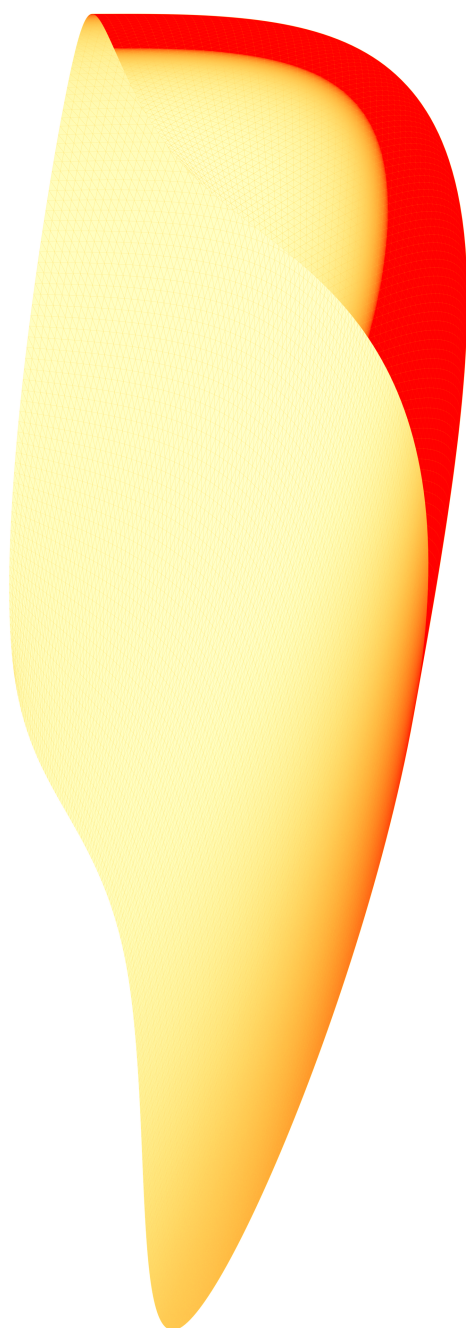




UNIVERSIDADE ESTADUAL PAULISTA

Instituto de Artes – Campus São Paulo

Francisco Ortiz de Carvalho

UMA TRIDIMENSIONALIDADE LIVRE, COLABORATIVA E
SÍNCRONA
TENDO *VERSE* E LOQ AIROU COMO ESBOÇO

São Paulo
2024

Francisco Ortiz de Carvalho

UMA TRIDIMENSIONALIDADE LIVRE, COLABORATIVA E
SÍNCRONA
TENDO *VERSE* E *LOQ AIROU* COMO ESBOÇO

Dissertação apresentada ao Programa de Pós-Graduação em Artes, com a área de concentração em Artes Visuais do Instituto de Artes Universidade Estadual Paulista (Unesp), como requisito parcial para obtenção do título de Mestre em Artes Visuais.

Linha de pesquisa: Processos e Procedimentos Artísticos.
Orientadora: Professora Doutora Livre Docente Rosângela da Silva Leote

São Paulo
2024

Ficha catalográfica desenvolvida pelo Serviço de Biblioteca e Documentação
do Instituto de Artes da Unesp. Dados fornecidos pelo autor.

C331t Carvalho, Francisco Ortiz de (ChicOrtiz), 1978-

Uma tridimensionalidade livre, colaborativa e síncrona tendo *verse* e Loq
Airou como esboço / Francisco Ortiz de Carvalho. -- São Paulo, 2024.
76 f. : il. color.

Orientadora: Prof.^a Dr.^a Rosangela da Silva Leote.

Dissertação (Mestrado em Artes) – Universidade Estadual Paulista “Júlio de
Mesquita Filho”, Instituto de Artes.

1. Arte e tecnologia. 2. Arte por computador. 3. Modelagem tridimensional.
4. Programas de computador. I. Leote, Rosangella (Rosangela da Silva Leote).
II. Universidade Estadual Paulista, Instituto de Artes. III. Título.

CDD 700.105

Bibliotecária responsável: Luciana Corts Mendes - CRB/8 10531

Francisco Ortiz de Carvalho

UMA TRIDIMENSIONALIDADE LIVRE, COLABORATIVA E
SÍNCRONA
TENDO *VERSE* E *LOQ AIROU* COMO ESBOÇO

Dissertação apresentada ao Programa de
Pós Graduação em Artes, do Instituto de
Artes da Unesp, como requisito parcial
para obtenção do título de Mestre em
Artes Visuais.

Dissertação aprovada em: 01 / 07 / 2024

Banca Examinadora

Professora Doutora Livre Docente Rosangela da Silva Leote
Orientadora

Professora Doutora Silvia Regina Ferreira de Laurentiz
Universidade de São Paulo

Professor Doutor Flavio Soares Correa da Silva
Universidade de São Paulo

Dedico este trabalho a meu filho João Braga Ortiz.

AGRADECIMENTOS

Ao Instituto de Artes da UNESP,
Aos colegas do grupo de pesquisa GIIP (Grupo Internacional e
Interinstitucional de Pesquisa
em Arte, Ciência e Tecnologia),

A minha orientadora Rosangela da Silva Leote,
Aos professores Silvia Laurentiz e Flavio Correa
Ou seja: a todos os docentes que encorajam,

À família que suporta o insuportável que por tantas vezes sou,
Aos meus amigos.

Um agradecimento especial à Silvia Golombek Zaclis pela ajuda na revisão de
português,

E também a Cristiani Pansonato Guessi Balieiro e Antonio Deusany de Carvalho
Junior por participarem como suplentes.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de
Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001

O desenho é de tanta excelência que não só busca as obras da natureza, mas infinitas outras mais do que as que a natureza produz.

Leonardo da Vinci

(Bosi, 2017)

RESUMO

Esta dissertação toma como esboço de dinâmicas de interação em grupo para construção de formas tridimensionais em meios virtuais e síncronos os programas *verse server* e Loq Airou. Tais programas foram desenvolvidos a partir de um projeto da União Europeia denominado *Uni-Verse*. Assim sendo, os programas *verse server* e o modelador minimalista poligonal compatível com trabalho em rede Loq Airou são subprodutos deste esforço maior, servindo então como alegoria desse “universo” de programas e possibilidades artísticas de criação coletiva a partir do meio eletrônico.

Palavras-chave: Uni-Verse, Loq Airou, verse protocol, Blender 3D, União Europeia, sincronicidade, software de modelagem 3D, programa de computador, aplicativo.

ABSTRACT

This dissertation takes as an outline group's interaction dynamics for constructing three-dimensional forms in virtual and synchronous media the softwares verse server and Loq Airou. Such programs were developed from an European Union Project called Uni-Verse. Therefore, the softwares "verse server" and the minimalist polygonal modeler compatible with networking "Loq Airou" are by-products of this greater effort, functioning as an allegory of this "universe" of programs and artistic possibilities for collective creation using the electronic medium.

Keywords: Uni-Verse, Loq Airou, verse protocol, Blender 3D, European Union, synchronicity, 3D modeling software, computer software, application.

SUMÁRIO

1 INTRODUÇÃO.....	11
1.1 APRESENTAÇÃO.....	11
1.2 SOBRE GEOMETRIA.....	12
1.3 TEMA E PROBLEMA. O PROJETO <i>UNI-VERSE</i> DE RELANCE.....	14
1.4 JUSTIFICATIVA. A ALEGORIA DA TRANSPARÊNCIA.....	15
1.5 OBJETIVOS. CÓPIAS DE SEGURANÇA E ANÁLISE.....	16
2 METODOLOGIA. DIÁLOGO COM PÚBLICO DE DIFERENTES NÍVEIS DE FORMAÇÃO.....	17
3 DESENVOLVIMENTO. SOBRE A PERSPECTIVA CÔNICA EM ESCALA E UM MODELADOR 3D MINIMALISTA.....	31
3.1 SOBRE A ESTRUTURA DO MODELADOR LOQ AIROU.....	47
3.2 SUGESTÕES DE ARRANJO PARA TRABALHO EM REDE.....	53
4 CONCLUSÃO.....	56
REFERÊNCIAS.....	59
APÊNDICE A: RESENHA QUE TRADUZ GRAFICAMENTE ARTIGO DE STEENBERG SOBRE SUBDIVISÃO DE SUPERFÍCIES.....	63
APÊNDICE B : SCRIPTS CONVERSORES DE .XML PARA WAVEFRONT .OBJ..	68

1 INTRODUÇÃO

1.1 APRESENTAÇÃO

A redação desta dissertação foi desenvolvida priorizando o entendimento de um leitor de artes.

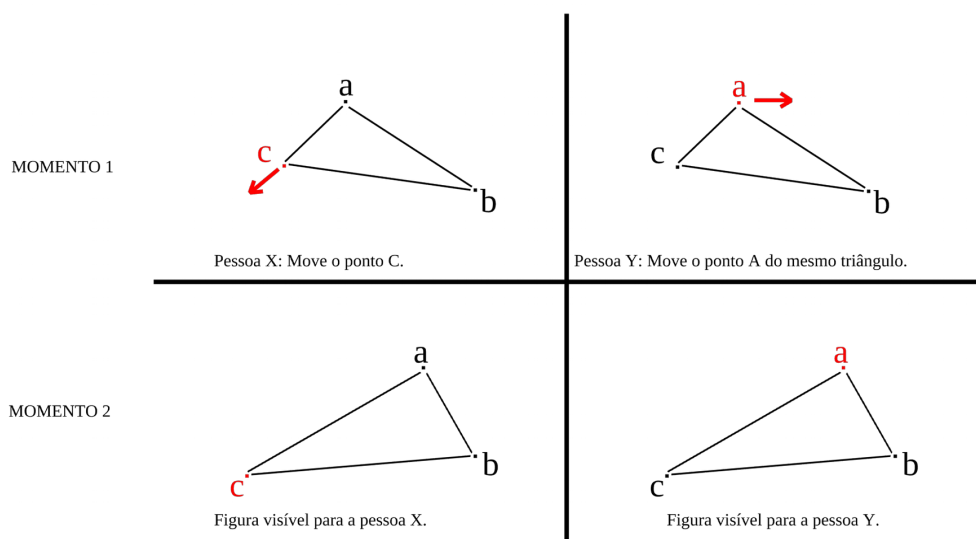
Outros leitores, de outras áreas, também são bem-vindos. No entanto, vale ressaltar que talvez haja algum estranhamento quanto a sequência expositiva. Isso porque se preferiu uma abordagem intimista de diálogo com o leitor.

Este modelo de diálogo visa apresentar temas que são próprios de geometria e de programação, de ferramentas de computador, com o intuito de oferecer ao artista uma visão crítica da relação que este pode estabelecer com o meio digital. Um subproduto possível é a conservação de geometrias tridimensionais (e por que não dizer escultóricas?) desenvolvidas neste meio para que durem muitos anos.

1.2 SOBRE GEOMETRIA

O plano geométrico mais simples que existe é um triângulo. No contexto da modelagem tridimensional tendo o meio digital como ferramenta, o triângulo por vezes recebe outros dois nomes: face e polígono. Imaginemos que um triângulo possa se reconfigurar a partir da interação entre duas pessoas, cada qual utilizando um computador, ligados em rede, e que as modificações ocorridas neste triângulo sejam vistas ao mesmo tempo, e sem atrasos, por estas duas pessoas. Que tipo de interações podem surgir daí? Supondo que estas duas pessoas tenham escolhido vértices (o nome de pontos quando estamos em um contexto de tridimensionalidade) diferentes e aplicado um movimento em cada um deles teríamos alguma situação parecida com a Figura 1.

Figura1 - Sincronia em um triângulo



Fonte: elaborada pelo autor (2017)

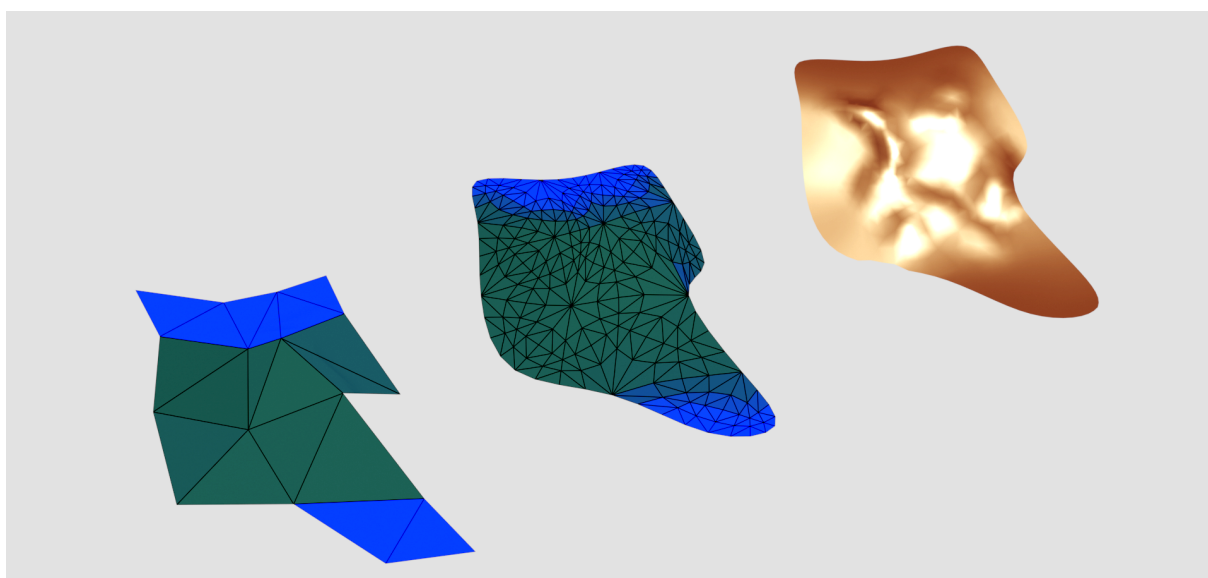
Em um ambiente com visualização tridimensional, o triângulo poderá ser visto de ângulos diferentes, o que pode levar à conclusão precipitada de que são triângulos diferentes, enquanto trata-se apenas de uma mudança de ponto de vista.

Caso as duas pessoas que interagem com o triângulo escolham o mesmo vértice para movimentar, podemos ter uma situação do tipo “cabo de guerra”, onde uma pessoa puxa o ponto para uma determinada direção que não coincide com a direção escolhida pelo outro.

É claro que, do ponto de vista plástico, a interação com apenas um triângulo talvez não seja muito empolgante para dois artistas que estejam experimentando este sistema. Mas, se permitirmos que se construam mais triângulos, perfeitamente conectados formando uma superfície, o apelo para a experimentação pode se tornar maior.

Também podemos imaginar que, sobre a superfície construída, algum tipo de cálculo possa ser aplicado automaticamente para torná-la mais orgânica, bem como algum tratamento de luz e sombra para que seu entendimento visual seja mais claro. Em havendo representação da iluminação nessa superfície, talvez possamos até mesmo abstrair o fato de ela ser formada por triângulos, adicionando agora detalhes como fazemos na modelagem em argila, como ilustrado na Figura 2

Figura 2 - Uma mesma superfície formada por triângulos evoluindo em complexidade e nível de representação gráfica



Fonte: elaborada pelo autor (2024)

Outro elemento que poderá empolgar mais ainda esta construção coletiva será a possibilidade de mais do que duas pessoas possam interagir no ambiente conectado inicialmente sugerido.

1.3 TEMA E PROBLEMA. O PROJETO *UNI-VERSE* DE RELANCE.

Nossa proposição se origina do estudo de um projeto de parceria gestado por instituições europeias, aprovado pela União Europeia para receber fundos de execução no ano de 2002.

Liderava a parceria o Instituto Real de Tecnologia sueco (KTH); Os outros parceiros eram: Instituto Interativo e Agência de Comunicação Digital Paregos, também da Suécia, Universidade de Tecnologia de Helsinque, da Finlândia, Instituto de Pesquisa Fraunhofer IGD, da Alemanha, escritório de arquitetura *Minusplus Architects* da Hungria e Fundação *Blender*, da Holanda. Posteriormente, o desenvolvimento ficou concentrado na Universidade Técnica de Liberec, da República Tcheca (Hnidek, 2011).¹

Esse projeto assumiu o nome de *Uni-Verse*, tendo recebido uma dotação de aproximadamente dois milhões de Euros. A parte alocada para pesquisa e desenvolvimento de programas de computador, no entanto, foi apenas uma fração desta quantia.²

Em sendo o projeto *Uni-Verse* muito extenso, optamos por nos concentrar nas dinâmicas de interação humanas coletivas e síncronas que os programas *verse server* e *Loq Airou*, de modelagem e navegação tridimensional, podem oferecer; refletindo que estas interações podem ser genéricas o suficiente para serem aplicadas a outros modelos.

Essas dinâmicas, no entanto, estão longe de ser perfeitas. Nesse sentido, esta dissertação procura servir como guia das idiossincrasias que um artista terá de enfrentar caso queira estabelecer um servidor próprio que permita uma modelagem

1 HNIDEK, Jiri. **Network Protocols for Applications of Shared Virtual Reality**. Liberec: Technical University of Liberec, 2011.

2 FÓRUM da Fundação Blender. Disponível em: <https://www.blender.org/forum/viewtopic.php?t=2952&p=16829>. Acesso em: jul. 2017.

tridimensional síncrona em rede com dois ou mais participantes, amparada por sistemas abertos e livres.

1.4 JUSTIFICATIVA. A ALEGORIA DA TRANSPARÊNCIA.

O mapeamento parcial que propomos surge, então, como uma alegoria, se entendermos aqui a palavra alegoria como uma ideia mais simples capaz de ilustrar algo mais complexo. Ao mesmo tempo nossa intenção neste trabalho é a transparência total, a publicação em domínio público a partir de componentes que também são de domínio público ou estejam no limiar do domínio público; visando a facilitar o entendimento de estudantes. Os programas escolhidos requerem computadores modestos, o que democratiza o acesso e permite que pesquisadores com maior bagagem teórica e melhor maquinário possam trabalhar em equipe, por exemplo, com estudantes que ainda estão se iniciando na matéria em questão e que, eventualmente, não possuam um equipamento de última geração.

O Raio X é uma metáfora que nos vem à mente para ilustrar a importância da transparência no sistema que propomos analisar. Se para localizar com precisão uma fratura utilizando equipamentos de raios-x pode ser um desafio, imaginem quão difícil e imprecisa seria a tarefa se não tivéssemos este recurso.

Outra questão importante presente no projeto *Uni-Verse* é a possibilidade de se estabelecer um servidor e conectar clientes de maneira independente de serviços oferecidos por grandes empresas da internet, empresas essas que têm ultimamente agido muitas vezes à revelia de valores constitucionais democráticos. Nesse sentido o projeto *Uni-Verse* - e mais especificamente o servidor *verse* e o modelador *Loq Airou* - tem muito a oferecer do ponto de vista didático para aqueles que querem aprender os fundamentos de um serviço de comunicação voltado à modelagem tridimensional (nosso foco); mas não apenas isso, pois o escopo do projeto *Uni-Verse* é ainda mais abrangente.

1.5 OBJETIVOS. CÓPIAS DE SEGURANÇA E ANÁLISE.

Posto o contexto do problema e da justificativa, surge nosso objetivo: documentar as versões dos programas estudados que consideramos mais adequadas para que artistas visuais possam experimentar com maior facilidade o tipo de interação que, de outra maneira, poderiam levar anos para decifrar. Ao mesmo tempo, indexamos cópias de segurança do código-fonte dos programas em questão; essa medida pode beneficiar grupos de artistas pesquisadores que vierem a trabalhar com cientistas da computação, visando assim aperfeiçoar o sistema, testando-o em contextos educacionais, de prototipagem e pesquisa plástica tridimensional.

Acreditamos que o trabalho seja útil também, por exemplo, para a disciplina de Linguagem Tridimensional, que é parte do núcleo comum da licenciatura e bacharelado do Instituto de Artes; e ainda como apoio no diálogo as disciplinas de Linguagem Tridimensional e de Mídia.

2 METODOLOGIA. DIÁLOGO COM PÚBLICO DE DIFERENTES NÍVEIS DE FORMAÇÃO.

Método: Do grego μέθοδος; métodos; caminho para chegar a um fim; pelo latim methodu.

Logia: Do grego λόγος, λογότυπα indica o estudo de determinada matéria.

Ou ainda λογική: Lógica.

Portanto: lógica do caminho.

Metodologia: do grego métodos, método. λόγος, tratado, e sufixo ia.

(Nascentes, 1932)

O caminho que propomos para delimitar as virtudes, dificuldades e deficiências de se trabalhar com um programa de computador similar ao estudado passa por citar brevemente a reação do público em quatro diferentes contextos de ensino e aprendizagem no período de 2010 a 2018, quando a criação coletiva com os programas de computador estudados estava em ação.

Para além deste critério, também é procedimento recomendável decifrar a estrutura institucional organizacional que existe por trás do desenvolvimento de um programa como *verse*, pois nos coloca numa posição de vantagem em longo prazo, especialmente no contexto de pesquisa, ensino e aprendizagem: Conhecendo qual será o comprometimento ético dos desenvolvedores ao longo do tempo, teremos uma base sólida para trabalhar com maior confiança, resiliência e transparência.

Os contextos de ensino e aprendizagem que analisamos são:

a) Centros culturais, sendo a maior parte os localizados em unidades do SESC São Paulo

b) Turmas de Ensino Fundamental II em escolas particulares

Escola Projeto Vida e Escola Nossa Senhora das Graças (“Gracinha”), onde foram atendidos por volta de 260 alunos de 2010 a 2019.

c) Ambientes universitários da Rede Pública: grupo de extensão universitária (UspGameDev) e palestras pontuais com público maior em relação aos de extensão universitária (FATECs).

d) Diálogo com especialistas. Yorik van Havre, Jacques Corby Tuech, David Oreilly, Rodrigo Dorta Marques e Maurício Valerini

A inclusão destes ambientes é relevante para delimitar com maior clareza a qual leitor e área de interesse esta dissertação se destina, bem como os recursos materiais necessários para sua aplicação. Ao mesmo tempo identificamos as deficiências e virtudes do modelo estudado caso alguém resolva prototipar sobre a mesma base a fim de aperfeiçoá-la ou mesmo implantar alterações mais profundas.

Em relação a Centros Culturais, destaca-se o trabalho de oficinas realizadas nas unidades do Serviço Social do Comércio (SESC), onde pudemos propor atividades nas antigas “Internet Livres”, atuais “Espaços de Tecnologia e Artes” (ETAs) em diferentes unidades do SESC na capital e no interior de São Paulo.

De maneira geral, quando o espaço voltado a atividades com computador era denominado “Internet Livre”, ele era organizado em torno de computadores com torre de processamento, *desktops*, ligados entre si em rede local com sistema operacional *Linux Ubuntu*. Quando iniciamos nossos experimentos nesse arranjo, limitados à versão *Linux* dos programas estudados e com pouco acesso às máquinas, era impraticável configurar o programa *Wine Is Not an Emulator* “W.I.N.E”.³ A partir desta opção, de saída tivemos um problema relacionado à interpretação da construção da geometria: a tela inicial do modelador, após a abertura e campo para endereço de conexão, não possui nenhuma imagem de referência. Além disso, a implementação do algoritmo Catmull Clark⁴ não se atualiza em tempo real, o que cria mais uma dificuldade na capacidade de o público interpretar o que está ocorrendo no *software*. Conseguimos mitigar este problema

3 WINE Is Not an Emulator: W.I.N.E Disponível em: <https://www.winehq.org/> Acesso em: maio 2024.

4 CATMULL, E; CLARK, J. **Recursevely generated B-spline surfaces on arbitrary topological meshes** IPC Bussiness Press computer-aided design, 1978.

depois de muita tentativa e erro; deduzindo que na câmera inicial do modelador, que não possui nenhuma representação física; estamos sempre no interior da mesma, iniciando num plano paralelo aos eixos X e Y; logo, é conveniente que se comece criando uma forma retangular a partir destes eixos, para que se possa voltar à visualização inicial, possibilitando voltar aos eixos cartesianos com maior facilidade.

Outro elemento que dificultou o desenvolvimento dessas atividades de oficina foi a questão dos “endereços de IP”. Inicialmente, acreditamos que não haveria problema se tivéssemos diversos servidores rodando na mesma rede, porém estávamos enganados. Os conflitos advindos dessa configuração podem surgir quando diversos programas cliente tentam se conectar a diferentes servidores em uma rede única, gerando sobrecargas. Nesse momento, os clientes vão perder a capacidade de utilizar os servidores, nos obrigando a reiniciar o sistema como um todo.

Em sendo adulto, o público recebeu com abertura as propostas das oficinas. No início, enfatizávamos que a atividade teria um caráter bastante experimental e que seria comum encontrarmos erros. Ainda que muitos participantes não soubessem modelar tridimensionalmente, a iniciativa frente à atividade proposta adicionava elementos de aprendizado e interação coletiva.

Outra tentativa que fizemos foi tentar nos conectar a um servidor externo à rede que estávamos utilizando, recorrendo à internet para este fim. A partir desta tentativa descobrimos três coisas: a rede de uma unidade é protegida quando se tenta acessar um recurso externo o termo técnico em inglês, nos contaram os administradores – É *Proxy* ou *Proxies*.

Além disso, existia a possibilidade de o endereço dinâmico do servidor mudar. Sem contar que é possível que exista algum tipo de proteção onde o servidor ficou instalado, o que foi o caso certa vez, quando deixamos o servidor-teste ligado na rede de um hotel próximo ao local tentando acessá-lo a partir das máquinas do Centro Cultural.

Em outro contexto, pudemos testar a modelagem tridimensional colaborativa em escolas da rede particular de Ensino Fundamental II, atuando, a exemplo dos anteriores,- como pessoa jurídica tendo instrução como objetivo social.

Nessas escolas encontramos um arranjo de computadores muito parecido com o de centros culturais, porém com sistema operacional *Windows XP*, *Windows Vista* e *Windows 7*. Ainda que não sejam nossos sistemas operacionais favoritos por não serem abertos e livres, tivemos a vantagem de poder utilizar a versão do modelador com o algoritmo Catmull Clark atualizando em tempo real, o que facilitou bastante a leitura das formas que estavam sendo construídas. Os estudantes, no entanto, tiveram muita dificuldade em construir formas que estivessem subordinadas aos eixos cartesianos. Para ajudá-las, construímos algumas formas elementares e pedimos que as interligassem. A maior parte das crianças se sentiu mais atraída por desfazer as formas construídas do que por criar outras formas. Nesse contexto, os estudantes que tentavam criar formas se sentiam frustrados neste “ambiente hostil emergente”.

Para contornar este problema, ensinamos aos “alunos construtores” como criar outros “nós” de geometria, onde poderiam trabalhar de maneira independente e depois copiar os elementos para o nó principal, estabelecendo assim uma espécie de jogo de “monta e desmonta”. Portanto a recepção foi carregada de elementos infantis, que não deixavam de ser ludicamente interessantes.

Por ser uma contratação de longo prazo tivemos a vantagem de poder testar os programas com maior tranquilidade, e aprofundar o conteúdo didático, o que se mostrou uma vantagem em relação aos centros culturais.

Com o tempo, os computadores de torre, tanto das escolas como de centros culturais, foram sendo substituídos por computadores portáteis (*notebooks*, antigamente chamados de *laptops*) conectados à *internet* via *WiFi*. Então, para seguir oferecendo esta atividade, adquirimos um pequeno *Hub / Switch* de rede, que foi bastante proveitoso, pois nos deu a oportunidade de introduzir também uma atividade de montagem de cabos do tipo de par trançado. Acabamos tendo menos tempo para testar o modelador funcionando em rede, pois antes de começar a atividade era preciso remontar a rede toda vez. Este problema, no entanto, pôde ser contornado com o uso de um roteador funcionando localmente, que nos permitiu estabelecer as conexões rapidamente.

Outro contexto a ser mencionado se refere à apresentação do funcionamento dos programas a ambientes universitários como em palestras e em participações

em grupos de extensão e pesquisa. Nosso objetivo inicial era verificar se outras pessoas teriam interesse em estudar conosco o código fonte dos programas em questão. De forma resumida apresentamos os programas estudados em três instituições: Faculdades de Tecnologia do Estado de São Paulo (FATECs), a convite dos estudantes; Instituto de Matemática e Estatística da Universidade de São Paulo (IME USP) e grupos de pesquisa do Instituto de Artes da Universidade Estadual Paulista (IA – UNESP).

Nas FATECs foram duas palestras que foram recebidas com poucas perguntas dos estudantes. Na FATEC Guarulhos, os alunos nos ajudaram a configurar uma rede com servidor *Linux* com clientes *Windows* e vice-versa, fornecendo uma troca de aprendizado preciosa.

Já no IME USP aprendemos rudimentos de compilação de programas utilizando o *GNU's Not Unix Compiler Collection* (GCC) com o comando *make* e *Makefile*. Uma descoberta importante foi que, quando usamos comandos desse tipo para compilar os programas do projeto *Uni-Verse*, pode-se criar uma sequência de execução; por exemplo, para que seja compilado o modelador Loq Airou, o servidor *verse* deve ser compilado anteriormente; em outras palavras: para se obter a compilação de Loq Airou, a compilação do servidor *verse* é um pré-requisito.

Outro aprendizado muito interessante no que se refere à compilação de programas que aprendemos com colegas do IME foi a necessidade de observar quais dependências devem ser instaladas no sistema para que a compilação tenha êxito. Aqui nos deparamos com um paradigma interessante: do ponto de vista lógico, parece ideal que tenhamos todas as dependências instaladas localmente de tal maneira que possamos estudar todos os componentes *offline*. O problema é que este procedimento ideal pode se mostrar extremamente contraproducente pois assim as dependências poderiam ocupar um espaço gigantesco na memória de armazenamento local. Dessa maneira, pudemos constatar que uma visão global do projeto em termos de interpretação de código é bastante difícil para aqueles que não são da área de desenvolvimento de programas de computador. Só resta nos conformarmos com um entendimento na maior parte das vezes profundamente fragmentário da estrutura do código fonte da parte de profissionais provenientes das áreas de Humanas. Isso em nome de uma exploração plástica que, em jargão de

Exatas, se diria de “mais alto nível” em relação ao programa de computador / *software*. Poderíamos dizer então que quando o trabalho do matemático termina, começa o trabalho do artista; uma vez criado o programa, o matemático não tem muito interesse por sua implementação e uso, a não ser para adicionar correções ou obter dados quantitativos; toma esta decisão sob risco de perder contato com sua área de especialização sendo o avesso verdadeiro para o artista.

Essa cisão no processo criativo poderíamos classificar como um divórcio. Talvez uma maneira menos melancólica de lidar com este assunto seja encará-lo como uma corrida de revezamento.

Além das formas de participação acima mencionadas, tivemos também a oportunidade de interagir com especialistas.

Infelizmente estas interações foram bastante exíguas, devido à dificuldade técnica de se configurar a rede e a liberação de agenda junto às pessoas com quem conversamos.

O primeiro especialista com quem conversamos foi Yorik van Havre, arquiteto belga formado pelo *Institut Supérieur d'Architecture St-Luc* (agora parte da *UC Louvain*), desenvolvedor do projeto *FreeCad*, especialista em 3D, croquis urbanos e histórias em quadrinhos. Ainda que Yorik seja de origem belga, no tempo em que desenvolvemos este trabalho ele estava morando no bairro de Santa Cecília em São Paulo com sua então companheira Maíra Zasso, ambos mantendo o Escritório de Arquitetura *Uncreated*⁵. Yorik nos ensinou a configurar um roteador específico e fazer algumas sessões de desenho com o programa Loq Airou. Yorik inicialmente avaliou que o programa era bastante instável pois a janela de Loq Airou fechava bruscamente após alguns minutos de uso. A janela do servidor *verse*, no entanto, nunca vimos fechar. Aventamos a hipótese de Loq Airou cair devido à mudança de endereço IP dinâmico.

Passado algum tempo, lembramos de um serviço que os desenvolvedores da *Insite*, uma empresa de hospedagem de *websites* em São Paulo operando no Edifício Itália, depois adquirida pelo Grupo Universo *Online HOST* (UOL HOST) utilizavam para fazer testes: o projeto de compartilhamento de domínios *Domain*

5 ESCRITÓRIO de Arquitetura Uncreated. Disponível em: <https://uncreated.net/>
Acesso em: ago. 2024

Name System (DNS) mooo.com que existe até a presente data⁶. A partir deste projeto pudemos associar de maneira criptografada nosso IP dinâmico com um nome simples como, por exemplo, “chicortiz.moos” o que facilitava na hora de preencher o campo de autenticação na tela inicial do modelador, fazendo também com que as quedas praticamente não ocorressem. Com a posse deste aprendizado, descobrimos também outra maneira de deixar o funcionamento do modelador Loq Airou ainda mais resiliente: no computador que montamos o servidor *verse*, não criar e conectar nenhum cliente local, a menos que trabalhemos sozinhos (*localhost*).

Fruto das experimentações com Yorik foi produzido também um vídeo representativo das interações com o modelador. Além de nossas memórias de armazenamento locais fizemos *backup* (cópia de segurança) nos links a seguir.

Na plataforma *online* YouTube:

SYNCHRONY

<https://youtu.be/yMvjg-r9XVM>

Acesso em abril de 2024.

No armazenamento *online* oferecido pelo Instituto de Artes da UNESP (contrato com a plataforma *Google Drive*):

SYNCHRONY

<https://drive.google.com/file/d/1uE0dhn9a7X-3zn8g3tcqcByRMvHC2zYg/view?usp=sharing>

Acesso em Agosto de 2024.

E também em domínio próprio (arquivo *Synchrony.mp4*):

SYNCHRONY

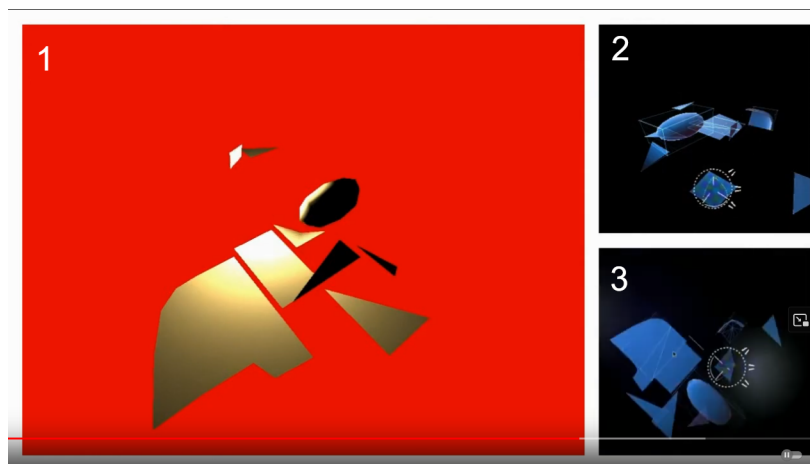
prototipe.com.br/backup_produtos_uni_verse/Synchrony.mp4

Acesso em abril de 2024.

⁶ DOMAIN Name System. Disponível em: <http://moos.com/> Acesso em: jul. 2024

Para que se compreenda com maior clareza os eventos ocorridos no vídeo em questão, criamos a Figura 3 com as indicações 1, 2 e 3:

Figura 3 - captura de tela de modelagem 3d colaborativa, livre e síncrona



Fonte: elaborada pelo autor (2024)

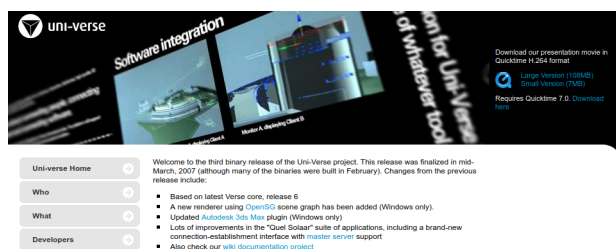
Em “1” temos a situação no servidor *verse* desenhada pelo programa Quelsolaar, que é uma máquina de renderização e visualização em tempo real também produto do projeto *Uni-Verse*. Em “2” temos um registro do modelador sendo operado a partir do computador “A”. E finalmente em “3” temos uma captura da situação do modelador Loq Airou sendo operado a partir do computador “B”. Estas imagens mostram o mesmo objeto tridimensional em diferentes ângulos no mesmo instante temporal. Desta forma, temos uma representação estática de algo que ocorreu “em tempo real”. Outro fator relevante é que o vídeo acima mencionado está acelerado em relação ao fato ocorrido, tratando-se de um vídeo do tipo *timelapse*.

No caso do modelador Loq Airou, a ênfase está na manipulação evidente de vértices, arestas e faces. Estes termos guardam uma equivalência com pontos, linhas e lados. A diferença é que a primeira terminologia é usada para definir que estamos em um contexto tridimensional. Já a segunda indica um contexto de bidimensionalidade.

O fato de Loq Airou dar ênfase a uma construção cuidadosa de polígonos, não exclui a possibilidade de este modelador operar com outros modeladores tridimensionais onde a presença dos polígonos fica menos evidente, como é o caso do *Sculpt Mode*

do programa *Blender 3D*; com sua aparência de uma argila ou massinha de modelar. Não nos esqueçamos que a Fundação *Blender* foi uma das integrantes do consórcio *Uni-Verse*, e o protocolo *verse* permite a criação de extensões (chamadas de *plugins* ou *addons*). Estas extensões permitem que outros programas possam trabalhar em conjunto no mesmo ambiente de rede estabelecido pelo protocolo, inclusive programas proprietários tais como *3D Studio Max*, *Maya* e *ZBrush*, dentre outros. A página original do projeto *Uni-Verse* apontava que já existia uma extensão para o *3D Studio Max*. Ou seja: ainda que o nosso escopo foque em um contexto livre onde possamos consultar o código fonte, todos são bem-vindos na hora da criação coletiva síncrona, podendo escolher para isso seu *software* de modelagem tridimensional predileto, tal como a Figura 4 anunciava.

Figura 4 - captura de tela do website projeto *Uni-Verse* onde a extensão para *3D Studio Max* é mencionada



Fonte: Consórcio *Uni-Verse* (2011)

Seguindo na interação com Yorik, em um dado momento, para observar sua reação, destruímos uma parte da geometria da superfície que estava sendo construída, tal como os estudantes de ensino fundamental costumavam fazer, por brincadeira, nas aulas extras de 3D. Este movimento surpreendeu Yorik. Perguntamos a ele se desejava que utilizássemos a função desfazer.

Explicamos a ele que, muitas vezes este tipo de intervenção ocorria também pela maneira como a ferramenta de apagamento funcionava: por não haver uma representação física de câmeras virtuais em Loq Airou, um tipo específico de apagamento (desenhar uma linha que intercepta o trecho da geometria que queremos apagar) faz com que linhas que estejam “nas costas” do usuário por assim dizer, possam ser apagadas também. Essa forma de apagar específica pode ser observada depois do primeiro minuto do vídeo mencionado acima (*Synchrony*).

Hamlet: Ato V, Cena 1. (Estão presentes Hamlet, Horácio e o 1º Coveiro).

1º Coveiro: — ...Aqui tendes um crânio. Este crânio esteve enterrado durante vinte e três anos.

Hamlet: — A quem pertencia?

1º Coveiro: — A um louco filha da mãe. Quem acreditais que seja?

Hamlet: — Não posso saber.

1º Coveiro: — Que a peste caia sobre ele! Louco imbecil! Um dia esvaziou-me na cabeça uma garrafa de vinho do Reno. Este crânio que estais vendo pertenceu a Yorick, o bobo do rei.

Hamlet: — Esse?

1º Coveiro: — Exatamente.

Hamlet: — Deixa-me vê-lo (segura o crânio) Ai! Pobre Yorick! Eu o conheci Horácio: era um homem engraçadíssimo e de fantasia portentosa. Mil vezes me carregou nas costas e, agora, sinto horror ao recordá-lo! Meu estômago até se revolta! Aqui pendiam aqueles lábios que eu beijei não sei quantas vezes. Que fizeram de teus sarcasmos, de tuas cabriolas, de tuas canções, de teus rasgos de bom humor, que faziam toda mesa prorromper em gargalhada? Nada, nem uma só graça sequer para ridicularizar sua própria careta? Tudo descarnado? Vai agora ao toucador de minha senhora e diz-lhe que coloque a mais espessa camada de pintura, nada impedirá que venha ela para aqui. Faze-a rir com isso!

(Shakespeare, 1623)

Afirmamos no início deste segmento que este vídeo é uma captura representativa; outro elemento que justifica esta definição é que, aqui, a linguagem tridimensional comum ao designer, ao arquiteto, ao artista escultor foi reduzida a seu mínimo. Deste fato surge uma partitura na qual se torna possível, neste contexto específico, sincronizar não apenas dois computadores, mas duas mentes com uma formação profissional similar. É muitas vezes possível até mesmo antecipar o movimento da pessoa (ou das pessoas) com quem estamos interagindo de uma maneira que anteriormente não seria possível com tanta rapidez; num improviso similar ao estilo musical do *Jazz* por exemplo.

Outra tentativa feita para tentar organizar uma pequena comunidade interessada em utilizar modeladores tridimensionais minimalistas ocorreu através do fórum *gaminglove.net*⁷ no ano de 2011. Este era um site dedicado ao jogo

7 INTERNET ARCHIVE: captura Fórum *Gaming Love*. Disponível em: <https://web.archive.org/web/20120508075406/http://www.gaminglove.net/> Acesso em: maio 2024.

multijogador “Love” que também utilizava a tecnologia do servidor *verse* e Loq Airou. Quem mantinha este site naquela época era o então estudante da Universidade Oeste de Londres de operações de receita Jacques Corby-Tuech.

A partir deste *website* foi iniciada uma postagem convidando os frequentadores a participarem de sessões com o modelador multiusuário minimalista. Infelizmente não houve muito interesse. Então tentamos dialogar com outra comunidade que trabalhava a modelagem com poucos polígonos e se organizava em torno do artista de animação irlandês David Oreilly. O trabalho de Oreilly encontrou maior projeção depois que ele criou o *videoclip* da banda U2 *I'll Go Crazy If I Don't Go Crazy Tonight*⁸. Naquela época, Oreilly mantinha um fórum de discussões online chamado *lowend3d*⁹. Esta tentativa de conexão não logrou êxito uma vez que o fórum *lowend3d* foi descontinuado logo depois.

Ainda dentro das interações com especialistas também entramos em contato com nosso colega Rodrigo Dorta Marques, Bacharel em Artes Visuais pela Universidade Federal de Pelotas e Doutorando pelo Instituto de Artes da UNESP, especialista em *Game Studies*.

Ao retomar as interações com especialistas, em dialogo com Rodrigo, constatamos que algo no contexto da rede de internet havia mudado: o roteador que pretendíamos conectar ao servidor agora pertencia ao fornecedor de serviços de internet, neste caso específico à Operadora Vivo Telefônica S/A,¹⁰ Assim, não nos foi possível estabelecer imediatamente uma rede de maneira não burocrática. Então, nosso diálogo se restringiu à interface do programa de modelagem tridimensional Loq Airou e sua interação com o servidor *verse* e a marcar uma nova sessão quando descobríssemos como estabelecer a conexão novamente. Durante este diálogo, Rodrigo apontou que a interface em questão não parecia algo voltado a uma produtividade pragmática, se assemelhando mais a um programa de *Vjing* (uma designação bem ampla para performances artísticas visuais em tempo real). Também discutimos sobre o fato de como o algoritmo Catmull Clark ter sido implementado nesta ferramenta, ou seja, a presença de *Level of Detail* (L.O.D) que é

8 U2 I'll Go Crazy If I Don't Go Crazy Tonight. Disponível em: https://www.youtube.com/watch?v=4q_AHAMVQ9c Acesso em abr. de 2024.

9 INTERNET ARCHIVE: Captura do Fórum de discussões *lowend3d* Disponível em: <https://web.archive.org/web/20110202173944/https://www.lowend3d.com/> Acesso em: abr. 2024.

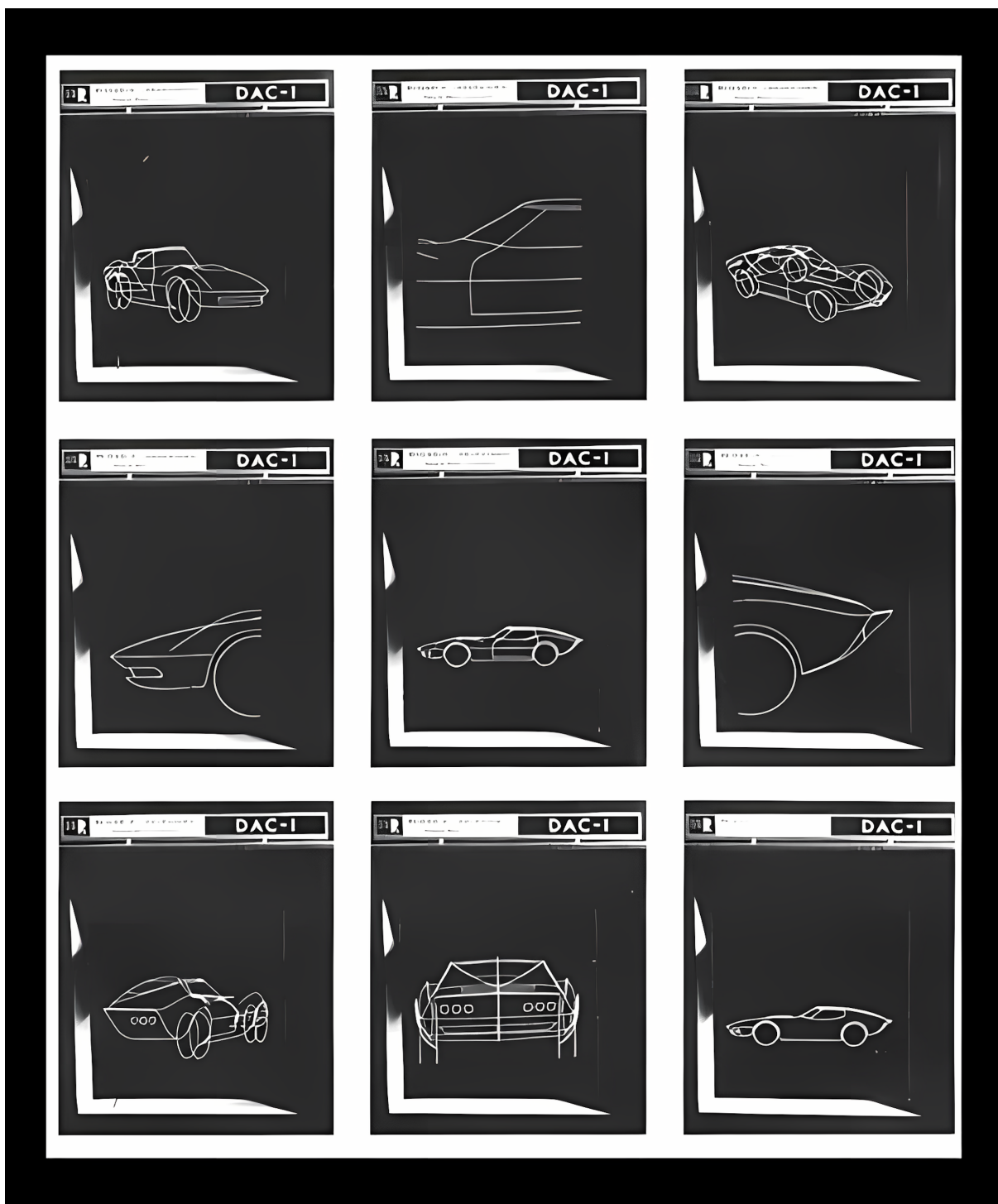
10 OPERADORA Vivo Telefônica S/A Disponível em: <https://vivo.com.br> Acesso em: abr. 2024. Cadastro Nacional de Pessoa Jurídica (CNPJ): 02.558.157/0001-62.

a variação da quantidade de polígonos conforme a distância do observador: mais perto, mais polígonos e mais detalhes; mais longe menos polígonos são necessários, logo menos detalhes são tratados graficamente pelo programa. Esta solução melhora substancialmente o desempenho do programa e adiciona outra camada de elegância ao já competente e clássico algoritmo. Ao conversarmos sobre este tópico, imediatamente Rodrigo mencionou também Ivan Sutherland, informático pioneiro estadunidense que trabalhou com o cientista da computação Edwin Catmull e o influenciou. Assim foi possível verificar uma ligação entre as pesquisas que desenvolvemos, mais especificamente o estudo da evolução das interfaces gráficas. A interface gráfica de Loq Airou fez Rodrigo lembrar de interfaces dos anos 1970 tais como a do computador DAC 1¹¹ que vinha sendo desenvolvido desde os anos 1960 e cuja interface pode ser vista na Figura 5.

Curiosamente, o aluno do IME Lucas Hong, durante palestra que apresentamos em 2016, também achou que a interface de Loq Airou, representada pela figura 6, remetia a programas de computador dos anos 1970.

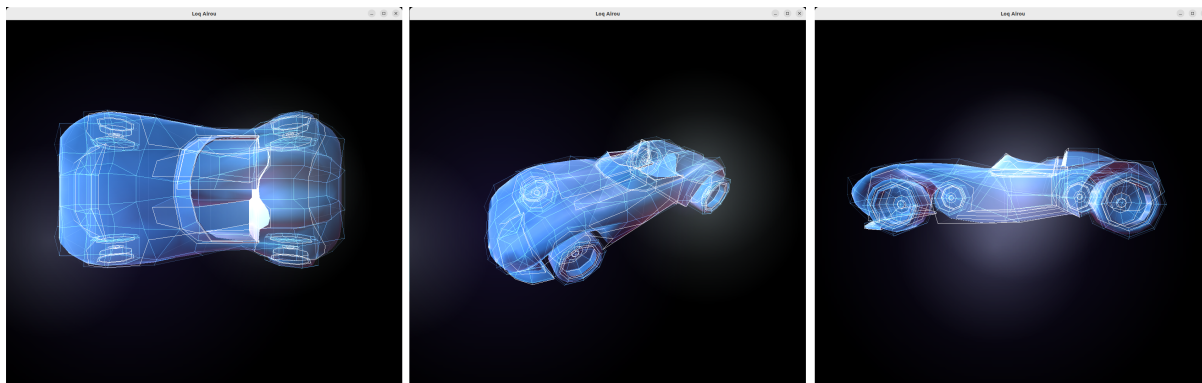
11 THE OHIO State University Pressbooks. Disponível em: <https://ohiostate.pressbooks.pub/graphicshistory/chapter/3-3-general-motors-dac/> Acesso em: abr. de 2024.

Figura 5 - design de carroceria no Computador DAC 1 desenvolvido pela General Motors e International Business Machines Corporation (IBM)



Fonte: The Ohio State University Pressbooks (2024)

Figura 6 - Carroceria simplificada do TVR (Trevor Willkinson) exemplo de geometria do pacote original *Uni-Verse* lançado no ano de 2005



Fonte: Consórcio *Uni-Verse* (2005)

Após o diálogo com Rodrigo Dorta, procuramos o auxílio profissional de Maurício Valerini, administrador de sistemas especialista em Linux, que estava a serviço da seguradora BNP Paribas Cardif¹². Juntos tentamos configurar a rede para que o servidor *verse* pudesse atravessar a porta que escuta por padrão a 4950. O roteador em questão, segundo Maurício, não permite a abertura de portas. Tentamos o estabelecimento de uma comunicação *Secure Shell Linux* (SSH) para ter certeza, nada nos restando a não ser abrir uma solicitação junto a operadora, o que levava até cinco dias úteis para retorno quando foi feita esta tentativa.

De posse destas informações voltei a falar com Rodrigo Dorta, que sugeriu algum tipo de serviço *Virtual Private Network* (VPN) como por exemplo o Hamachi¹³, para tentar driblar a dificuldade de configurar o roteador. Testamos esta possibilidade com relativo sucesso; relembramos que a pasta com os aplicativos *verse*, ao ser descompactada, e um dos aplicativos ser iniciado, um arquivo do tipo *.bin* é gerado (*address.bin*). Este arquivo armazena informações do computador onde o arquivo *.bin* foi gerado, de tal sorte que se simplesmente enviamos a pasta descompactada com o arquivo *address.bin* de um determinado computador – o que pode ser tentador, uma vez que é tão leve este programa em termos de memória de armazenamento – vai ocorrer um paradoxo na interface: se ambos os usuários estiverem trabalhando no mesmo nó de geometria, apenas um manipulador ficará disponível a ambos.

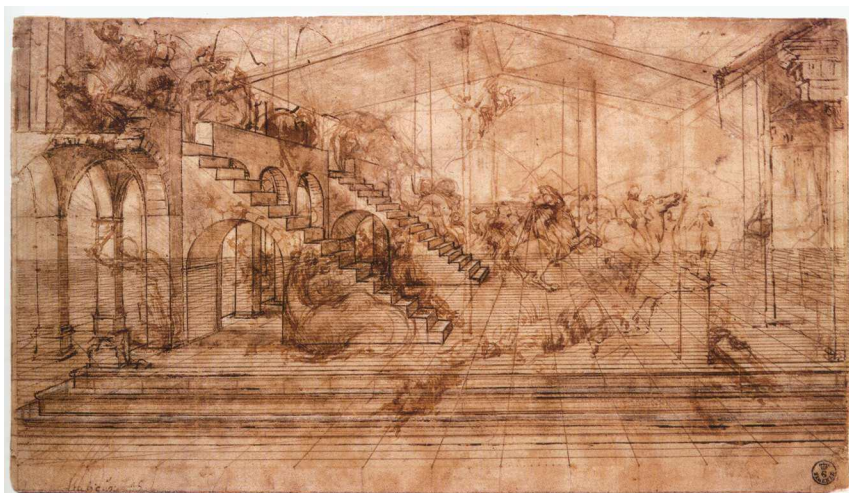
12 BNP Paribas Cardif. Disponível em: <https://bnpparibascardif.com.br/> Acesso em: abr. 2024. Cnpj: 05.711.959/0001-59.

13 HAMACHI Virtual Private Network service. Disponível em: <https://vpn.net/> Acesso em abr. 2024.

3 DESENVOLVIMENTO. SOBRE A PERSPECTIVA CÔNICA EM ESCALA E UM MODELADOR 3D MINIMALISTA.

Os modeladores tridimensionais para computador vêm para endereçar um problema clássico de desenho técnico, que é agilizar o desenho de perspectivas cônicas em escala, tal como nas construções geométricas desenvolvidas por Leonardo da Vinci, e outros mestres da Renascença. Um exemplo de desenho deste artista pode ser visto na Figura 7.

Figura 7 - Estudo de perspectiva para o plano de fundo da Adoração dos Magos. Bico de pena metálico sobre papel. (28.5 x 21.5cm)



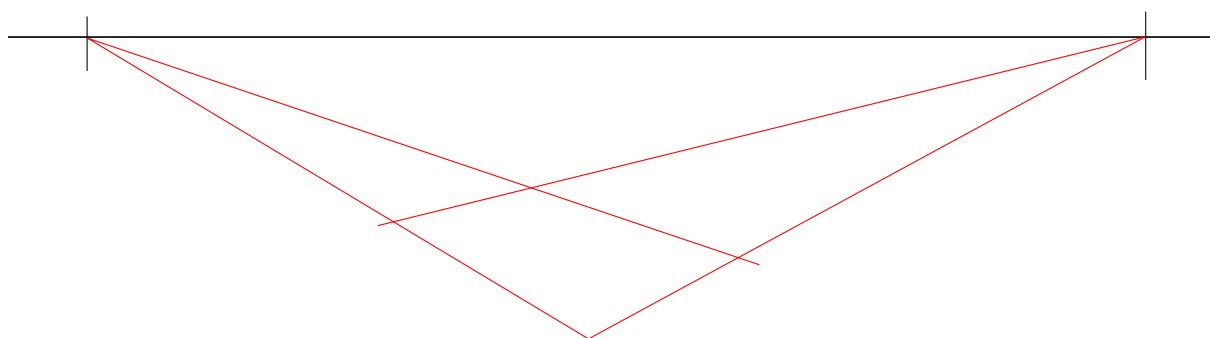
Fonte: Leonardo da Vinci (1481)

Ainda dentro deste enunciado, temos outro problema específico: a dificuldade de se desenharem curvas e, mais adiante, de se construírem superfícies que são compostas por duas curvas ou mais.

Para facilitar o entendimento, vamos recorrer ao uso das Figuras 8, 9 10 e 11 que remontam aos métodos clássicos de desenho de perspectiva cônica que tem início na Renascença e se desenvolvem até os anos cinquenta do século passado.

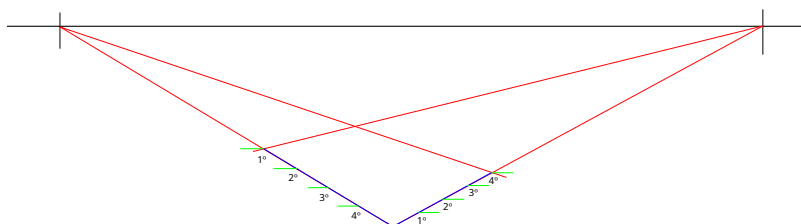
Primeiramente, veremos a sequência para se desenhar uma curva em perspectiva cônica sem nos preocuparmos ainda com a questão da escala.

Figura 8 - depois de definida a linha do horizonte e os pontos de fuga, delimitar o retângulo de interesse;



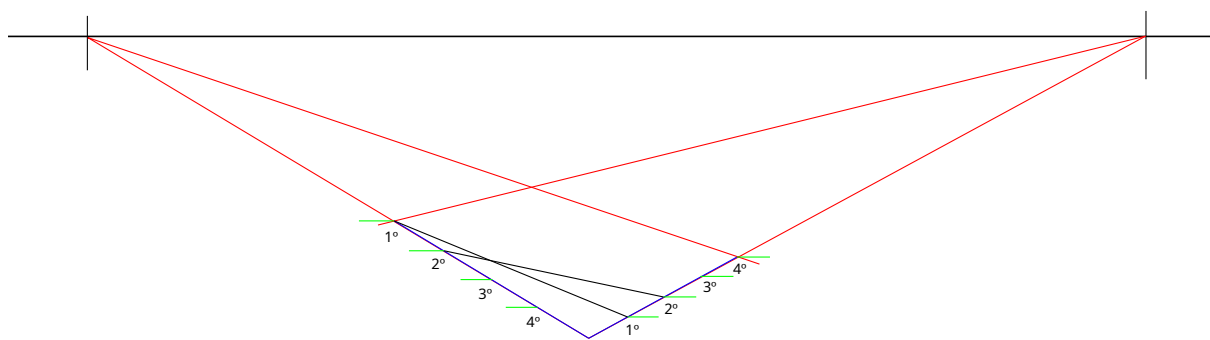
Fonte: elaborada pelo autor (2023)

Figura 9 - definição dos segmentos de interesse, em azul, e divisão pelo mesmo número, em verde;



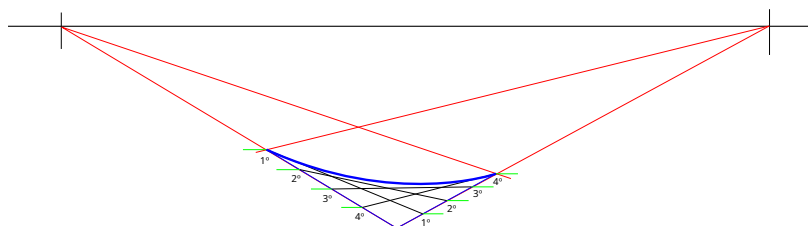
Fonte: elaborada pelo autor (2023)

Figura 10 - ligando os pontos na sequência definida;



Fonte: elaborada pelo autor (2023)

Figura 11 - reforçando a curva a partir da montagem;



Fonte: elaborada pelo autor (2023)

Como podemos observar, um número considerável de linhas auxiliares tem de ser construído para obtermos a curva desejada. Esse número aumenta exponencialmente caso o objetivo seja construir, em escala, uma superfície que interligue cinco curvaturas também em perspectiva cônica. Observemos os desenhos a seguir, que têm como objetivo ilustrar esta última colocação. Os desenhos a seguir foram feitos a partir da sobreposição de folhas de papel manteiga transparentes, sendo interdependentes. Assim optamos por preservar a posição em que se encontram no caderno de desenho em questão, para facilitar a leitura conforme a mudança de páginas deste trecho da dissertação. Após esta etapa prosseguiremos com a análise dos três desenhos. Neste momento é importante salientar que foram feitos três desenhos em sequência que são interdependentes devido à transparência das folhas que foram usadas. A sobreposição, em estando bem próxima, nos permitirá manter uma leitura mais clara dessa sequenciação.

Esses desenhos receberão a numeração de Figuras 12, 13 e 14.

Figura 12 - Primeiro passo para construção de perspectiva cônica em escala de uma superfície com cinco curvaturas.

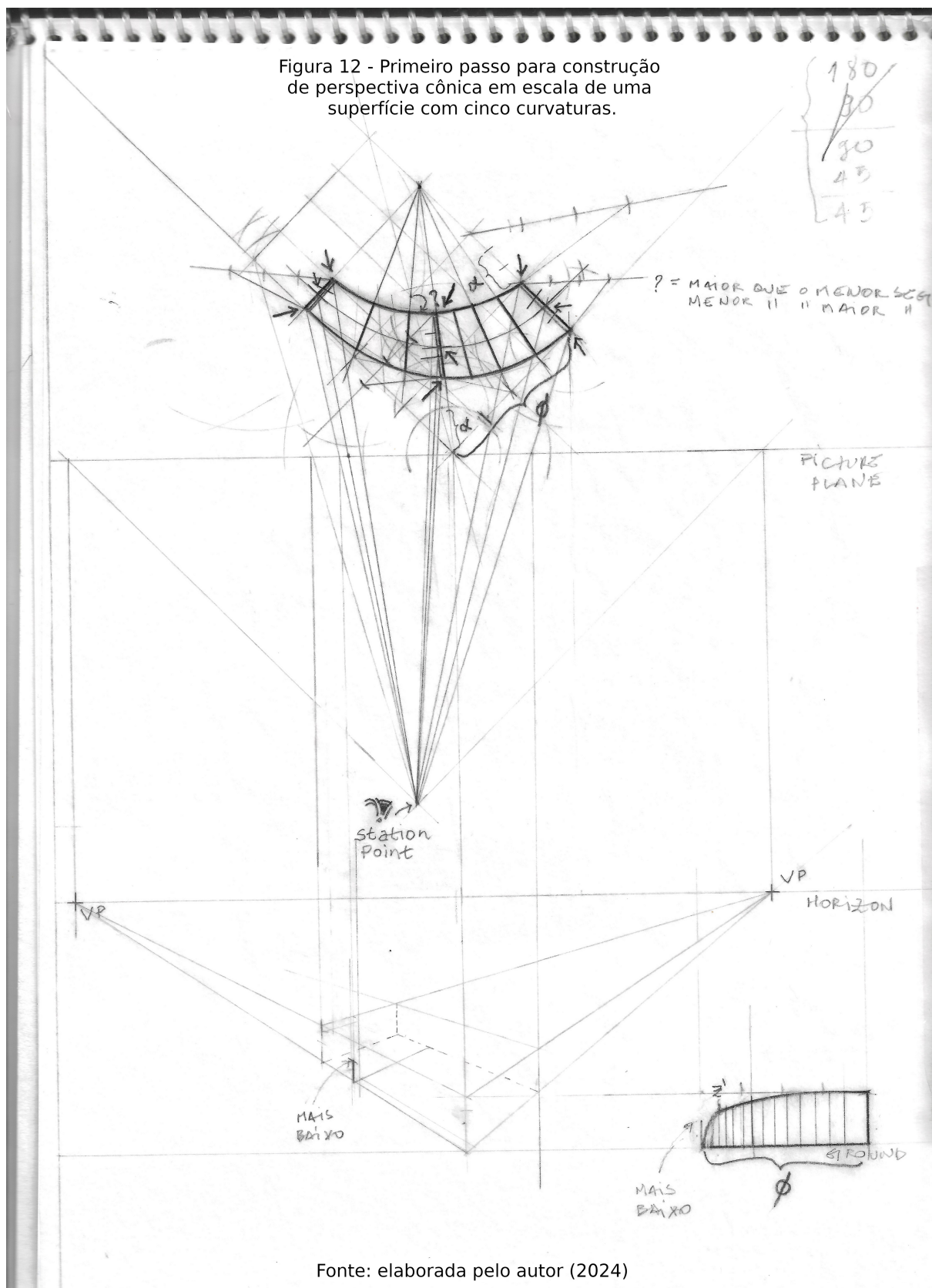
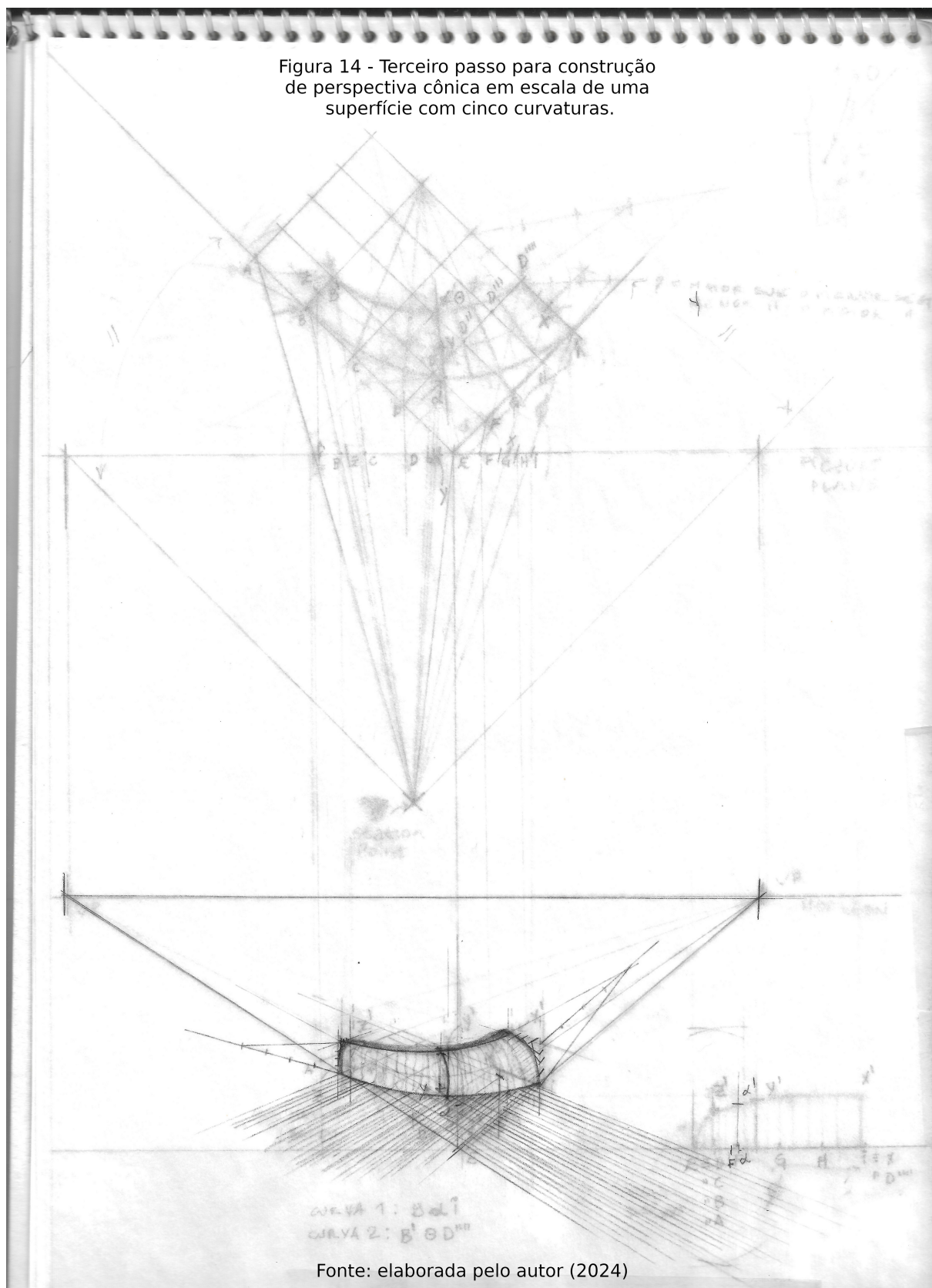


Figura 14 - Terceiro passo para construção de perspectiva cônica em escala de uma superfície com cinco curvaturas.



Fonte: elaborada pelo autor (2024)

Como podemos notar nesta sequência, um grande número de linhas auxiliares teve de ser traçado para se obter o singelo resultado final que interliga as cinco curvaturas principais propostas; tanto as divisões de segmento que acontecem em planos horizontais e verticais como a correção de posição dada pelo rebatimento das linhas que saem do observador e tocam a linha de espelho indo parar no plano de referência abaixo da linha do horizonte.

Além destes passos anteriores, um tratamento que acompanha a superfície com linhas paralelas e uma sombra difusa que responde aos pontos de fuga ajudam a estabelecer maior peso gráfico.

Este tipo de desenho, feito à mão e auxiliado por instrumentos tais como régua paralela, esquadros, compasso e escalímetro, consome muitas horas de produção e, caso um novo ponto de vista tenha de ser obtido, uma grande parte do processo não pode ser reaproveitado, aumentando assim consideravelmente o tempo despendido.

O computador pessoal com a modelagem em três dimensões real a partir de polígonos, que começam a se tornar ubíquos a partir dos anos 1990, vem como uma ferramenta que revoluciona este processo, justamente por possibilitar a criação de novas vistas em tempo real.

O modelador navegador de geometrias Loq Airou resolve de maneira particularmente elegante estas questões. Nele, a construção de vértices, arestas e faces se dá de uma maneira que conta a história da evolução deste tipo de ferramenta desde os anos 1970 até 2015 aproximadamente.

Antes de prosseguirmos com a descrição de Loq Airou, a fim de apontarmos em que medida ele pode servir de exemplo para dinâmicas coletivas em meios computacionais tridimensionais, é importante deixarmos registradas as versões que existem deste programa para futura referência.

A primeira versão de Loq Airou data de 2005. Foi construída a partir de código elaborado por Eskil Steenberg e Emil Brink, naquela época trabalhando como pesquisadores para o KTH. Essa versão usava a biblioteca *Simple Direct Layer*

(SDL).¹⁴ A subdivisão de superfícies e sua variação conforme a distância não ocorriam em tempo real ainda, a pedidos da equipe de desenvolvimento do *Blender*, para que se pudesse verificar como Loq Airou e o *Blender* estavam trabalhando em conjunto. Fomos testemunhas deste pedido na conferência *Blender* 2005, quando Steenberg e Brink apresentaram o andamento da frente de trabalho sueca do Projeto *Uni-Verse*. Para registro, imprimimos esta versão em folhas tamanho A4 com fonte doze. Visto neste formato, o programa tem um total de aproximadamente duzentas e cinquenta páginas frente e verso.

A licença dos programas de Steenberg é a *Berkeley Software Distribution* (BSD).

A licença BSD permite que o desenvolvedor futuramente pare de distribuir seu código gratuitamente permitindo sua incorporação em projetos proprietários e fechados. Foram lançados binários executáveis para Linux, MacOSX e *Windows*.

Esta primeira versão do pacote *Uni-Verse* pode ser obtida no seguinte website:

UNI-VERSE <https://web.archive.org/web/20111121231237/http://www.uni-verse.org/Download-Files-rel-3.73.0.html>
(Acesso em Abril de 2024)

Além de nossos arquivos pessoais há também uma cópia de segurança do código fonte dessa primeira distribuição no serviço do *Google Drive* oferecido pelo Instituto de Artes (é necessário uma conta gratuita *Google* para visualizar, mas os arquivos estão abertos para leitura para qualquer um que possua este endereço):

UNI-VERSE
https://drive.google.com/drive/folders/1agNm1nwh4h3woWGq_cpUc9qysCJvCZsO?usp=sharing
(Acesso em Agosto de 2024)

14 SIMPLE DIRECT LAYER S.D.L. Disponível em: <https://www.libsdl.org/> Acesso em: abr. 2024.

Este mesmo repositório também se encontra como uma pasta compactada com a extensão *.zip* no seguinte endereço:

UNI-VERSE

http://prototipe.com.br/backup_produtos_uni_verse/Z_Back_verse_svn_2_sem_2017.zip
(Acesso em Abril de 2024)

Uma outra versão do Loq Airou com alguns incrementos relevantes começou a ser distribuída no website da empresa de Eskil Steenberg, Quelsolaar¹⁵ (www.quelsolaar.com), por volta de 2008. Nesta versão, os algoritmos de subdivisão e detalhamento estão ligados por padrão. As letras da interface possuem uma melhor resolução, as operações de desenho podem ser feitas com maior facilidade.

Foi lançado um binário executável para *Windows* junto ao código fonte, também em BSD. Esta é a versão em que concentramos nossos estudos e que serviu de principal parâmetro para a escrita desta dissertação. Essa versão pode ser usada com o servidor *verse* do pacote original para Linux, já citado neste trabalho. Para tornar possível a utilização desta versão de Loq Airou para *Windows* no Linux, temos de iniciá-lo com o aplicativo *W.i.n.e.*¹⁶ cuja função é permitir o funcionamento de programas nativos do *Windows* no Linux e no MacOSx.

Essa versão, nosso objeto de estudo, pode ser encontrada no endereço eletrônico a seguir e também em nossos arquivos pessoais:

LOQ AIROU 1 2008

http://prototipe.com.br/steenbergbbackup/verse_apps_chico_2024_rezipped.zip
(Acesso em Abril de 2024)

15 QUELSOLAAR Disponível em: <https://www.quelsolaar.com> Acesso em: set. 2024.

16 W.I.N.E.: acrônimo de "Wine Is Not an Emulator". Disponível em: <https://www.winehq.org/> Acesso em: ago. 2023.

Neste caso não foi possível fazer um backup no *Google Drive* pois o mesmo não permite o carregamento de arquivos com extensão “.exe” (binário executável do sistema operacional *Windows*).

Como este binário de Loq Airou já tem uma idade avançada, ele começou a ser vítima do chamado *bit rot*, que é a degradação de dados que ocorre no meio eletrônico ao longo do tempo. Uma pessoa que se apresenta como Phillip Warren na plataforma de desenvolvimento de aplicativos *GitHub*, tomou a iniciativa de fazer um repositório com o código fonte do programa em questão no ano de 2018. Este repositório pode ser encontrado no seguinte endereço:

VERSE APPS

https://github.com/dannyow/verse_apps

(Acesso em Abril de 2024)

A diferença de como este repositório compila os programas em relação à distribuição do Projeto *Uni-Verse* é a utilização do comando “*tup*” é utilizado ao invés do comando “*make*”.

Já Eskil Steenberg costuma compilar seus aplicativos no programa *Visual Studio*.

Fizemos uma cópia de segurança deste repositório no seguinte endereço:

VERSE APPS

https://github.com/chicortiz/verse_apps

(Acesso em abril de 2024)

A iniciativa de Phillip Warren teve importância também como prova de conceito, pois a partir dela fomos capazes de compilar um binário para Linux em que o algoritmo de subdivisão aparece em tempo real, ao contrário da versão inicial do pacote *Uni-Verse*. No entanto, o programa compilado dessa forma possui erros

típicos de linguagem estilo ISO C89¹⁷ que precisam ser corrigidos (*segfaults*) Esses erros levam a fechamentos abruptos do programa durante sua utilização (“*crashes*”).

O binário compilado para Linux Ubuntu 18.04 LTS pode ser encontrado no seguinte endereço:

LOQ AIROU 1 2008 LINUX UBUNTU
prototipe.com.br/disserta/LoqAirouUbuntuLinux18dot04LTS.zip
(Acesso em Abril de 2024).

Existem outras versões de Loq Airou que merecem ser citadas, principalmente devido a incrementos na interface.

Em 14 de março de 2014 a versão 2 de Loq Airou foi lançada através de link na rede social *Twitter*, atual “X”¹⁸. Essa versão se diferencia das anteriores pois Steenberg escreveu seu próprio código para repor a biblioteca SDL. Para propósitos de indexação, chamemos esta versão de 2014 de “Loq Airou 2a”. Esta versão não foi distribuída com o código fonte.

Na sequência de Loq Airou 2a, veio o que estamos chamando de “Loq Airou 2b”, lançada em 13 de Julho de 2014. Esta versão foi distribuída com o código fonte.

Em 2018 chegou “Loq Airou 2c” e, em 2021, a “Loq Airou 2d”, todas elas aperfeiçoamentos para maior estabilidade. Ambas não foram distribuídas com o código fonte. *A versão Loq Airou 2d possui a conveniência de salvar arquivos de geometria com a extensão Wavefront.obj.* Sendo assim, podemos concluir que o raciocínio de organização mais recente do código fonte para a versão 2 já publicado é a de Loq Airou 2b. Até onde sabemos, as versões “dois” de Loq Airou ainda não funcionam em rede. A conformação do aramado que forma a geometria ainda não incorporou a variação conforme a distância (*level of detail*) que existe nas versões “um”.

Além de cópias de segurança em nossos computadores e discos de memória externos, há também um endereço eletrônico contendo as versões de Loq Airou 2 “b”, “c” e “d”. A versão “a” infelizmente se perdeu de nossos registros.

¹⁷ STEENBERG, Eskil. How do I program in C, 2016. 1 vídeo (2:11:31) Publicado pelo canal Eskil Steenberg Disponível em: <https://www.youtube.com/watch?v=443UNeGrFoM> Acesso em: set. 2021.

¹⁸ Rede Social Twitter, atual X. Disponível em: www.twitter.com Acesso em: abr. 2024.

VERSÕES DE LOQ AIROU 2 “B”, “C” E “D”

<http://prototipe.com.br/disserta/BackupLoqAirou2Versions.zip>

(Acesso em abril de 2024)

Para facilitar o entendimento das versões e repositórios criamos a Tabela 1, na mesma sequência do texto que a precedeu.

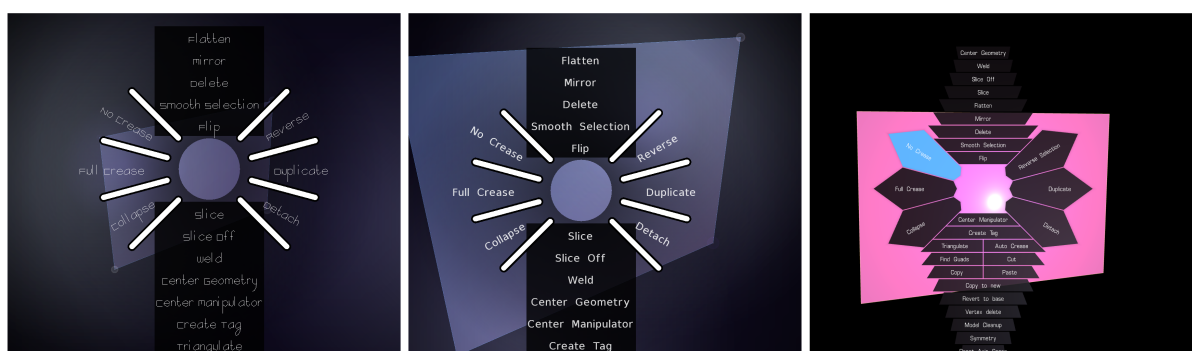
Além disso, a Figura 15 serve como referência gráfica para identificarmos as evoluções de interface relevantes de Loq Airou.

Tabela – 1: Diferentes versões de Loq Airou.

Nº	Pacote	Ano	Observações	Endereços eletrônicos
1	Uni-Verse	2005	Versões para Linux, Windows e MacOSX. Sem algoritmo de subdivisão em tempo real. Menor responsividade da interface, especialmente iniciar uma linha a partir de um vértice já existente. Fonte mais antiga e menos legível. 32 bit	1) https://web.archive.org/web/20111121231237/http://www.uni-verse.org/Download-Files-rel-3.73.0.html 2) https://drive.google.com/drive/folders/1agNm1nwh4h3woWGq_cpUc9qysCJvCZsO?usp=sharing 3) http://prototipe.com.br/backup_produtos_uni_verse/Z_Backverse_svn_2_sem_2017.zip
2	verse_apps distribuído por Quelsolaar.com (Empresa sueca de Steenberg) .	2008	Loq_Airou com algoritmo de subdivisão em tempo real. Maior responsividade da interface. Fontes mais legíveis. Repositório no github iniciativa de Warren. Compilação a partir do comando “tup” 32 bit	1) http://prototipe.com.br/steenbergbbackup/verse_apps_chico_2024_rezipped.zip 2) https://github.com/dannyow/verse_apps 3) https://github.com/chicortiz/verse_apps
3	verse_apps Cópia de segurança iniciada por Phillip Warren no website github.com	2018	Loq_Airou com algoritmo de subdivisão em tempo real. Maior responsividade da interface. Fontes mais legíveis. Versão conceito compilada para Linux Ubuntu 18.04 LTS interrompe o funcionamento bruscamente devido a <i>segfaults</i> . 32 bit	1) http://prototipe.com.br/disserta/LoqAirouUbuntuLinux18dot04LTS.zip
4	Loq Airou 2 ^a , 2b, 2c e 2d Grandes incrementos na interface.	Abril de 2014, Julho de 2014, 2018 e 2021 consecutivamente	a: Não possui código fonte. b: possui código fonte. c e d: sem código fonte. d: possui menu para salvar no formato Wavefront.obj Multiusuário apenas se na máquina do servidor e os outros com versões de Loq Airou 1 32bit	1) http://prototipe.com.br/disserta/BackupLoqAirou2Versions.zip

Fonte: elaborada pelo autor

Figura 15 - Evolução dos menus de Loq Airou. Esta imagem pode ajudar a identificar qual versão está sendo executada em contextos exploratórios, de prototipagem e desenvolvimento. Da esquerda para direita: Loq Airou do pacote *Universe* (2005) Loq Airou distribuído por Quelsolaar.com (2008) e Loq Airou 2d de Quelsolaar.com (2021)



Fonte: elaborada pelo autor a partir de captura de imagem do programa de Eskil Steenberg (2024)

Como podemos perceber a partir da análise das versões de Loq Airou, muitas variáveis estão em jogo. Isso sem nem mencionar que uma segunda implementação do protocolo *verse* foi proposta por Jiri Hnidek¹⁹ Nenhuma versão de Loq Airou que mencionamos é compatível com este segundo protocolo *verse*.

Partindo de nossas experimentações descritas na metodologia optamos, como já citado anteriormente, por nos restringirmos à primeira implementação do protocolo *verse* em ambiente Linux, rodando o Loq Airou de 2008 para Windows a partir do programa *W.I.N.E*. Temos assim: servidor *verse* (aplicativo nativo Linux) e Loq Airou 2008 versão *Windows* sendo executado com programa auxiliar em contexto Linux.

Essa nossa restrição se deve em grande parte devido a nossos recursos modestos e à necessidade de endereçar conceitualmente a parte inicial dessa seção que fala sobre desenho de superfícies curvas em perspectiva cônica em escala.

Caso alguma equipe multidisciplinar se forme inspirada por este registro, não é preciso de maneira alguma se apegar a este contexto. Uma abordagem ainda mais contemporânea poderia ser trabalhar com a implementação dois (2) do protocolo

¹⁹ APLICATIVO VERSE 2. Disponível em: <https://verse.github.io/> Acesso em abr. 2024.

verse e apenas o *Blender* no lugar de de Loq Airou. Essa última sugestão exigiria uma equipe com grande capacidade técnica de programação. A curva de aprendizado do *Blender* também é muito maior do que a de Loq Airou, de fato não faz nem muito sentido compará-los a menos que estejamos em busca de reaver a compatibilidade de outrora. Talvez configurar o *Blender* de maneira que fique mais simples possa resolver este entrave. Uma alternativa intermediária que provavelmente poderia ser explorada por uma equipe multidisciplinar menor e que não tenha uma capacidade de programação tão avançada é o *Blender multi user addon*²⁰. Não sabemos, no entanto, indicar o quanto este *addon* pode ser expandido para funcionar em conjunto com outros programas que não o *Blender*. Por outro lado, seria injusto não mencionar os esforços de evolução de interface em Loq Airou 2, ainda que a implementação do *Level Of Detail* ainda não esteja reintegrada. Precisaríamos de orientação técnica especializada para mensurar quão complexo seria adequá-lo à versão número dois do protocolo verse. A interface de Loq Airou 2 aponta que talvez o esforço de Steenberg esteja indo na direção do servidor verse como serviço com a complexidade mais encapsulada ao invés de programa onde o próprio usuário estrutura sua rede. O que pode evoluir para o tipo de paradoxo que temos hoje - de facilidade de uso versus autonomia e complexidade – o que é facilmente observável no estado deletério de redes sociais virtuais. Esta seria uma visão mais estreita, Steenberg é um desenvolvedor com histórico de generosidade, talvez ele queira trabalhar nos dois *fronts* como mostra seu esforço de construção de uma biblioteca mais enxuta que SDL e a criação de um comando de estilo make, o makemake²¹ de autoria própria, o que o libertaria do *Visual Studio*²².

Nossa abordagem nessa dissertação aponta assim para um ambiente e interface minimalistas, contendo ferramentas voltadas a uma exploração formal refletida, valorizando o aspecto expressivo de cada vértice; uma abordagem portanto muito distante da realidade hiperprodutiva e de altíssimo poder de processamento a que assistimos nos dias atuais. Mas isso não significa a exclusão do diálogo entre estas duas pontas nem diminui a excelência sobre a qual *verse server* e Loq Airou

20 BLENDER MULTI USER ADDON. Disponível em: <https://gitlab.com/slumber/multi-user> Acesso em abr. 2024.

21 COMANDO MAKEMAKE. Disponível em: <https://github.com/quelsolaar/makemake> Acesso em: abr. 2024.

22 PROGRAMA VISUAL STUDIO. Disponível em: <https://visualstudio.microsoft.com/pt-br/> Acesso em: abr. 2024.

foram gestados. O aspecto mais relevante no final das contas é que tipo de relação e interação entre pessoas estes *softwares* podem oferecer.

3.1 SOBRE A ESTRUTURA DO MODELADOR LOQ AIROU

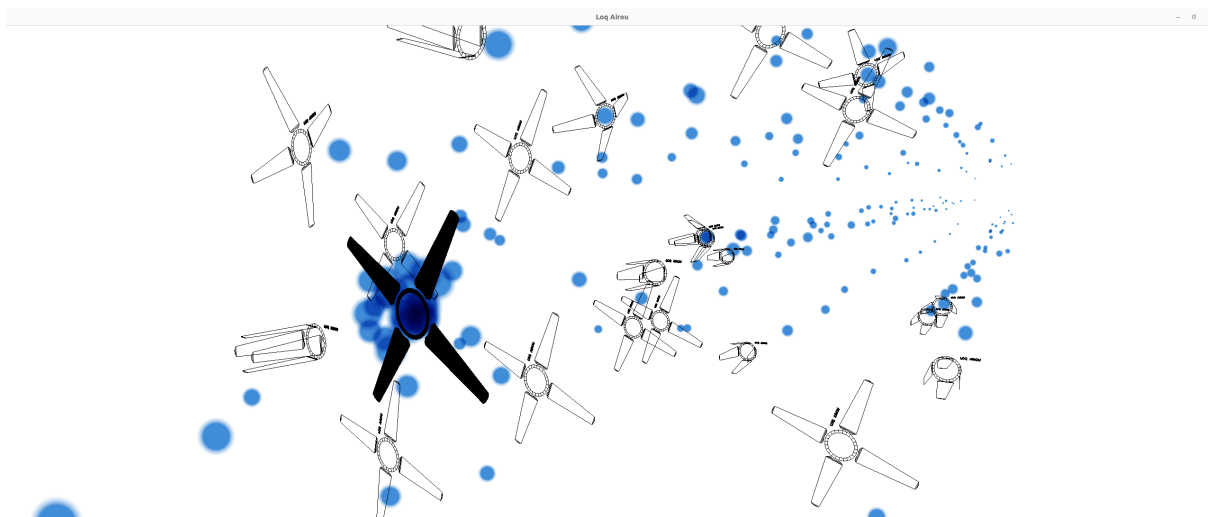
Um aplicativo de modelagem poligonal tridimensional, poderíamos chamá-lo também de *software* ou programa de computador tem de implementar os recursos de desenho que apresentamos anteriormente, a saber: como representar linhas retas e curvas no espaço tridimensional e como utilizar estas linhas para construir superfícies. No meio digital costumamos pensar as curvas como a utilização de diversos segmentos de reta minúsculos que interligados por serem tão pequenos ficam parecendo uma curva (isso até que ocorre em certa medida com as linhas de montagem feitas a mão). No entanto, ao contrário do problema clássico que apresentamos com desenho a lápis, o computador é capaz de calcular as três dimensões e redesenhar com muita agilidade novos pontos de vista, podendo servir inclusive de base para desenhos analógicos. Ao desenhar uma superfície virtual no entanto, uma necessidade permanece: algum tipo de tratamento gráfico precisa ser estabelecido para facilitar o entendimento da conformação da mesma. Hierarquicamente estamos trabalhando então com a sequência: interligar vértices para formar arestas e depois polígonos. Polígonos interligados formarão uma superfície com algum tipo de sombreamento em inglês *shaders*. Não vamos adentrar aqui nas especificidades da discussão grafismo versus sombreamento que Wittkower aborda de maneira tão competente em *Escultura*²³, em especial, quando discute a diferença entre como a estatuária grega era executada em comparação a Michelangelo e o uso do formão dentado por exemplo.

Loq Airou resolve essas operações com certa competência mas existem diversas idiosincrasias a serem observadas que poderiam ser corrigidas dando origem a um aplicativo melhor, ou em alguns casos, se apresentarem de maneira mais didática; já que Loq Airou, em certa medida, nos apresenta também como a modelagem tridimensional no meio computacional foi evoluindo ao longo do tempo. Esta é uma característica preciosa em um ambiente educacional.

23 WITTKOWER, Rudolf *Escultura*. São Paulo: Martins Fontes, 1989.

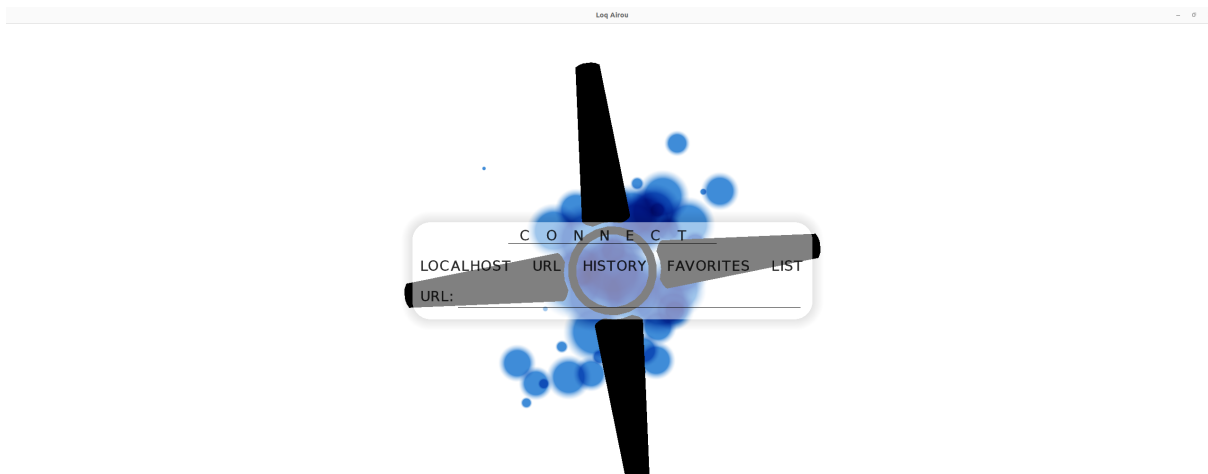
Suponhamos a interação mais simples entre o servidor *verse* e o modelador Loq Airou. Aqui nem estamos pensando ainda em trabalho coletivo. A janela do servidor *verse* tem de ser iniciada para que o aplicativo Loq Airou (outra janela) se conecte a ele. Ao iniciar Loq Airou, algo surpreende de imediato: uma janela de abertura com animação aparece (Figura 16), o que não é muito convencional. Ao clicarmos com o botão esquerdo do mouse, esta animação se encerra, e um painel com uma série de campos de informação aparece, tal como visto na Figura 17. Caso queiramos iniciar uma sessão sozinhos basta clicar no campo onde se pode ler *Localhost*. Loq Airou se conecta então ao servidor local *verse*.

Figura 16 - Loq Airou ao ser iniciado nos apresenta uma animação



Fonte: elaborada pelo autor a partir de captura de imagem do programa de Eskil Steenberg (2024)

Figura 17 - o painel inicial onde aparece (dentre outras) a opção *Localhost* e URL (*Uniform Resource Locator*)



Fonte: elaborada pelo autor a partir de captura de imagem do programa de Eskil Steenberg (2024)

Ao adentrarmos o ambiente de modelagem apenas uma tela preta pode ser vista. Loq Airou não possui uma grade (*grid*) como diversos programas de sua categoria e a representação dos eixos cartesianos é muito sutil, só aparece quando acessamos manipuladores. Permanece no entanto, a ideia de uma interface animada, algo não convencional mas que em certa medida encoraja o usuário na exploração artística do programa, valorizando as operações geométricas em desenho realizadas.

Existe também outra singularidade: a câmera, ou pontos de vista, não possuem representações físicas como objetos dentro do espaço de modelagem tridimensional. Sendo assim, o usuário nunca deixa uma visão em primeira pessoa. Em não havendo nenhuma representação visível do usuário, outros participantes só podem deduzir a posição aproximada um do outro a partir da edição de alguma geometria. A percepção da origem do ambiente ($X = 0$, $Y = 0$ e $Z = 0$) também pode ficar oculta se a condição inicial não for observada.

Um dado importante que não é mencionado no manual de Loq Airou²⁴: assim que adentramos o espaço tridimensional o ponto de vista ou “câmera invisível” está

24 MANUAL DE LOQ AIROU. Disponível em: <http://quelsolaar.com/> Acesso em: abr. 2024.

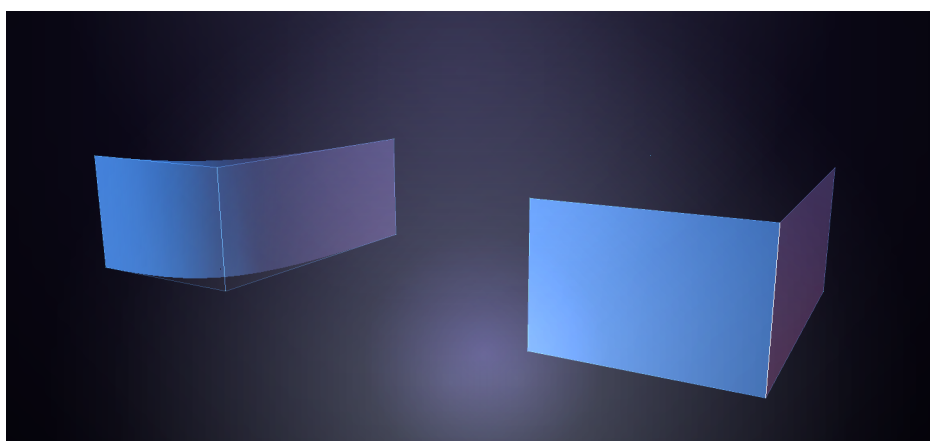
paralelo com o plano formado por um par de eixos cartesianos. Se tomarmos construção convencional de gráficos bidimensionais seriam os eixos X e Y. Dizemos isso pois ao experimentar a interface gráfica de Loq Airou, a convenção para os eixos X, Y e Z também não é representada, então num primeiro olhar, o usuário não tem como deduzir qual é a convenção.

Se enfrentarmos estas dificuldades iniciais, que idealmente deveriam ser corrigidas (e em parte foram, na versão “2d” de Loq Airou) ou constar no manual, e conseguirmos construir um polígono de quatro lados com ângulos retos sem alterar nosso ponto de vista primário, teremos um referencial que nos permitirá um certo respiro, pois poderemos então retornar ao plano paralelo inicial. Isso é bem conveniente para construções geométricas de polígono no espaço tridimensional amparadas por planos cartesianos.

Supondo que a partir deste plano inicial consigamos construir ainda outro plano retangular só que agora perpendicular a nosso plano inicial e conectado com precisão ao mesmo, vamos assim obter uma “esquina”.

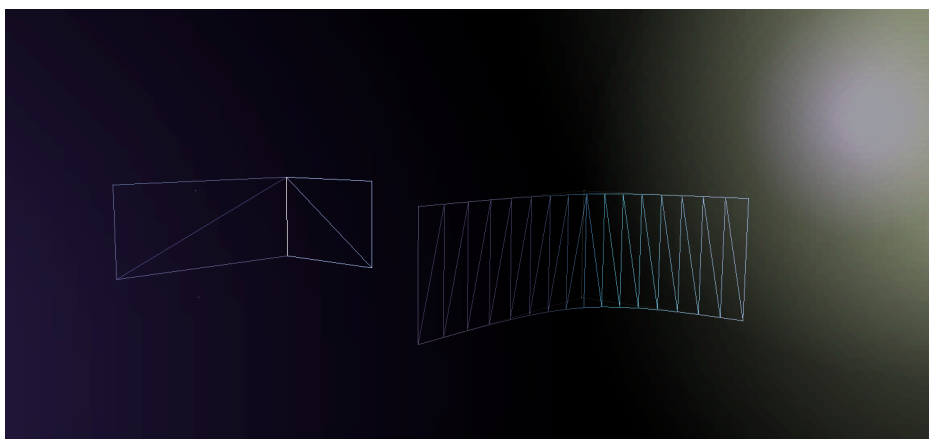
Em Loq Airou, o algoritmo Catmull Clark de subdivisão vem ligado por padrão, o que faz com que a superfície forme uma curva que obedece conformação concordante com nossa esquina. Caso queiramos que a esquina coincida com nossos planos retangulares, basta desenhar sobre a mesma um vinco - em inglês chamado de *crease* - para que a curvatura desapareça, e fiquemos com um canto vivo, também chamado de borda reta que pode ser visto nas Figuras 18 e 19.

Figura 18 - bordas curva e reta em Loq Airou



Fonte: elaborada pelo autor a partir de captura de tela (2024)

Figura 19 - o verso das bordas aparece em modo aramado (*wireframe*) e transparente. Trata-se de uma técnica tradicional de economia de recursos computacionais em programas deste tipo batizada *backface culling*



Fonte: elaborada pelo autor a partir de captura de tela (2024)

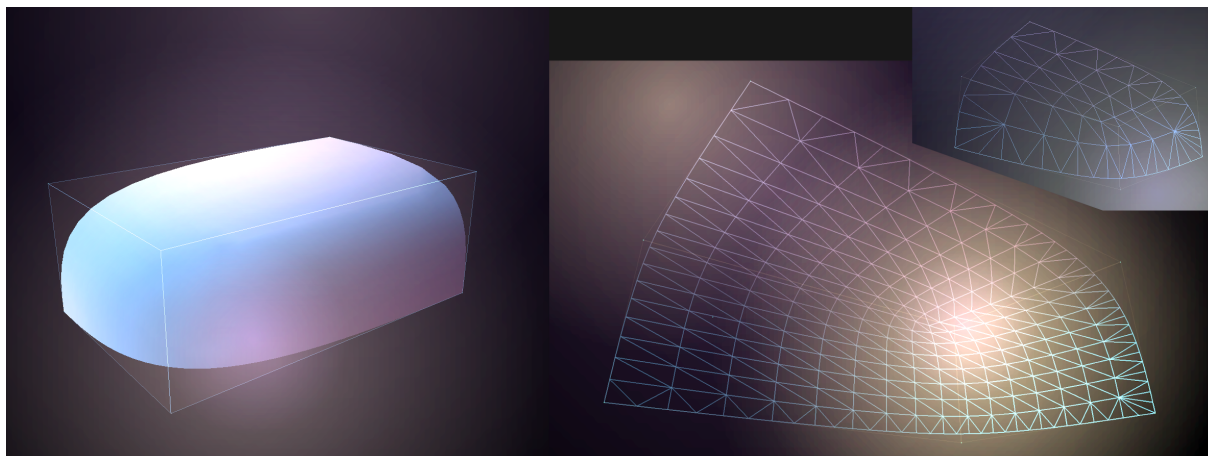
A implementação do algoritmo Catmull Clark varia de programa para programa. No *Blender* por exemplo ela é controlada por um modificador.

No caso de Loq Airou, além de já aparecer automaticamente tem uma camada de tratamento gráfico a mais. Essa implementação melhora o desempenho em tempo real, que subdivide a superfície de acordo com a distância da geometria em relação ao ponto de vista. Quanto mais perto, mais polígonos serão mostrados e detalhes eventuais poderão aparecer; quanto mais longe, menos polígonos aparecerão — pois já não há a necessidade de se mostrarem tantos detalhes. Esta técnica, comum em videogames que utilizam tecnologia 3D, recebe o nome em inglês de *Level Of Detail*. (L.O.D.).

O nível de detalhe, tal como foi implementado em Loq Airou, é um de seus elementos de maior elegância - e que deveria ser aproveitado no desenvolvimento de outros aplicativos na mesma categoria de tal maneira a se tornar genérico.

Com a adição de uma pequena cobertura retangular conectada à borda proposta anteriormente, esse recurso se torna observável, como na Figura 20.

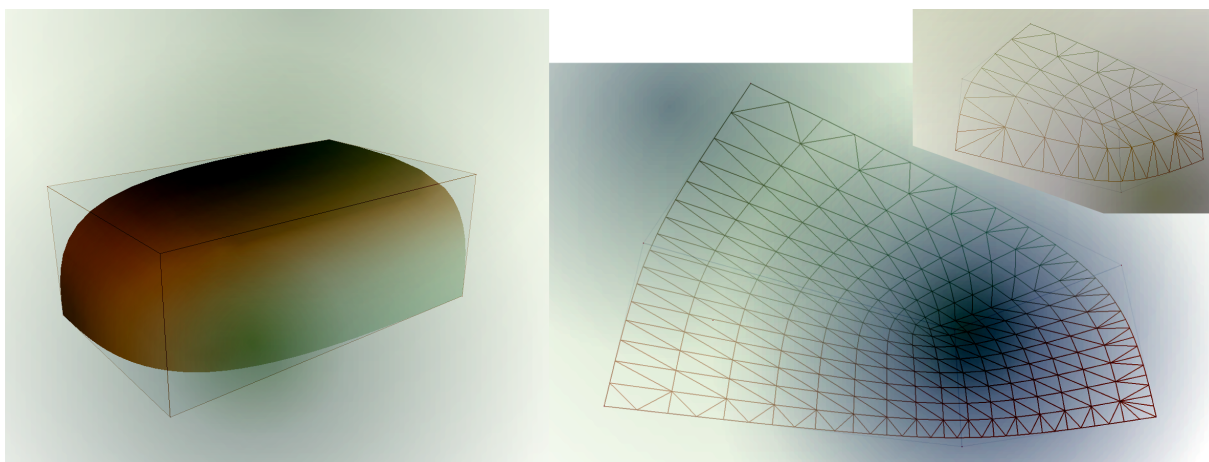
Figura 20 - a casca mais simples possível. À direita no centro verso desta casca visto de perto. À direita no topo verso da casca observada mais de longe



Fonte: elaborada pelo autor (2024)

Uma questão complicada do modelador tridimensional Loq Airou é seu fundo preto. Ele funciona bem para o ambiente digital, mas para o meio impresso é uma desvantagem. Seria uma boa ideia implementar uma opção que permitisse inverter as cores do ambiente de modelagem em um programa deste tipo, assim teríamos um resultado similar ao da Figura 21.

Figura 21 - mesma imagem anterior, porém com as cores invertidas.



Fonte: elaborada pelo autor (2024)

No Apêndice A inserimos uma resenha desenvolvida para a disciplina de Metodologia da Pesquisa Científica em Ciências Humanas, apresentada pelo Professor João Cardoso Palma Filho, para explicar de forma gráfica detalhada o que é proposto no artigo de Steenberg, *Complexity Reduction of Catmull-Clark/Loop Subdivision Surfaces*. Este artigo poderá ser muito útil não apenas no contexto que discutimos aqui, mas na implementação desta ferramenta em outros aplicativos.²⁵

3.2 SUGESTÕES DE ARRANJO PARA TRABALHO EM REDE

Vistas as condições iniciais de utilização individual dos softwares, talvez queiramos partir para a construção de uma rede simples em um ambiente local, ou seja, sem a utilização ainda, de internet para começar a experimentar dinâmicas coletivas neste tipo de ambiente.

Para tornar isso possível é importante não perder de vista a seguinte limitação: o servidor *verse* precisa estar conectado ao nosso aparato de rede com um cabo do tipo de par trançado. Os clientes Loq Airou, no entanto, conseguem se conectar a este servidor *verse* utilizando *WiFi*. Dadas estas variáveis temos dois arranjos ideais mínimos:

1) Três computadores conectados via rede com cabo de par trançado a um *switch* de rede modesto (um dos computadores alocado para o servidor e outros para os clientes);

2) Três computadores e um roteador, sendo que o computador servidor está conectado ao roteador com um cabo de par trançado; e os clientes com o modelador Loq Airou chegam ao servidor via o sinal de *WiFi* do roteador. Em se tratando de uma interação local, com todos os participantes no mesmo espaço físico real, não é necessário que o roteador esteja conectado à internet para que as dinâmicas coletivas entre *verse server* e modeladores Loq Airou se estabeleçam.

Acima descrevemos então dois cenários relativamente simples para interação entre dois usuários.

25 STEENBERG, Eskil. **Complexity Reduction of Catmull-Clark/Loop Subdivision Surfaces**. Estocolmo: Interactive Institute, 2001.

Observação: para que a conexão múltipla com o servidor se estabeleça, é necessário que os usuários conheçam alguns rudimentos sobre endereços IP. No caso de os usuários estarem utilizando Linux, o comando para se achar o endereço IP estático local é `ifconfig`. Em *Windows*, o comando é `ipconfig`. Então, em vez de *Localhost* os usuários terão de preencher o endereço URL com o endereço do servidor. Será algo com o formato similar ao número 192.168.1.1.

Atenção: na hora de digitar o endereço URL nos Loq Airous não recomendamos utilizar a ilha numérica e, sim, os números acima das letras do teclado. Por algum motivo que ainda não identificamos, a ilha numérica certas vezes não funciona.

Também as opções *History*, *Favorites* e *List* do painel superior de Loq Airou, até onde sabemos não funcionam inteiramente. Em suma: mesmo que queiramos estabelecer nosso próprio servidor em uma rede local bem pequena, os procedimentos ainda são bem abstratos para a maior parte das pessoas; a interface deveria ter comentários que ajudassem na interpretação de cada campo para usuários que não têm familiaridade com configurações de ambientes de rede.

Para alguém que tenha decidido enfrentar estas dificuldades iniciais de interface, faz sentido ter em mãos o manual de Loq Airou traduzido para o português e elaborado a partir do original. Recomendamos que se use a versão do ano de 2008 de Loq Airou, que era distribuída no *website* de Eskil Steenberg “Quelsolaar”. O link para o programa em questão se encontra na tabela 1.

O manual em questão pode ser encontrado no endereço eletrônico abaixo, devendo ser considerado apenas como material didático de apoio e não como fazendo parte do escopo principal desta dissertação.

MANUAL DO LOQ AIROU 1 TRADUZIDO

http://prototipe.com.br/disserta/Manual_Loq_2008_Quel_verse_apps.zip

(Acesso em Abril de 2024).

MANUAL DO LOQ AIROU 1 TRADUZIDO

https://drive.google.com/file/d/1YrhWEIkFwakiHFjgp5VLSsl2vU_a3lVW/view?usp=drive_link

(Acesso em Agosto de 2024).

Em um contexto onde cientistas da computação interagem com artistas para desenvolver um programa de modelagem tridimensional, a forma como as geometrias são armazenadas pode ser um ponto de contato e diálogo. Em se tratando do protocolo *verse*, são didáticas por serem legíveis por humanos num formato do tipo *Extensible Markup Language* (.xml). Um exercício interessante nesse sentido pode ser a conversão de geometrias que funcionam bem para serem transmitidas em rede (.xml) para um formato estático tradicional como o *Wavefront obj* (.obj). Não confundir com arquivos do tipo *object file* no contexto de programação de ISO C 89 por exemplo, pois tem a mesma extensão (.obj).

No Apêndice B pode ser encontrado um exemplo de *script* que converte arquivos .xml para arquivos do tipo *Wavefront.obj*.

4 CONCLUSÃO

Ao longo desta dissertação procuramos destacar as dificuldades que surgem quando descrevemos dinâmicas humanas a partir de programas de computador que se apresentam de maneira bastante simples - e por vezes inacabada ou na forma de protótipo - mas que guardam, sob sua casca minimalista, complexidade, elegância intelectual e histórica em relação à mídia da qual descendem. Fizemos essas observações no intuito de que outros pesquisadores, especialmente nas áreas de humanidades, não precisem passar pelos mesmos entraves.

No decorrer de nossa pesquisa lamentamos que iniciativas como o fórum de discussão Low End 3D de uma comunidade que se organizava em torno do artista gráfico David Oreilly e outras similares tenham acabado, pois por ali circulava um público que poderia valorizar um modelador minimalista aberto e livre tal como Loq Airou. Esse ambiente poderia gerar um método criativo em rede. Acreditamos que talvez o que tenha faltado fosse alguma ferramenta comunicacional em vídeo de acesso extremamente facilitado como por exemplo o Jitsi Meet,²⁶ que surgiu depois, onde membros de uma comunidade internacional pudessem se comunicar de maneira mais calorosa.

Temos conhecimento da existência de outros programas proprietários similares, com um número absurdo de ferramentas; mas decidimos enfatizar aqueles que oferecem o máximo de transparência, abertura e liberdade de desenvolvimento com enfoque numa interface minimalista, apostando também na autonomia do usuário. Acreditamos que esta transparência pode ser um atalho dentro de um ambiente de pesquisa universitária público profundamente similar ao de um orçamento participativo aberto e transparente como deve ser sempre o caso na administração pública de um Estado democrático.

Um dos resultados magníficos deste ambiente de pesquisa público brasileiro do qual tivemos o privilégio de participar são os repositórios na *internet* abertos à população. Além de fonte de conhecimento e sabedoria, esses repositórios são um

26 SITE DE REUNIÕES JITSI MEET. Disponível em: <https://meet.jit.si/> Acesso em: ago. 2024.

fator de motivação ao espírito pesquisador e testemunho dos investimentos desta mesma população.

Posto isso, gostaríamos de relatar também diferenças culturais que são de difícil assimilação. Não há dúvidas de que o clima nórdico joga um importantíssimo papel na profusão de softwares desenvolvidos naqueles países, como também em sua linguagem. Essas pessoas que vivem próximas ao Polo Norte tem de enfrentar um clima “bipolar”, por assim dizer: invernos de noites intermináveis e verões com luz sem fim. Isso sem mencionar o mundo político bipolar que pressionava essa mesma região fazendo-a engendrar, por uma questão de vida ou morte, estados de bem-estar social inovadores e bem-sucedidos.

Na microescala que estudamos, em nosso entendimento, plasmou-se um programa de modelagem tridimensional (Loq Airou) para trabalho em grupo muito hermético, de difícil diálogo com as realidades vivenciadas em nosso entorno metodológico. Desmontar este hermetismo foi trabalho longo e penoso, terá sido válido?

Imaginamos que, dentro de um contexto universitário multidisciplinar de pesquisa e desenvolvimento de *software*, que a resposta seja “sim”, mas não deixaria de ser um enorme desafio continuar a desenvolvê-lo. Isso significaria articular artistas, cientistas da computação e designers pesquisadores com paixão e entendimento das dinâmicas possíveis. Uma sugestão de ponto de partida para estes pesquisadores poderia ser: dado um aplicativo que nos permita modelar em conjunto utilizando elementos mínimos como vértices, arestas, faces, superfícies formadas por diversas faces dialogando ou não com um algoritmo para tornar essa construção plástica geometricamente orgânica, como manter a sinergia dentro deste grupo durante o desenvolvimento de um projeto deste tipo? Não seria mais fácil organizar estas mesmas pessoas para trabalharem ao mesmo tempo num enorme bloco de massa cerâmica (nome difícil para “argila”) lá fora, à sombra das árvores num parque, num dia de sol? Certamente mais erótico seria!

Descrever para pesquisadores o escopo do Projeto *Uni-Verse* é tarefa sobrehumana. Mesmo dois de seus subprodutos, *verse server* e Loq Airou, podem ser considerados como sofrendo de gigantismo, tendo em vista o pequeno número

de desenvolvedores (Emil Brink e Eskil Steenberg, principalmente, e depois apenas Steenberg).

E finalmente temos a condição atual, em que os programas analisados pela dissertação são antigos e seus registros começam a desaparecer.

Depois de longa análise, foi necessário decidir quais partes do programa seriam priorizadas para arquivamento; também analisar seu apelo visual e considerar de que forma a modelagem poligonal a partir de uma interface minimalista pode ser intuitiva.

Ao mesmo tempo, a seguinte questão despertava dúvidas: em que medida esse esforço pode dialogar com a realidade brasileira ou de países ditos “em desenvolvimento”? Que tipo de apelo o programa pode ter no clima subtropical úmido muito mais ameno de São Paulo? Clima este, aliás, que também corre o risco de desaparecer em função do gigantismo da metrópole e do antropoceno. Metrópole essa, diga-se de passagem, de clima social nada ameno, onde a diferença brutal de renda abre um abismo social de grandes proporções.

Diante destes problemas e dificuldades profundos, algumas pessoas ainda insistem em descer na estação Barra Funda e atravessar a rua para trocar complexidades no Instituto de Artes. Ainda pegam o ônibus no terminal Butantã para depois aproveitar o peripatos com direito a Mata Atlântica na Cidade Universitária.

E este é o entorno no qual, por enquanto, procuro me ater, sob o risco de me tornar ainda mais diletante, já que é este o lugar onde encontrei respostas que me levaram a novas perguntas ainda mais instigantes.

REFERÊNCIAS

- AMBIENTE DE DESENVOLVIMENTO INTEGRADO (IDE) THONNY. **Thonny**. Tartu: [s.n., 2024]. Disponível em: <https://thonny.org/>. Acesso em: 08 maio 2024.
- APLICAÇÕES VERSE DE 2008: **Verse apps**. São Francisco: [s.n., 2016]. Disponível em: https://github.com/dannyow/verse_apps Acesso em: 08 maio 2024.
- APLICAÇÕES VERSE DE 2008 CÓPIA: **Verse apps**. São Francisco: [s.n., 2016]. Disponível em: https://github.com/chicortiz/verse_apps. Acesso em: 08 maio 2024.
- APLICATIVO VERSE 2: **Verse**. Liberec: [s.n., 2018]. Disponível em: <https://verse.github.io/>. Acesso em 14 de abr. 2024.
- BLENDER MULTI USER ADDON. **Multi-user**. São Francisco: [s.n., 2017]. Disponível em: <https://gitlab.com/slumber/multi-user>. Acesso em: 08 maio 2024.
- BNP PARIBAS CARDIF. **Quem-somos**. Nova York: [s.n., 2024]. Disponível em: <https://bnpparibascardif.com.br/quem-somos>. Acesso em: 08 maio 2024.
- BOSI, Alfredo. Arte e Conhecimento em Leonardo da Vinci: São Paulo, Edusp, 2017
- CARROCERIA DO TVR (TREVOR WILLKINSON) GEOMETRIA DO PACOTE ORIGINAL UNI-VERSE DE 2005. **Downloads** São Francisco: [s.n., 2024]. Disponível em: <https://web.archive.org/web/20111121231237/http://www.uni-verse.org/Download-Files-rel-3.73.0.html> Acesso em: 20 set. 2024.
- CATMULL, E; CLARK, J. Recursevely generated B-spline surfaces on arbitrary topological meshes. **IPC Bussiness Press computer-aided design**, [S.l.], v. 10 p. 350-355, nov. 1978.
- COMANDO TUP. **Tup**. São Francisco: [s.n., 2024]. Disponível em: <https://gittup.org/tup/>. Acesso em: 8 maio 2024.
- CONSTANTINO, Maria. **Leonardo**. Nova Iorque, Crescent Books. 1993.
- DE CARVALHO, Francisco Ortiz. Synchrony, 2014. 1 vídeo (0:02:17) Publicado pelo canal chicortiz. Disponível em: <https://youtu.be/yMvjg-r9XVM> Acesso em 9 out. 2024.
- DE CARVALHO, Francisco Ortiz. Synchrony: cópia 1, 2014. 1 vídeo (0:02:17). Disponível em: prototipe.com.br/backup_produtos_uni_verse/Synchrony.mp4 Acesso em 9 out. 2024.
- DE CARVALHO, Francisco Ortiz. Synchrony: cópia 2, 2014. 1 vídeo (0:02:17). Disponível em: https://drive.google.com/file/d/1uE0dhn9a7X-3zn8g3tcqcByRMvHC2zYg/view?usp=drive_link Acesso em 9 out. 2024.

DOMAIN NAME SYSTEM. **Home**. California: [s.n., 2024] Disponível em: <http://mo00.com/>. Acesso em: 23 set. 2024.

FUNDAÇÃO BLENDER. **Fórum**. Amsterdam: [s.n., 2017]. Disponível em: <https://www.blender.org/forum/viewtopic.php?t=2952&p=16829> Acesso em: 01 jul. 2017.

GAMING LOVE. **Forums Eskill's Tools**. Londres: [s.n., 2011]. Disponível em: <https://web.archive.org/web/20120508075406/http://www.gaminglove.net/>. Acesso em: 08 maio 2024.

GENERAL MOTORS. Carroceria no computador dac 1 desenvolvido pela General Motors e International Business Machines Corporation (IBM). *In: Computer Graphics and Computer Animation: A Retrospective Overview*. Columbus: [s.n., 2024]. Disponível em: <https://ohiostate.pressbooks.pub/graphicshistory/chapter/3-3-general-motors-dac/> Acesso em: 08 mai. 2024.

HAMACHI Virtual Private Network service. Budapeste: [s.n., 2024] Disponível em: <https://vpn.net/> Acesso em 02 abr. de 2024.

HNIDEK, Jiri. Network **Protocols for Applications of Shared Virtual Reality**. Liberec: Technical University of Liberec, 2011.

JITSI MEET. Site de reuniões Jitsi Meet. Disponível em <https://meet.jit.si/> Acesso em ago. 2024.

LOQ AIROU QUELSOLAAR VERSÃO 2008. Estocolmo: [s.n., 2024]. Disponível em: http://prototipe.com.br/steenbergbbackup/verse_apps_chico_2024_reziped.zip. Acesso em: 08 maio 2024.

LOQ AIROU 1 2008 LINUX UBUNTU. São Paulo: [s.n., 2024]. Disponível em: prototipe.com.br/disserta/LoqAirouUbuntuLinux18dot04LTS.zip Acesso em: 02 abr. 2024.

LOQ AIROU 2 VERSÕES B, C e D. Estocolmo: [s.n. 2024]. Disponível em: <http://prototipe.com.br/disserta/BackupLoqAirou2Versions.zip> Acesso em: 02 abr. 2024.

LOWEND3D. **Forum**. Kilkenny: [s.n., 2011]. Disponível em: <https://web.archive.org/web/20110202173944/https://www.lowend3d.com/>. Acesso em: 08 maio 2024.

MAKEMAKE. **Makemake**. Estocolmo: [s.n., 2024]. Disponível em: <https://github.com/quelsolaar/makemake>. Acesso em: 08 maio 2024.

MANUAL DO LOQ AIROU. Google Drive. São Paulo: [s.n., 2023]. Disponível em: https://drive.google.com/file/d/1YrhWEIkFwakiHFjgp5VLSsl2vU_a3lVW/view?usp=drive_link Acesso em: 08 maio 2024. Traduzido e corrigido para o português.

OPERADORA Vivo Telefônica S/A Disponível em: <https://vivo.com.br> Acesso em abr. 2024 Cadastro Nacional de Pessoa Jurídica (CNPJ): 02.558.157/0001-62.

MANUAL DO LOQ AIROU. Prototipe. São Paulo: [s.n., 2023]. Disponível em: http://prototipe.com.br/disserta/Manual_Loq_2008_Quel_verse_apps.zip. Acesso em: 08 maio 2024. Traduzido e corrigido para o português.

MANUAL DE LOQ AIROU. **Quelsolaar**. Estocolmo: [s.n.,2024]. Disponível em: <http://quelsolaar.com/>. Acesso em: 08 maio 2024.

NASCENTES, Antenor. **Dicionário Etimológico da Língua Portuguesa**. Rio de Janeiro, Oficina da Encadernadora, 1932.

PACOTE UNI-VERSE: cópia 1. Estocolmo [s.n., 2005]. Disponível em: https://drive.google.com/drive/folders/1agNm1nwh4h3woWGq_cpUc9qysCJvCZsO Acesso em: 10 ago. 2024.

PACOTE UNI-VERSE: cópia 2. Estocolmo [s.n., 2005]. Disponível em: http://prototipe.com.br/backup_produtos_uni_verse/Z_Back_verse_svn_2_sem_2017.zip. Acesso em: 10 ago. 2024.

PACOTE UNI-VERSE. **Downloads**. [S.l.: s.n., 2005]. Disponível em: <https://web.archive.org/web/20111121231237/http://www.uni-verse.org/Download-Files-rel-3.73.0.html> Acesso em: 10 ago. 2024.

REDE SOCIAL TWITTER, ATUAL X. Califórnia [s.n.,2024]. Disponível em: <https://www.twitter.com>. Acesso em 14 abr. 2024.

SCRIPT CONVERSÃO .XML PARA WAVEFRONT.OBJ: primeiro. Prototipe. São Paulo: [s.n., 2023] Disponível em: prototipe.com.br/disserta/vml2obj.py Acesso em: 10 ago. 2024.

SCRIPT CONVERSÃO .XML PARA WAVEFRONT.OBJ: primeiro. Google Drive. São Paulo: [s.n., 2023] Disponível em: prototipe.com.br/disserta/vml2obj.py Acesso em: 10 ago. 2024.

SCRIPT CONVERSÃO XML PARA WAVEFRONT.OBJ: segundo. Prototipe. São Paulo: [s.n.,2023]. Disponível em: prototipe.com.br/disserta/vml2obj32b.py Acesso em: 10 ago. 2024.

SCRIPT CONVERSÃO XML PARA WAVEFRONT.OBJ: segundo cópia. Google Drive. São Paulo: [s.n.,2023]. Disponível em: https://drive.google.com/file/d/1_Ac6Z5zVFuvrv4veKmzRFLhBnYMKksAt/view?usp=drive_link Acesso em: 10 ago. 2024.

SHAKESPEARE, William. Hamlet. **Obra completa em três volumes**: volume 1. Rio de Janeiro. Companhia José Aguilar Editora, 1969. Obra original de 1623.

SIMPLE DIRECT LAYER. **About.** Davies: [s.n.,2024]. Disponível em: <https://www.libsdl.org/index.php>. Acesso em: 08 maio 2024.

STEENBERG, Eskil. Complexity Reduction of Catmull-Clark/Loop Subdivision Surfaces. **The Eurographics Association Short Presentations**, Estocolmo, p. 1-6, Set. 2001.

STEENBERG, Eskil. How do I program in C, 2016. 1 vídeo (2:11:31) Publicado pelo canal Eskil Steenberg. Disponível em: <https://www.youtube.com/watch?v=443UNeGrFoM> Acesso em: 9 out. 2024.

STEENBERG, E. How one word broke c. *In: QUELSOLAAR, Blog News Quelsolaar*, Estocolmo, 16 de abr. 2020. Disponível em: <http://web.archive.org/web/20200316035336/https://news.quelsolaar.com/2020/03/16/how-one-word-broke-c/>. Acesso em: 11 de maio 2024.

O'REILLY, David. I'll go crazy if i don't go crazy tonight, 2010. 1 vídeo (0:04:13) Publicado pelo canal U2. Disponível em: https://www.youtube.com/watch?v=4q_AHAMVQ9c Acesso em: 9 out. 2024.

U2. I'll go crazy if i don't go crazy tonight. **Lyrics.** Dublin: [s.n.,2010]. Disponível em: <https://www.u2.com/music/lyrics/184> Acesso em: 8 maio 2024.

UNCREATED ESCRITÓRIO DE ARQUITETURA. **Home.** São Paulo [s.n., 2024] Disponível em: <https://uncreated.net/>. Acesso em: 26 ago. 2024.

VISUAL STUDIO. **Microsoft.** Redmond: [s.n., 2024]. Disponível em: <https://visualstudio.microsoft.com/pt-br/>. Acesso em: 8 maio 2024.

WEBSITE PROJETO UNI-VERSE EXTENSÃO 3D STUDIO MAX. **Downloads.** São Francisco: [s.n., 2024]. Disponível em: <https://web.archive.org/web/20111121231237/http://www.uni-verse.org/Download-Files-rel-3.73.0.html> Acesso em: 20 set. 2024.

W.I.N.E. **About.** [S.l.: s.n., 2024]. Disponível em: <https://www.winehq.org/about>. Acesso em: 8 maio 2024.

WITTKOWER, Rudolf Escultura. São Paulo: Martins Fontes, 1989.

APÊNDICE A: RESENHA QUE TRADUZ GRAFICAMENTE ARTIGO DE STEENBERG SOBRE SUBDIVISÃO DE SUPERFÍCIES

Aqui encontra-se transcrita resenha desenvolvida na disciplina Metodologia da Pesquisa Científica em Ciências Humanas, apresentada pelo Prof. Dr. João Cardoso Palma Filho no primeiro Semestre de 2020. Ela serve de guia de leitura para o artigo STEENBERG, Eskil. Complexity Reduction of Catmull-Clark/Loop Subdivision Surfaces. Estocolmo: Interactive Institute, 2001.

“A curvatura de uma superfície tridimensional no modelador minimalista Loq Airou”

Nesta resenha pretendemos apresentar o resumo de uma bibliografia central em nosso projeto de pesquisa para o processo seletivo no Mestrado do Instituto de Artes da Unesp no ano de 2020/21, na linha de pesquisa “Processos e Procedimentos Artísticos”.

A bibliografia em questão é o artigo “Complexity Reduction of Catmull-Clark/Loop Subdivision Surfaces” escrito em 2001 por Eskil Steenberg para o Instituto Interativo de Estocolmo.

Além de resumir este artigo, pretendemos mostrar como a ideia central dessa bibliografia foi incorporada também por Steenberg, a um programa de computador para modelagem de superfícies tridimensionais chamado “Loq Airou”. Acreditamos que este programa pode ser de grande interesse para a comunidade do Instituto de Artes da Unesp, pois trata-se de uma ferramenta que possibilita aos usuários se exprimirem a partir das regras de perspectiva, criando assim, esboços de projetos tridimensionais.

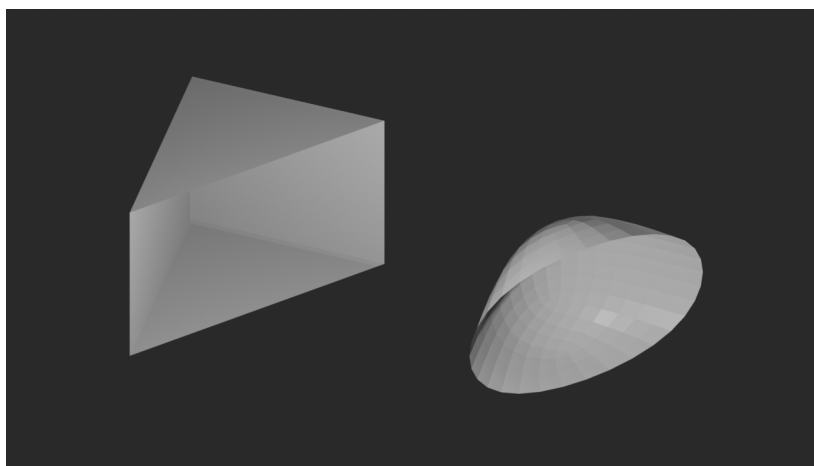
Este programa possui uma outra característica relevante: permite que os usuários possam trabalhar concomitantemente no mesmo espaço tridimensional virtual utilizando a rede (local ou internet) para este fim. Isto é possível, pois o Modelador “Loq Airou” opera como cliente de uma outra ferramenta: o servidor *verse*, seguindo para isso uma especificação de protocolo de comunicação em rede.

Os programas que mencionamos anteriormente, possuem o código fonte aberto, ou seja; uma pessoa que pretenda estudar como estes programas foram produzidos considerando os textos de código necessários para sua formação, tem total acesso a esta camada de informações podendo reutilizá-las e modificá-las conforme seu próprio interesse. Este fator combina com a natureza de abertura e transparência ao público presente em repositórios de produção científica de universidades como a Unesp a USP e a Unicamp.

Um primeiro problema que se coloca é a tradução do título *Complexity Reduction of Catmull-Clark/Loop Subdivision Surfaces*. Uma tradução possível seria “Redução da complexidade de superfícies subdivididas pelo algoritmo Catmull/Clark”. Na sequência precisamos esclarecer para o leitor de maneira breve o que é o algoritmo de subdivisão de Edwin Catmull e Jim Clark. Trata-se de uma solução de desenho: a partir da subdivisão de segmentos de uma determinada forma geométrica obtêm-se uma outra forma arredondada correspondente (mais orgânica). A Figura 22 procura ilustrar este conceito:

Figura 22 - Superfície à esquerda: sem o processo de subdivisão Catmull/Clark.

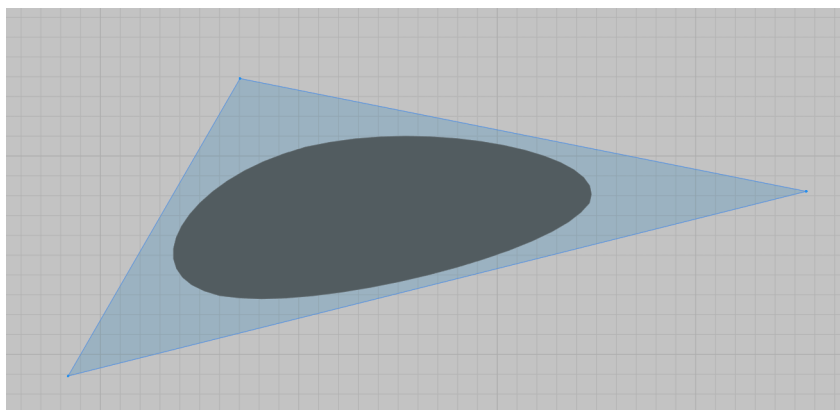
Superfície à direita: mesma superfície porém com o processo de subdivisão Catmull/Clark



Fonte: elaborada pelo autor (2021)

A figura anterior ilustra como o algoritmo Catmull/Clark é tradicionalmente aplicado a uma superfície. Para facilitar o entendimento acrescentamos a Figura 23, que ilustra uma das formas mais básicas que podemos ter num ambiente de modelagem 3D: o triângulo, também chamado no contexto de computação gráfica de “face” ou “polígono”.

Figura 23 - Em azul o triângulo que origina a forma ovoide do tipo Catmull/Clark, em cinza escuro, no miolo



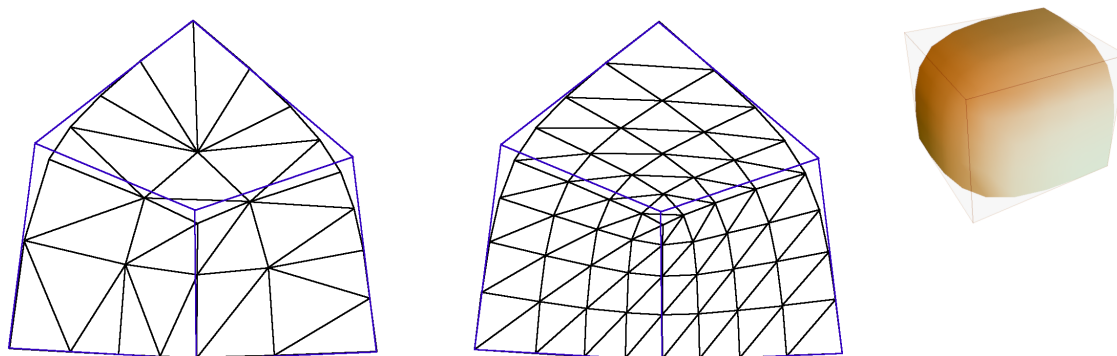
Fonte: elaborada pelo autor (2021)

A abordagem Catmull/Clark é eficiente para a construção de formas orgânicas, pois a partir de formas mais simples podemos alterar formas mais complexas no interior destas formas mais simples. Em outras palavras: com um número menor de pontos no exterior podemos alterar proporcionalmente um número maior de pontos ancorados no interior.

O artigo de Steenberg, no entanto, propõe uma outra camada de tratamento que pode ser assim resumida: conforme o modelo tridimensional se aproxima da câmera que o observa, ele se apresenta com mais detalhes — ou em outras palavras com mais polígonos. Conforme a câmera se afasta deste modelo ele é desenhado com menos detalhes, ou seja, menos polígonos. Esta transição de detalhes no entanto, não ocorre de maneira linear. A solução de Steenberg é capaz de avaliar em quais trechos da superfície a subdivisão é mais necessária, redesenhando então a forma seguindo o parâmetro estabelecido.

Na Figura 24 procuramos ilustrar este último trecho, a partir de uma captura de tela do programa “Loq Airou” que implementa esta solução.

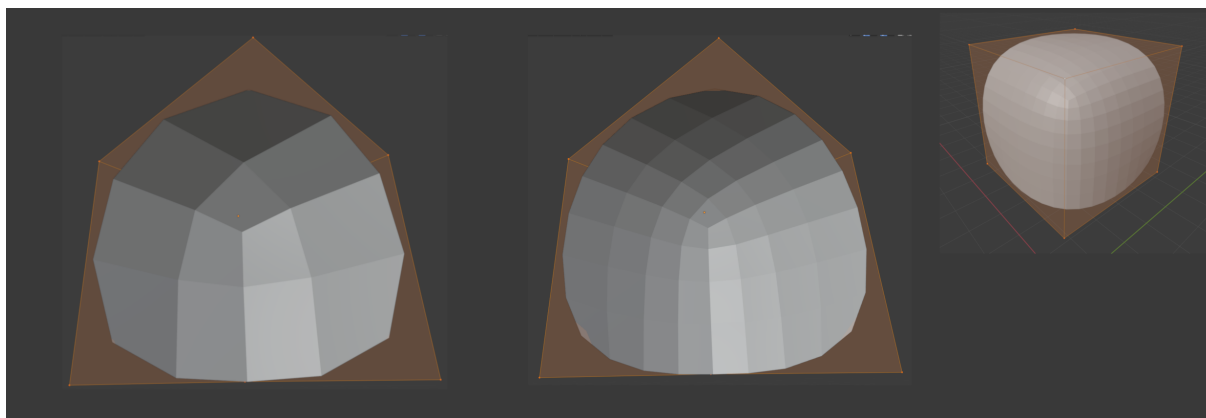
Figura 24 - Em azul: superfície que controla a forma em casca no interior. À esquerda: superfície com menor complexidade (mais distante da câmera ou ponto de vista). À direita no centro: superfície com maior complexidade (mais próxima da câmera ou ponto de vista). À direita em laranja: mesma casca vista por outro ângulo com tratamento gráfico na superfície. As imagens foram retocadas para melhor visualização



Fonte: elaborada pelo autor (2021)

Para efeito de comparação vejamos como esta complexidade é normalmente tratada observando a Figura 25:

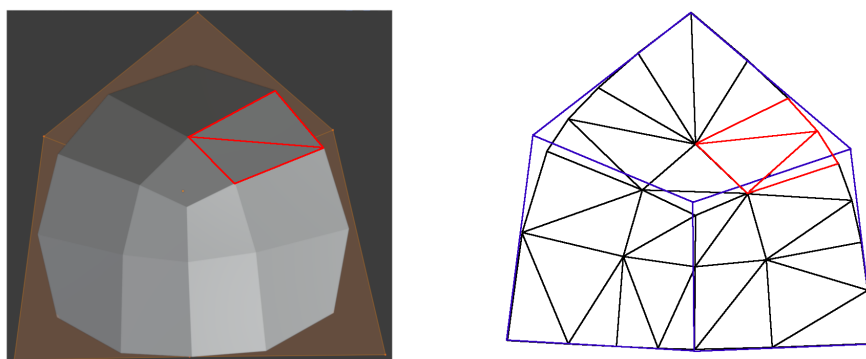
Figura 25 - Mesma superfície que a figura anterior mostrada no Programa “Blender 3D” (www.blender.org)



Fonte: elaborada pelo autor (2021)

É na triangulação da forma interior que melhor podemos observar a solução de Steenberg funcionando, conforme Figura 26 demonstra:

Figura 26 - Comparação entre a solução tradicional para uma casca do tipo Catmull/Clark à esquerda e a solução de Steenberg, à direita. O trecho ressaltado em vermelho mostra como se dá a triangulação em ambos os casos



Fonte: elaborada pelo autor (2021)

Um último apontamento que não possui tanta relação com o que apresentamos anteriormente: o modelador de superfícies “Loq Airou”, trabalha com polígonos de três ou quatro lados, já o programa *Blender*, é capaz de trabalhar com polígonos de mais de quatro lados ocultando a triangulação no tratamento da imagem.

Crítica: para um conteúdo de tanta complexidade o artigo de Steenberg é muito compacto (seis páginas no total contando a bibliografia). Isso torna o conteúdo muito denso e de muito difícil absorção para um leitor que seja leigo no assunto.

A figura referência do segmento sobre tesselação fica em uma coluna diferente de onde a explicação é dada, dificultando a sua leitura em meios eletrônicos; o leitor é obrigado a ficar comparando o canto inferior esquerdo da página com o canto superior direito para decifrar o que parece ser o núcleo da solução que estrutura o trecho de código.

Ainda assim, a sutileza da solução técnica faz valer a pena a leitura do artigo.

APÊNDICE B : SCRIPTS CONVERSORES DE .XML PARA WAVEFRONT .OBJ

Abaixo apresentamos dois *scripts*; textos que podem ser executados como programas. Eles foram desenvolvidos e aperfeiçoados de 2012 até 2024. O objetivo destes *scripts* é converter geometrias armazenadas em formato *Extensible Markup Language* (.xml) para formato *Wavefront.obj* (.obj).

O formato .xml é prático para fazer registros do que está ocorrendo nas dinâmicas de comunicação em rede durante a criação de uma forma 3D no Loq Airou por exemplo. No entanto, não é comum existirem importadores de .xml para programas tridimensionais; sendo assim o arquivo *Wavefront.obj* se apresenta como uma alternativa comum e didática, pois assim como o .xml é *human readable*, ou seja; legível por humanos ao contrário de outros códigos de máquina criptografados. A “desvantagem” relativa do .xml e do *Wavefront.obj* em relação a arquivos criptografados, é que esses últimos podem ser mais compactos e portanto muito mais rápidos de se processar pelo computador do que os primeiros.

O primeiro *script* apresenta uma solução parcial, porém mais simples, e pode ser útil como introdução. Ambos os *scripts* foram escritos em uma linguagem de programação chamada *Python* e foi utilizada a versão 3 (três) desta linguagem.

Ambos os *scripts* podem ser executados com o comando `python3` no terminal do Linux ou em uma *integrated development environment* (IDE), ambiente integrada de desenvolvimento, tal como por exemplo a Thonny IDE²⁷.

No caso do primeiro *script*, o mais simples, pode ser que os números entre colchetes tenham de mudar, dependendo de como estiver a estrutura de nós do arquivo .xml. Ao copiar e colar os exemplos de *script* apresentados neste anexo, pode ser que a endentação dos mesmos se perca. Nos links abaixo podem ser encontrados os *scripts* salvos com a extensão .py, para evitar que este erro ocorra em um ambiente de testes:

²⁷ integrated development environment (IDE) Disponível em: <https://thonny.org/> Acesso em: abr. 2024.

Primeiro *script* (mais simples):

(prototipe.com.br/disserta/vml2obj.py) Acesso em Maio de 2024

(https://drive.google.com/file/d/1nie5-IADVGAExulxHeI39GNHNu4OFBYB/view?usp=drive_link) Acesso em Agosto de 2024

```

1 # vml2obj (.vml to .obj).
2 #
3 # 09/Julho/2023 Script Python que pode ser usado para converter arquivos .vml
  simples em arquivos .obj. Python 3 é recomendado. Ele roda no terminal com:
  python3 vml2obj.py Não se esqueça de colocar seu arquivo .vml no mesmo
  endereço do script e permitir a execução do script.
4 # Este script também roda no Thonny 4.1.1 (https://thonny.org/)
5 # Arquivos .vml são arquivos .xml criados para o protocolo verse destinado a
  modelagem 3D em rede entre outras possibilidades
6 # (https://en.wikipedia.org/wiki/Verse\_protocol).
7 # .obj Arquivos Wavefront são um formato clássico para arquivos 3D.
  (https://en.wikipedia.org/wiki/Wavefront\_.obj\_file)
8 # Este script suporta apenas vértices e faces no momento.
9 # Autores: Alexandre Villares, OM Yasuda, Roberto Araki and Francisco Ortiz de
  Carvalho.
10
11 import xml.etree.ElementTree as ET # a forma mais usual é importar as coisas
  no começo.
12                                     # The usual form is to import things at the beginning.
13
14 def vml2obj(fin, fout):
15     # Analisa o arquivo vml em uma árvore.
16     mytree = ET.parse(fin) # isso aqui abre e lê o XML.
17     myroot = mytree.getroot()
18     # Prepara para escrever no arquivo.
19     with open(fout, 'w') as fobj:

```

```

20     # Lista os vertices preparando a conversão.
21     iVertex = 0 # Contador de vértices.
22     VertexDicio = dict()
23     for vertex in myroot[1][0][0]:
24         VertexNum, VertexX, VertexY, VertexZ = vertex.text.split()
25         fobj.write('v ' + str(VertexX) + ' ' + str(VertexY) + ' ' + str(VertexZ) + '\n')
26         iVertex += 1
27         VertexDicio[VertexNum] = iVertex
28     # Lista as faces com os vertices renumerados.
29     for face in myroot[1][0][1]:
30         aface = face.text.split()
31         fobj.write('f ')
32         for vertice in aface:
33             if vertice != '4294967295': # Investigar estes casos!
34                 fobj.write(str(VertexDicio[vertice]) + ' ')
35         fobj.write('\n')
36     # Aqui o arquivo fobj vai ser fechado!
37
38 # Exemplo para ler e escrever arquivos na mesma pasta do script.
39 entrada_vml = 'dumpTijolinoNocreate.vml' # Entrada.
40 saida_obj = 'saidaNoCreaseTesteThonny.obj' # Saída.
41 vml2obj(entrada_vml, saida_obj)

```

Segundo *script* (mais complexo, no entanto consegue transcrever mais que um nó de geometria e identificar exceções):

(prototipe.com.br/disserta/vml2obj32b.py) Acesso em Maio de 2024
 (https://drive.google.com/file/d/1_Ac6Z5zVFuvrv4veK mzRFLhBnYMKksAt/view?usp=sharing) Acesso em Agosto de 2024

```

1 # vml2obj (.vml to .obj).
2 #
3 # Maio/2024 Script Python que pode ser usado para converter arquivos .vml
  simples em arquivos .obj. Python 3 é recomendado. Ele roda no terminal com:
  python3 vml2obj.py Não se esqueça de colocar seu arquivo .vml no mesmo
  endereço do script e permitir a execução do script.
4 # Este script também roda no Thonny 4.1.1 (https://thonny.org/)
5 # Arquivos .vml são arquivos .xml criados para o protocolo verse destinado a
  modelagem 3D em rede entre outras possibilidades
6 # (https://en.wikipedia.org/wiki/Verse\_protocol).
7 # Arquivos .obj Wavefront são um formato clássico para 3D.
  (https://en.wikipedia.org/wiki/Wavefront\_.obj\_file)
8 # Este script suporta apenas vértices e faces no momento.
9 # Autor: Alexandre Villares, OM Yasuda, Roberto Araki and Francisco Ortiz de
  Carvalho.
10
11 import xml.etree.ElementTree as ET # a forma mais usual é importar as coisas
  no começo.
12
13
14 def vml2obj(fin, fout, verbose = False):
15     def report(msg):
16         if verbose:
17             print(msg)
18
19     # Analisa o arquivo vml em uma árvore.
```

```

20 mytree = ET.parse(fin) # isso aqui abre e lê o XML.
21 myroot = mytree.getroot()
22 # Prepara para escrever no arquivo.
23 with open(fout, mode="w", encoding="utf-8") as fobj:
24     # Lista os vertices preparando a conversão.
25
26     countnode = 0
27
28     for aObject in range(len(myroot)):
29
30         countnode += 1
31
32         # tratando os vertices
33         iVertex = 0 # Contador de vértices.
34         VertexDicio = dict() # tabela de conversão de número de verticies
35
36         if myroot[aObject].tag == 'node-geometry':
37             # encontramos geometria (objeto).
38             report(myroot[aObject].tag + " : " + myroot[aObject].attrib['name'] + "
found")
39             fobj.write('o ' + myroot[aObject].attrib['name'] + '\n' )
40
41
42             for vertexies in myroot[aObject][0][0]:
43                 VertexNum, VertexX, VertexY, VertexZ = str(vertexies.text).split(" ")
44                 fobj.write('v ' + str(VertexX) + ' ' + str(VertexY) + ' ' + str(VertexZ) + '\n')
45
46                 iVertex += 1
47                 VertexDicio[VertexNum] = iVertex
48
49             # Lista as faces com os vertices renumerados.
50             for face in myroot[aObject][0][1]:

```

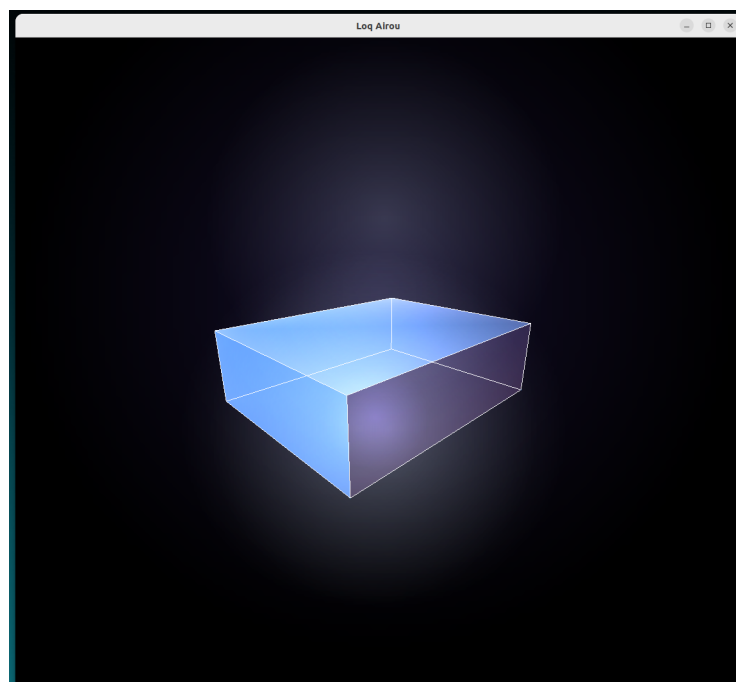
```

50         aface = str(face.text).split(" ")
51         fobj.write('f ')
52         for vertice in aface:
53             if vertice != '4294967295': # Investigar estes casos!
54                 fobj.write(str(VertexDicio[vertice]) + ' ')
55         fobj.write('\n')
56
57     else:
58         report(myroot[aObject].tag + " : " + myroot[aObject].attrib['name'] + "
skipped!")
59     return
60
61 # print(vmlcountgeometry(entrada_vml))
62 vml2obj("./dumpTijolinho.vml", "./dump30.obj", True)

```

Abaixo segue exemplo de arquivo .xml bem simples que pode ser usado para estudar o funcionamento dos *scripts*. Ele desenha o bloco da Figura 27.

Figura: 27 - Bloco simples construído no Loq Airou



Fonte: elaborada pelo autor (2024)


```

1 <?xml version="1.0" encoding="latin1"?>
2
3 <vml version="1.0">
4
5 <node-geometry id="n1" name="Geometry_Node_1">
6   <layers>
7     <layer-vertex-xyz name="vertex">
8       <v>1 0.489197 -0.049386 -0.949375</v>
9       <v>2 -0.280556 -0.049386 -0.949375</v>
10      <v>4 0.489197 0.241667 -0.949375</v>
11      <v>5 -0.280556 0.241667 -0.949375</v>
12      <v>6 -0.280556 0.241667 0.000000</v>
13      <v>7 0.489197 0.241667 0.000000</v>
14      <v>8 0.489197 -0.049386 0.000000</v>
15      <v>9 -0.280556 -0.049386 0.000000</v>
16    </layer-vertex-xyz>
17    <layer-polygon-corner-uint32 name="polygon">
18      <p>1 4 5 2</p>
19      <p>6 7 8 9</p>
20      <p>4 1 8 7</p>
21      <p>7 6 5 4</p>
22      <p>9 2 5 6</p>
23      <p>1 2 9 8</p>
24    </layer-polygon-corner-uint32>
25    <layer-polygon-corner-uint32 name="crease">
26      <p>4294967295 4294967295 4294967295 4294967295</p>
27      <p>4294967295 4294967295 4294967295 4294967295</p>
28      <p>4294967295 4294967295 4294967295 4294967295</p>
29      <p>4294967295 4294967295 4294967295 4294967295</p>
30      <p>4294967295 4294967295 4294967295 4294967295</p>
31      <p>4294967295 4294967295 4294967295 4294967295</p>

```

```

32     </layer-polygon-corner-uint32>
33     <layer-vertex-real name="select">
34         <v>1 0.000000</v>
35         <v>2 0.000000</v>
36         <v>4 0.000000</v>
37         <v>5 0.000000</v>
38         <v>6 0.000000</v>
39         <v>7 0.000000</v>
40         <v>8 0.000000</v>
41         <v>9 0.000000</v>
42     </layer-vertex-real>
43 </layers>
44 <vertexcrease layer="" default="0"/>
45 <edgecrease layer="crease" default="0"/>
46 </node-geometry>
47
48 </vml>

```

Convertido para Wavefront .obj a partir dos *scripts*, o arquivo do bloco simples de seis faces, citado anteriormente, fica com a seguinte configuração:

```

v 0.238565 -0.030814 -0.506365
v -0.394444 -0.030814 -0.506365
v 0.238565 0.258333 -0.506365
v -0.394444 0.258333 -0.506365
v -0.394444 0.258333 0.000000
v 0.238565 0.258333 0.000000
v 0.238565 -0.030814 0.000000
v -0.394444 -0.030814 0.000000
f 1 3 4 2
f 5 6 7 8
f 4 5 8 2
f 4 3 6 5
f 1 7 6 3
f 8 7 1 2

```

Essa dissertação foi escrita no Libre Office Writter versão 7.6.7.2 (x86_64).

A versão em .odt (Open Document Text) pode ser encontrada em:

https://drive.google.com/file/d/1iJ0a8-2QRUF0Slp_RQdpZBL0XqS3z8vJ/view?usp=drive_link

e também em:

https://prototipe.com.br/disserta/Francisco_Ortiz_De_Carvalho_3D_2024_UNESP.odt