typer utsyr (plottere, skjermer o.l.). GPGS støtter mange grafikkstandarder slik som Postscript, HPGL/2 og CGM. GPGS er fortsatt under utvikling og støtter stadig nye standarder.

GPGS eies av NORSIGD, og leies ut til foreningens medlemmer.

Eurographics

NORSIGD samarbeider med Eurographics. Personlige medlemmer i NORSIGD får rabatt på medlemskap i Eurographics, og vi formidler informasjon om aktuelle aktiviteter og arrangementer som avholdes i Eurographics-regi. Tilsvarende får Eurographics medlemmer kr 100 i rabatt på medlemskap i NORSIGD.

Eurographics ble grunnlagt i 1981 og har medlemmer over hele verden. Organisasjonen utgir et av verdens fremste fagtidsskrifter innen grafisk databehandling, *Computer Graphics Forum*. *Forum* sendes medlemmene annen hver måned. Eurographics konferansen arrangeres årlig med seminarer, utstilling, kurs og arbeidgrupper.

NORSIGD
v/ Reidar Rekdal
Postboks 290
1301 Sandvika

Returadresse:
NORSIGD v/ Reidar Rekdal
Postboks 290
1301 Sandvika

Styret i NORSIGD 2017

Funksjon	Adresse	Telefon	email
Leder og Sekretær	Trond Runar Hagen SINTEF Applied mathematics Pb. 124 Blindern 0314 Oslo	22 06 77 79 (direkte) 22 06 73 00 (sentralbord) 22 06 73 50 (fax)	TrondRunar.Hagen @sintef.no
Fagansvarlig	Wolfgang Leister Norsk Regnesentral Postboks 114 Blindern 0314 OSLO	22 85 25 78 (direkte) 22 85 25 00 (sentralbord) 22 69 76 60 (fax)	Wolfgang.Leister @nr.no
Varamedlem	Magnar Granhaug ProxyCom AS Klæbuvn. 194 7037 Trondheim	73 95 25 00 97 72 76 98 (mobil) 73 95 25 09 (fax)	Magnar.Granhaug @proxycom.no

Svarkupong

- ○ Innmelding – institusjonsmedlem (Kr 1000)
- ○ Innmelding – personlig medlem (Kr 250)
- ○ Innmelding – Eurographics medlem (Kr 150)
- ○ Ny kontaktperson
- ○ Adresseforandring

Navn: . . . .
Firma: . . . .
Gateadresse: . . . .
. . . .
Postadresse: . . . .
. . . .
Postnummer/sted: . . . .
. . . .
Telefon: . . . .
Telefaks: . . . .
email: . . . .