OpenSim 入門 その2（v0.8/0.9 対応）第 2 版

OpenSim 入門 その2 (v0.8/0.9 対応) 第 2 版

『キャラクター・プロフィール（再掲載）』



名前 リサ (Lisa)
性格 Macintosh という林檎が好き
OpemSim にはついてそこそこ詳しい．しっかりモノである．



名前 クラリス
性格 Linux については比較的初心者
すぐに何でもやろうとするが大体失敗する．



名前 ソナタ (Sonata)
性格 普段は Linux を使ってるが実は Mac 好き
OpenSim の管理をやってるがおっちょこちょいで，昔 MO にファイルをバックアップするとき，Mac に SCSI ドライブを刺したらマシンがクラッシュして，原稿が全部吹っ飛んだこともあるらしい．



名前 Fumi Hax
昭和3x年生まれのハッカー崩れ．
自称 NSL の鬼教官．オールドプログラマー．まだまだ元気 !?
最近老眼が …. 何の，まだまだ若いもんにゃ負けん !!
うりゃ〜 !!
未熟者め．修行が足らん !!!!! タイヤ交 … じゃない．Linux インストール・設定 100 回じゃ !!!!!!!!!

日本メタバース技術研究会
Japan Open Grid
JOG ユーザ会
東京情報大学 ネットワークシステム研究室 井関ゼミ
キャラ原案・イラストレーション 「サミッティー」

目次

1. はじめに 1
1.1 OpenSim とは何か 2
1.2 本書の大まかな内容 2

コミュニティ編

2. Japan Open Grid 3
2.1 Japan Open Grid とは 4
2.1.1 JOG ってなにかしら ..... (by ikim Destiny) 4
2.1.2 JOG ージャパンオープングリッドー とは？ (by motoko Moonwall) 10
2.2 JOG コミュニティ 13
2.2.1 OpenSim で Café (by misaki Wisent) 13
2.3 JOG 利用入門 (by motoko Moonwall) 15
2.3.1 JOG ー ジャパンオープングリッドへの入り方 その1 15
2.3.2 JOG ー ジャパンオープングリッドへの入り方 その2 21

3. ハイパーグリッドの紹介 25
3.1 ハイパーグリッドの巡り方 26
3.1.1 グリッドのアドレス 26
3.1.2 他のグリッドへのテレポート 26
3.2 教育用 SIM の紹介 (by Yan Lauria) 27
3.2.1 The city of math, Mathland@Craft World 27
3.2.2 Edu3d@Craft World 27
3.2.3 DigiLandia@Craft World 28
3.2.4 Museo del Metaverso@Craft World 28
3.2.5 Micrographia/ 顕微鏡図譜 @Kitely 29
3.2.6 Jurassic Park@FrancoGrid 29
3.2.7 Moya 美術館 @FrancoGrid 30
3.2.8 Cherry Manga@FrancoGrid 30
3.2.9 Archaeology@University of StAndrews (Apollo Grid) 31
3.2.10 Abyss Observatory/ アビス海文台 @Japan Open Grid 31
3.2.11 ″Wind Observatory″ and ″Wind Circle″/「風の展望台」と「風の円」@Japan Open Grid 32
3.2.12 The Modern Museum/ 近代博物館 @Japan Open Grid 32
3.2.13 Paradise Island@Japan Open Grid 33
3.2.14 Le Voyage De Chihiro/Spirited Away/ 千と千尋の神隠し @Japan Open Grid 33
3.2.15 Shibuya and Osaka/ 渋谷と心斎橋 @Japan Open Grid 34
3.2.16 Four Season Botanical Garden/ 四季の植物園 @Japan Open Grid 34
3.2.17 Ivory City of Prims@Sanctuary 35
3.2.18 The Spirit of Arcadia@Sanctuary 35
3.2.19 GENOME Island@VIBE 36
3.2.20 Nova Archaeology@VIBE 36
3.2.21 Africa Traveler@VIBE 37

3.2.22 Pompeii@Metropolis 37
3.2.23 Gerry Anderson's World@AiLand 38
3.2.24 Oil Rig@AiLand 38

技術編

4. OpenSim のインストール 40
4.1 Winows へのインストールとスタンドアローンモードでの起動 41
4.1.1 ファイルのダウンロード 41
4.1.2 スタンドアロンモードでの起動 41
4.1.3 初期設定 41
4.1.4 Local サーバへのログイン 42
4.1.5 Local サーバの停止 43
4.2 Linux へのインストール 43
4.2.1 開発環境のインストール 43
4.2.2 pkgconfig のライブラリディレクトリ 43
4.2.3 Mono のインストール 43
4.2.4 OpenSim のコンパイルとインストール 45
4.2.5 tmux コマンド 45

応用編

応用編 48

5. Japan Open Grid へのグリッド接続 48
5.1 事前準備 49
5.2 IP アドレスの設定（ルータが NAT ループバック非対応の場合） 49
5.2.1 外部向けのネットワーク情報を得る 49
5.2.2 PC のローカル IP アドレスを得る 49
5.2.3 hosts ファイルの書き換え 50
5.3 IP アドレスの設定（ルータが NAT ループバック対応の場合） 50
5.3.1 外部向けのネットワーク情報を得る 50
5.3.2 PC のローカル IP アドレスを得る 50
5.4 ファイアウォールの設定 51
5.4.4 ルータのファイアウォールの設定（ポートフォワーディング） 51
5.4.2 Windows のファイアウォールの設定 51
5.5 OpenSim の起動 53
5.5.1 Regions.ini 53
5.5.2 OpenSim の起動 54
5.5.3 自分の SIM への移動 55
5.6 より進んだネットワーク設定 55
5.6.1 ローカル IP アドレスの固定化 55
5.6.2 外向け（グローバルな）IP アドレスの固定化 56
5.6.3 ドメイン名の取得 56
5.6.4 実際の運用 56

7. オールインワン ポータブルシステム（Windows 用 ） 57
7.1 システムの概要 58
7.2 取りあえず OpenSim-Moodle all-in-one Portable を使う 59

7.2.1 パッケージのダウンロードと展開 59
7.2.2 初期設定ツールの起動 59
7.2.3 Moodle（Apache と MariaDB）の起動 59
7.2.4 OpenSim の起動 60
7.2.5 システムの利用 60
7.2.6 システムの停止 62
7.3 トラブルシューティング 62
7.3.1 Start Moodle.exe が失敗する 62
7.3.2 Reset_Env.bat で表示される IP アドレスがおかしい 62
7.4 各種パスワードの変更 63
7.4.1 Moolde 63
7.4.2 OpenSim 63
7.4.3 MariaDB（mysqld）の管理（root）ユーザ 63
7.5 Xampp 63
7.6 Modlos の設定 64
7.6.1 一般メニュー 64
7.6.2 管理メニュー 64
一般メニュー 64
7.6.3 全般設定 66
7,7 OpenSim の追加モジュールとヘルパースクリプト 66
6. Unity3D への Data 変換 68
6.1 OAR Converter の概略 69
6.2 OARConvWin 69
6.3 データ変換の手順 70
6.3.1 OAR ファイルの作成 70
6.3.2 OAR ファイルの読み込み 70
6.3.3 Collada 形式への変換 71
6.3.4 Unity でのデータの読み込み 71
6.4 変換結果 72
8. 終わりに 73

1. はじめに

この本は，前作「OpenSim サーバシステム構築入門」の続編です．2010 年に前作を出版後，我々は 2013 年 12 月から日本人向け OpenSim グリッドである Japan Open Grid（JOGrid/JOG）のサービスを開始し，現在に至っています．

そこで今回は，OpenSim(JOG) のユーザ視点の記事も盛り込んでいます．一方，技術志向的な内容に関しては,前作 (OpenSim v.7.x) との差分のみ記述しています．技術的な内容にご関心のある方は，前作「OpenSim サーバシステム構築入門」も併せてご覧ください．なお，前作「OpenSim サーバシステム構築入門」は下記 URL から，PDF ファイルとしてダウンロード可能です．

・**OpenSim サーバシステム構築入門**：
http://www.nsl.tuis.ac.jp/xoops/modules/d3downloads/index.php?page=singlefile&cid=8&lid=30



皆さん，お久しぶり!!

6 年ぶりだね～





私も色々成長したよ～

でも，おっちょこちょいは全然治ってないね．





という訳で，今回は Simian Grid 等の解説はなしです．予告していたのにすみません．

1.1 OpenSim とは何か

OpenSim（OpenSimulator）は Second Life サーバと互換性のある 3 次元仮想空間構築用のオープンソースのサーバシステムです（http://opensimulator.org）. OpenSim はクライアントとの通信に Second Life の通信プロトコルと同じプロトコルを使用します. 現在 Second Life 用の公式ビューア（Viewer）では OpenSim を利用できませんが（接続先サーバを Second Life 用サーバから変更できない), OpenSim でも使用可能なサードパーティ製のビューアが多数公開されています. この本の執筆時点での OpenSim の最新バージョンは 0.9.0（RC1）です（開発バージョンは 0.9.1Dev）.

世界的に有名な OpenSim のサービスとしては OSgrid（http://www.osgrid.org）が挙げられます. 誰でも無料でユーザ登録でき，自分の PC にインストールした OpenSim を OSgrid の一部として接続させること(OSgrid のグリッドに参加すること)ができます. しかし,OSgrid の公用語は英語であり，英語の苦手（!?）な我々日本人には敷居が高いものがあります. そこで，我々は 2013 年 12 月から日本人向けの無料の OpenSim サービスとして Japan Open Grid（JOGrid/JOG）のサービス（http://www.jogrid.net）を開始しました. 恐らくは日本国内での初の本格的な OpenSim グリッドサービスです（これ以前にも，小規模の実験的なものはいくつかありました）.

JOG（ジョグ）では OSgrid とほぼ同等のサービスを提供しており，一部の機能は OGgrid を超えていると自負しております. これを機会に是非皆さんもお試しください. なお，JOG は公用語が日本語というだけで,日本人以外の外国人に対する使用制限等は一切行っておりません（外国人のユーザもいらっしゃいます）.

1.2 本書の大まかな内容

本書は大きくコミュニティ編，技術編と応用編の三つに分かれています. コミュニティ編，技術編（第 1 版）は 2016 年 4 月に PDF として公開しましたが，その後の OAR(OpenSim ARchiver) ファイルを Unity3D データへ変換する OAR Conveter の開発や，誰でも手軽に使用できるようにしたポータブルシステム（OpenSim-Moodle Portable）の開発を経て，それらの解説を応用編として加筆・修正したものが本書（第 2 版）であります.

コミュニティ編では,OSgrid や JOG の参加の仕方や OpenSim 内でのコミュニティの紹介など，ユーザ視点の内容となっています. 執筆も実際の JOG ユーザが担当しました.

技術編では，OpenSim が v0.7 から 0.8（0.9.0）に変わったのを踏まえて，前作「OpenSim サーバシステム構築入門」からの変更点を記述しています. また，一般ユーザのために Windows でのスタンドアロン起動の解説や自分の PC 上に OpenSim を構築し，JOG に接続したいという人のために，その手法も紹介します. さらに上記でも述べた通り，応用編では当研究会でした OAR Converter やお手軽に OpenSim を実行管理できる OpenSim-Moodle Portable システムについての解説も行っております.

コミュニティ編

2. Japan Open Grid

ここでは，Japan Open Grid のコニュニティで活躍するユーザさん達から，Japan Open Grid に関する記事を寄稿して頂きました.Japan Open Gridのユーザさん達の生の声が聞けると思いますので，是非ご一読ください.



JOG もいっぱいユーザさんが増えたね.

そうだね．最初は閑古鳥が鳴いていたのに！





JOG ユーザ会というユーザ主体の会もあるんだよ.
先生はそう言う運営面は苦手だから，すごく助かっているみたい.

実際苦手なので，大変助かっています（＾＾



2.1 Japan Open Grid とは

2.1.1 JOG ってなにかしら …… (by ikim Destiny)

JOG（ｼﾞｪｲｵｰｼﾞｰ）とは、Japan Open Grid（ｼﾞｬﾊﾟﾝｵｰﾌﾟﾝｸﾞﾘｯﾄﾞ）の略称です。

皆さんも良くご存知（なのかしら？）のメタバースの一つです。そもそも、メタバースってなにかしらぁ、と思う方もいらっしゃる事と思います。 ?(ﾟ_ﾟ)?(。_ﾟ)?

既にご存知な方も、そうでない方もちょっと覗いて見てください。そして、興味を持って頂いて、試しにやってみようかしら、と思って頂けたらとっても嬉しゅうございます。（人´∀`）

OpenSim ってなにかしら …… ？

この JOG は OpenSim（ｵｰﾌﾟﾝｼﾐ）と呼ばれる仮想世界サーバーのソフトウェアを実装して、シミュレーションされた仮想世界を利用できる場、として存在しております。

現存している大手メタバースの中には、Second Life（ｾｶﾝﾄﾞﾗｲﾌ）と呼ばれる 3 次元仮想空間サービスがありますが、この OpenSim は Second Life のサブ的存在やクローン的存在ではなく、ユーザーさん達によって拡張機能を追加したり、アプリケーションを作成して追加したりといった行為が行えるシミュレーションシステムです。そのシミュレーションシステムを各ユーザーさんが所有するサーバー内に作成する事も可能です。

言い回しが難しくなっちゃいましたけれども、一例として、パソコン内に 3 次元の箱庭ジオラマを作れるシステム、とお考えいただければ解かり易いかと思います。（´∀`）ﾉ

っていうか、メタバースって何かしら …… ？

メタバースというのは、コンピューターで作成される、インターネット上に存在する仮想世界の事を意味しています。(σ´-｀)ﾌﾑﾌﾑ

一般的には 3 次元コンピューターグラフィックス（3DCG）によって、『高さ』や『奥行き』などの立体空間をコンピューターの画面内などに表現されるものの事を総称している様です。

こういうお話ですと、『あぁ、某、ファイナルファン〇ジーオンラインや、某、ドラ〇ンクエストオンライン、某、Te〇a ですとか、最近だと、某、黒〇砂漠』のような MMORPG をイメージされるかと思いますが、このメタバースというものは、それらのゲームのような背景の物語や、クエストのように『お使いに出向いて』『何かを取ってくる』というような決められた目的があるわけでもなく、また、倒すべき敵なども存在しません。（￣д￣）ｴｰﾃｷｲﾅｲﾉｰ？

決められた目的がないって、どゆこと？

現在、皆さんが存在している現実世界は、高さ、幅、奥行きで成り立つ立体空間であるのはご存知の通りです。その立体空間をパソコンの中に作っちゃいました。 。+.ﾟヽ(o´∀`)ﾉﾟ.+。ﾜｰｲ　というのがメタバースだと思っていただければ、解かりやすいと思います。

そのメタバースの中は先ほど書きましたとおり、決められた目的は存在しません。皆さんの現実世界では生活する為の目的(衣食住の充実の為のお仕事など半強制的な行為)はあるかと思いますが、それ以外は自由であり、決められた目的は、ほぼ存在しないと思います。

極論で、一日中目的なくボーっとしていても、親や上司などに怒られる事はあるかもしれませんけど、目的無く過ごしても何もおこりませんよね。(*´艸`)ﾌﾌﾌ

現実のような衣食住の充実のための半強制目的がない世界がメタバースと、いえるかもしれません。

目的は無いのに、何に使うの？

決められた目的は存在しません、と先に書きましたけれども、使い方はユーザさん達が自由に考え、自由に決める事ができます。何もしないで、ただメタバース内に立っているだけ、という事も

出来ます。

例をあげますと、メタバース内で学術的な研究を目的となさる方もおられます。研究？と聞いただけでアレルギーを感じてしまいそうで、何だか難しそう…. と思ってしまいそうですが、全てがそうではありません。

先に例としましたMMORPGですが、このメタバース内でユーザーさん同士がルールを作ったり、装備などを自作、またはどなたかが作成した装備アイテムをメタバース内で購入してRPGの世界感を作ったり、またはスクリプト（メタバース内で効果を発揮するプログラム）を作成しアイテムに搭載することによってゲーム化する事も不可能ではないでしょう。

もう一つ、例をあげてみます。芸術家さんや、音楽家さんのような表現者さんたちの発表の場、という使い方を実施されているユーザーさんもいらっしゃいます。ご自身が作成された絵画、写真などを、このメタバース内に持ち込んで展示する方。現実世界では実現が困難な形状をメタバース内で作成する方もおられます。

現実世界で実現が困難なものって？

現実世界では実現が困難な形状といわれても、すぐに あ！あれね！ M9(`･ω･´)っ ｿﾚﾀﾞｯ‼ とイメージが沸かないかと思います。

例えば、イタリア製のいわゆるスーパーカーと呼ばれる乗り物や、他にも飛行機、ロケット、豪邸などを現実世界で入手したいと仮定します。それらを現実世界で単独で自作できるはずも無く、また実際に購入などによって所有するには高額な金額を用意するなどの必要性が生じます。その様な理想をメタバース内では実現する事が可能です。（あくまでも仮想世界内ですけれども）

現実世界で表現が困難な物たちを仮想世界内では重さなどの制約がないことから、重量のある大型なものの表現をしやすいという利点があります。



現実には存在しない物体の例
メタバース内で作成された空想の乗り物と、
空想の建物の例

車や建物などの硬い物を仮想で作るだけ？

スーパーカーや豪邸などに限らず、現実には存在しないデザインのお衣装などもメタバース内では作成が可能でしょう。ファッションデザイナーさん達にとっては、現実に縫製する前に立体化することでイメージの把握が可能だと思います。仮想世界のマネキンへ着せ替えることも出来るかもしれません。

アクセサリーについても、加工が難しい硬度のものを仮想内では簡単に加工する事ができますので、デザインイメージも掴みやすいと想像します。

物体を表現するだけの場なの？

物事の表現は物体を表すだけとは限りません。音楽家さんでしたら、ご自身が作曲された曲や現実に演奏された曲などを、メタバース内で演奏する事で表現することができます。その音楽家さん

が現実世界でコンサートなどを実施しようとした場合、会場の手配など手続き等が必要となりますが、メタバース内では現実世界のような手続きは発生しません。

JOG ではコンサート会場も演奏者さんや会場設営クリエイターさんたちの手により作成され、コンサートの実施実績もあります。コンサート内では、照明の効果などの視覚効果を用いる事も、現実世界に比べ容易に可能とされる事でしょう。

表現者さんにはよさそうだけど……

特に表現するものが無いとお考えの皆さん。そんな事はありません。後ほどお話ししますが、メタバースではどなたでも表現者と成り得ます。 d(￣ー￣)ﾆﾔﾘ

更に例をあげますと、メタバースの利用の一つにユーザーさん同士の交流というのがあります。これは、ただのチャットじゃないの？と、想像する方もいらっしゃる事でしょう。通常、チャットというものは、活字のみのやり取りでお互いの交流が完了してしまいます。メタバース上でのユーザー同士の交流は、メタバース上で各ユーザーさんが所有するご自身の分身として、『アバター』という仮想人体を用いる事で通常の活字チャットだけではない表現をチャットと同時に行う事ができます。チャット用アバターは、某ブログサイトで有名なアメ〇バなどが知られていますが、そのアバターの自由度を更に向上させたものとお考え頂ければ良いかと思います。



某ブログサイトで有名なアメ〇バのアバター例



左：某アニメに登場する、現実世界には実在しないキャラクター
右：キャラクターを模して、メタバース内で作成されたアバター
（メタバース内のアバターの自由度の高さの例）

自由度の高いアバターって？

このアバターという仮想人体は、性別や体型、髪型、人体以外の形状など、自由度の高い編集をする事ができます。ユーザーさん個人々々でご自身を現したいといった欲求を満たしてくれる事でしょう。

例えば、現実世界では出来ない髪形、服装、背の高さを高くしたりといった表現をしたり、目の色を変える、お顔の形や目、鼻、口、耳などの形状を理想的に変えてみたり、メイクを施すなど。又は人体以外に動物や現実には存在しないキャラクターをご自身の分身として扱う事も可能です。

ご自身の分身と交流相手のアバター同士でやり取りをしていると、次第にご自身のアバターに愛着が沸き、一層着飾ったり表現を豊富にしたい気持ちになるはずです。アバターを着飾る事によって、それは既に『ご自身を表現する表現者』となっているといえるでしょう。そこから広がる会話もあることでしょう。



メタバース内で作成されたアバターの顔にある目、鼻、口、頬骨の高さやほほ肉の付き具合、顎などのパーツ位置や形状、体型などを調整する事で人物像を表現した例

具体的に、どういうことができるの？

先ほど書いた内容に類似したこととなりますが、メタバース内では現実世界で手にしたいと思ったものの殆どを、ご自身で作成する事ができます。ご自身の分身であるアバターをどの様にしたいかもご自身で作成する事もできるので自由度が高い事と思えるはずです。

これらの物達（アイテムなど）を作成するには、ちょっとした技術が必要となります。普段皆さんがご使用になっているパソコンでも、Excel や Word、Photoshop といったソフトを用いて色々作成したり利用している事と思います。同様に、このメタバース内で利用するアイテムも数ある作成ソフトの中から、ご自身が扱いやすいソフトを用いて作成し、メタバース内に持ち込む事で、ご自身の分身であるアバターに使わせる事ができるのです。これらのアイテム作成用のソフトは、フリーソフトで配布されている物もあります。

更に具体的にはどういうことが？

メタバース内では仮想の通貨が存在するものがあります。JOG においては J$ と呼ばれる仮想通貨によって通貨流通が実装されており、JOG 内ではこの仮想通貨を用いて全てやり取りされています。この J& は JOG 内の所定の場所に配置されている J$ ディスペンサーから手続きを踏む事で現実の円や $ などのリアルマネーを用いずに無料で入手可能となっています。 ｡+.ﾟヽ(o´ ∀`)ﾉﾟ.+｡

仮想通貨が流通しているメタバースでは、各ユーザーさんが自作されたアイテムを仮想通貨によって販売し、その販売したことで得られる通貨を貯めて他の作成者さんが作られたアイテムを購入する といった流通により商業活動を行っているメタバースも存在します。

もっとくだけた言い回しにしますと、子供の頃におもちゃのお金でお店屋さんごっこをした覚えはございませんか？ （ ´ ∀`） そのお店屋さんごっこでは、売り買いをするために、紙や粘土細工などで商品を自作なさって、お友達同士で顧客と店員とに成りきり、楽しんだことと思います。この仮想世界のなかで、商品となるものを PC 内のソフトなどで作成し、メタバース内の仮想通貨を用いて流通を図るという事がこのメタバース内で行える事柄の一つなのです。（人 ´ ∀`）

実は、この仮想通貨は現実世界の通貨へ変換できるといった仕組みも世界中には存在しており、仮想世界で販売したものによる利益を現実世界の通貨へ変換する事で現実の企業利益とする思想もあるようです。



JOG におけるショッピングモールの例

先に触れましたが、ご自身で作曲などをなさる音楽家ユーザーさんは、このメタバースの世界でも音楽活動を行い、ご自身の楽曲をアピールなさっていらっしゃいます。それらのアピール行動によってユーザー間での口コミで現実世界にも広がり、オンラインでの楽曲販売を実際に行っていらっしゃるユーザーさんも存在しているのです。

今後、未来のメタバースの世界は、メタバースを介しましてオンラインを用いての実際のお買い物（ネット通販的なもの）や、オンライン学校などへ展開されるかもしれません。

作る事って何だか難しそう.....

アイテムを作成するにはソフトを用いると先ほど書きました。『ソフトを扱う、作る行為』、それらがどうしても難しいとお考えの方がいらっしゃったとします。『作るためのソフトが使えないと、何も作れないから表現なんて何も出来ないんじゃないの？楽しめないんじゃないの？』と考えてしまうかもしれません。

ところがです。メタバースは、世界中に広がっています。メタバース内で扱える製作物のクリエイターさんも、世界各国に存在しています。世界中のクリエイターさんが制作されたアイテムを購入して歩くといった楽しみ方も出来ると思います。その場に居ながらにして、世界中からお買い物しまくりです！ ｡+.ﾟヽ(o´∀`)ﾉﾟ.+｡ 世界中のアイテムを購入し、組み合わせて使用する事で、自己表現とするユーザーさんもいらっしゃいます。

現実世界でも日頃みなさんも衣服などで着飾る事でご自身のファッション性を表現なさっている事と思います。これらは、いろんなアイテムの組合せでご自身を表現する一つだと思います。このメタバースの世界も同様に、何かを表現する行為の始まりをそう言った所から始めるのもよいかもしれません。

メタバースって世界中なの！！？

世界各国のユーザーさんと接続する事ができるのが、メタバースの世界です。当然ながら、会話言語はそのお国の言葉となりますので、英会話の上達に繋がると考えるユーザーさんもいらっしゃるようです。

チャットは活字だけでなく、ボイスチャットといったご自身の声を使ったものも存在します。これには、ヘッドセットやマイクロフォンなどのハード面（機器類）をご準備頂く事になりますが、メタバースを介してご自身の声を会話の相手に伝える事ができ、活字によるチャットだけでは伝えきれない会話のニュアンスや雰囲気までも伝える事が可能となります。これらを活かして、海外のアバターを前にして英会話を実施する事も可能となります。

活字チャットで海外ユーザーとのやり取りについては、翻訳ソフトを用いるという方法もありますので、『海外の言葉は苦手』とお考えの方でも交流は可能かと思います。

物体の形を作れるだけではないのです

メタバース（又はOpenSim）で可能なのは、造形物だけではないのです。メタバース（又はOpenSim）はサーバー内にシステム構築されたものと触れました。これらのシステム内において、プログラミングについてお詳しい方は、造型物だけではなく、システム的な作成物（スクリプト等）を持ち込む事が可能です。形は見えませんがスクリプトやプログラミングは非常に高度な表現力をもっています。そのプログラミングされた作成物(スクリプトと呼称します)によって、メタバース(又はOpenSim）内でのアバターをはじめとするあらゆる物体の表現を飛躍的に向上させる事が可能になるでしょう。

例えば、物体に動作をさせるスクリプトを持たせる事で動くことを可能にしたり、アバターの動作によって音を出させるなどで一層表せるものが広がります。

MMORPGの部分で先に触れましたが、これらのスクリプト等（メタバース内で効果を発揮するプログラム）を仕掛けることにより、メタバース内でユーザーさん同士の対戦など、ゲーム性を持たせたり、乗り物にのって他のユーザーさんが操るアバターと競い合うなどの楽しみ方も可能となるでしょう。

このメタバースではただ単にアバターであるお仮想人体がチャット会話するだけ、といった単調なものではないのです。

使い方の色々.....

いくつか例をあげてみましたけれども、メタバースでの楽しみ方は作ることが主な印象になって

しまったかと思います。こういった仮想世界のなかでは、ご自身がお店を持って運営、経営なさるユーザーさんも多くいらっしゃるようです。といっても、仮想通貨ですので、直接現実の利益になるわけではありません。

『お店』と漠然とお話ししましたが物を売るだけがお店ではないのがメタバースの世界です。一例をあげますと、人々が集まる場所をお店として、社交場のような場を経営なさる方もいらっしゃいます。

憩いの場を自作するという楽しみ方

多くのユーザーさんが集まり易い雰囲気や、そうした場をご自身のセンスで店内を彩るといった事もまた表現の一つとなりえるでしょうし、その集まる場では日常的な話題や、こういったメタバースに関する技術情報の交換を行うなどが活発に行われることで、交流を深めることができます。

このメタバースにおいて、ユーザーさんが単独で物作りに没頭して楽しむのも良いでしょう。それだけではなく、メタバースの世界の中でユーザーさん同士が集まる場はまた楽しい事に繋がるはずです。ユーザーとなる皆さんが憩いの場を自作する。という楽しみ方もメタバース（又はOpenSim）の醍醐味となるかと思います。 ﾜｸﾜｸ((o(ﾟ▽ﾟ〇)(〇ﾟ▽ﾟ)o))ﾄﾞｷﾄﾞｷ

ご自宅のPCにメタバースの世界をつくる

ここまではメタバースの中に入ってできる事の例を挙げてまいりました。このメタバースの世界に触れておりますと、次第に『自宅にこのメタバースの世界をおいておきたい』、『だれにも制約されず自分の自由に出来るSIMを持ちたい』、と希望する気持ちが芽生える事があります。（人´∀`）

実はこれも実現が可能です。(￣―￣)ﾆﾔﾘ

ご自宅のPCへOpenSIMを構築する事で自分だけのOpenSIMを作り、今まで例に挙げました行為が、ご自分のPCの中で行える事となります。しかしながら、この状態ではご自身だけが入る事ができるメタバースの世界でしかありません。ご自宅に折角開設したメタバースにご自身のみが存在するのでは、いささかさびしい世界と感じる事でしょう。

メタバースの世界は他のユーザーさんとの交流があってこそ、だと思います。構築したご自宅のメタバースへ他のユーザーさん達を招く方法が、ご自宅メタバースとJOGとの接続、という方法です。

メタバース内で出来る例や、ご自宅にメタバースを開設して他のユーザーさんとの交流を図ることが出来るお話しを、ほんの少し挙げてまいりました。これらのできる事を、JOGのOpenSimでやってみましょう、というのがこのOpenSim+JOGの入門書です。（人´∀`）

こちらをご覧の皆さんが、メタバースやOpenSimへ興味を持ってくださる一助となれば、幸いでございます。 。+.ﾟヽ(o´∀`)ﾉﾟ.+。

ikim Destiny

2.1.2 JOG ージャパンオープングリッドー とは？（by motoko Moonwall）

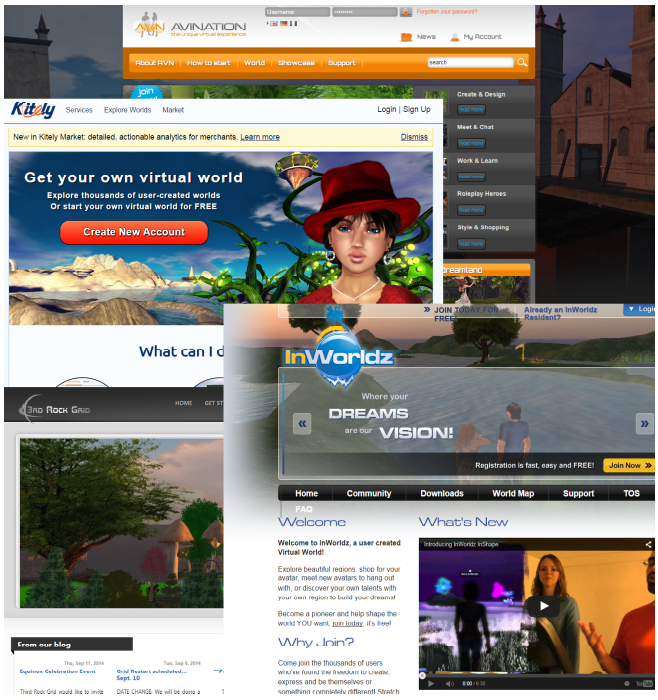
広大なネットの海には OSgrid をはじめとしてたくさんのオープングリッドが点在しています。たとえば FireStorm のグリッド選択リストを開いてみると

- 3RD Rock Grid
- Avination
- AviWorlds
- Craft World
- FrancoGrid
- Gay Nations
- Inworldz
- Island Oasis
- Kitely
- Littlefield Grid
- Metropolis Metaversum
- UFS Grid
- Your Alternative Life

こんなぐあいにずらりと OpenSim ベースのグリッドサービスの名前が並んでいます。そしてこれらのグリッドでは OSgrid もふくめてすべて英語（FrancoGrid はフランス語のようですが）が使われています。

つまりこれらのサービスを利用するにはホームサイトでのアカウント取得からはじまって，インワールドでの他人とのコミュニケーションや各情報の収集や発信をすべて外国語でこなさなければならないわけで，これは必ずしも語学が堪能とは言えないわたしのような一部の（大多数の？ ^^;）日本人にとってちょっと辛い状況です ^^;

そこで -- 完全に TV の通販番組風のいいまわしになりますけど ^^;-- そんな悩みを一気に解消するのが今回のタイトルにかかげた JOG グリッドサービスなのです。



Japan Open Grid のホームサイト

https://www.jogrid.net/wi/

を開くと，やれうれしや，「ようこそ!!　Japan Open Grid へ」と日本語の挨拶がでむかえてくれます ^^



さて，この原稿の元になったブログを書いていたのは 2014 年 9 月の下旬なのですが，その時はそのブログで何度もご紹介してきたあの OSgrid サービスがシステムトラブルのため長期にわたってダウンしていました。そのため OS グリッドで行っていたわたしたちの Blender についての勉強会が幾週にもわたって開けないという事態が続いていて，そのため当時やむなく全員がジャパンオープングリッドのアカウントを取得し，OSgrid に代わって JOG での集まりを開催することになりました。(注 1)

しかしその過程で JOG のアカウント取得とインワールドへのログインという作業が思いの外むずかしいということが分かりました。実際当日集まったメンバーのすべてをアバターとして目的のシムに集めるまでには，勉強会として予定していた時間の倍以上かかってしまったぐらいです。

JOG 管理スタッフの方々の名誉のために言い添えておきますが，これは必ずしも JOG のホームサイトの説明が不十分であるという意味ではありません。そこには必要な事柄は十分書かれています。

ただ SL や OSgrid といういわば「大手」のグリッドサービスに慣れた者から見ると自前の管理システムでなく Moodle という一般公開されたコースマネージメントシステムを援用している JOG のホームサイトのデザインは若干とまどってしまう部分を含んでいることも確かです。

加えて「大手」でないがためにビューアーのグリッドのリストに名前がなく (上の FireStorm にもありませんでしたよね？^^) 新たにリストに JOG の名前を登録する手順が必要である，という問題もあります。この手順，やっかいなことにビューアーごとに微妙にちがっているのでここでつまづいて立ち止まってしまう人もけっこういらっしゃるかもしれません。

そこでここではわたしたち勉強会スタッフの「はまった」点を参考にしつつ，JOG へのエントリー方法について可能な範囲で具体的かつケースごと説明していきたいと思います。(2.3 節参照)

（注1） こう言ってしまうとOSgridが使えないのでしょうがなくJOG にーー というふうに聞こえますが，けっしてそれだけではありません。

じつのところJOGの身近さはそれはそれで捨てがたいひとつの魅力です。インワールドでであうユーザーがほとんど日本人であること，規模が小さいおかげでどこで誰がどんな活動をしているかが手にとるようにわかること，同じ理由から管理スタッフの方たちとのコミュニケーションが非常にとりやすいこと… これらの利点はSLやOSgridのような大きなグリッドサービスではとうてい望めないことです。

いつかメタバースが将来ひとりひとりにより親しいものになり，個人のいとなむ客間的バーチャルワールドが縦横にハイパーリンクで結ばれるようなネット社会が到来する日がくると予想する人たちがいます。JOGのような小回りのきくグリッドサービスはそうした時代への橋渡しの役目をもつものかもしれません。

motoko Moonwall

編者コメント）

2.1.2項および2.3節の著者の motoko Moonwall（高木淳）さんは2016年2月にご病気のためご逝去されました．ここに謹んで哀悼の意を表します．

また本編におきまして，JOGサイトのデザイン更新により，motokoさんより頂きました画像を本文の意が通るように一部差し替えさせて頂きました．

motokoさんはJOGにおいても非常に活躍されていたユーザの一人であり，今後の活躍も期待しておりましたので，今回のことは非常に残念でなりません．

以下にJOGユーザのお一人である西村一さんによる高木淳（motoko Moonwall）さんの紹介ページのURLを記します．ご参照ください．

http://www.sf-fantasy.com/magazine/Jun-Takamoto/about_Jun-Takamoto_nishimura-hajime.html

2.2 JOG コミュニティ

2.2.1 OpenSim で Café（by misaki Wisent）

Japan Open Grid の住人となって、もうすぐ 1 年が経とうとしています。

今、何かと熱い Japan Open Grid ですが、私はここで知り合った友人と一緒に週に一度のカフェを開いています。

「で、カフェっていったい何をするところ？？」

チャットで情報交換する場所とでもいいましょうか。どこのグリッドでもそうですが Second Life のように常に人がいるわけではありません。ログインしているユーザーが全世界で数人とかでは､人に出会う確率は果てしなく 0 に近いことでしょう｡いつログインしても誰にも会わないのでは、面白くないですよね。

Japan Open Grid（JOG）に移住することになったきっかけは、OpenSim グリッドサービスの最大手「OSgrid」がシステムトラブルのために長期間オフラインになってしまったせいなのですが､ちょうどその頃、無料で SIM オーナーを募集するキャンペーンが JOG で始まり、JOG の人口は一気に膨れ上がったのでした。

とはいっても、ログインしている人数は、だいたい一桁（笑）。 0 なんてことも珍しくはありません。それでも、まだ小さな世界なので MAP で住人の気配はなんとなく感じられます。

「初期のころの Second Life みたい」

とは、住人たちの間でよく聞かれる感想です。私自身は、そのころの Second Life を知らないのですが異国で出会った日本人がコミュニティを作り始めた感じでしょうか。

JOG でも有志の住人でユーザー会が発足しました。

「ログインしたけど誰もいない」

「どこに行けばいいのかわからない」

という JOG 住人や新規ユーザーのために毎日決まった時間の集まりを作ろうということになりました。すでに行われている定時のイベントもあり、それに加わる形で私たちの Café::GardenOasis:: も誕生したのでした。

ところで、JOG のユーザーの大半は自分のリージョン（SIM）を持っています。Second Life でも SIM オーナーという人もなかにはいらっしゃいますが、大半の人は私のように「初めて自分の SIM を手に入れた！」という人ではないでしょうか。

Second Life で小さな土地を借りてプリム不足と戦いながら暮らす私の夢は、SIM オーナーになることでした。到底かなうことがないと思われたその夢は、こうして Second Life とは別の世界でかないました。ところが、予想に反して使い切れないほどの土地とプリムを与えられた私は途方に暮れてしまったのです。

景観を作ろうにも草も木も花もない。

素敵な家も家具もない。

Second Life では、インベントリーにあふれるほどにあるハイクォリティな品々が JOG の生まれたばかりの私のインベントリーには何もなかったのでした。

JOG に初めてログインしてショックを受けるのは、デフォルトアバターの姿です。

これがクリエーターさんなら自分の作品をアップロードすることができますが、それができない私には Second Life のようなアバターに近づけるまでの道のりは果てしなく遠いです。姿形もさることながら、髪もなければ服もない靴もない。

そんなときに、他グリッドから仕入れてきた品々を配布してくださったのは現ユーザー会の会長でした。ともすれば、自分のSIMに引きこもってしまいがちな住人のSIMをまわっていらっしゃったようです。

Second Lifeとは違って、他のグリッドにも行けることを知ったのもそのときでした。

おかげさまで、私たちのカフェも軌道にのり毎週たくさんの（といっても一桁）の住人に来ていただけるようになりました。

遠く外国からアクセスして下さる方もいます。年齢も性別も関係ない、ましてや現実では知り合いに成り得ない人とお友達になれるのはネットの醍醐味です。こんな素敵な場所を提供してくださったJOGと、何よりもカフェを共同経営してくれる友人、そして個性豊かなすてきな住人たちとの出逢いに感謝しています。

そんなわけで、ユーザー会から猪突猛進ペアとして恐れられるCafé::GardenOasis::の挑戦は、これからも続いていくのでした。

misaki Wisent



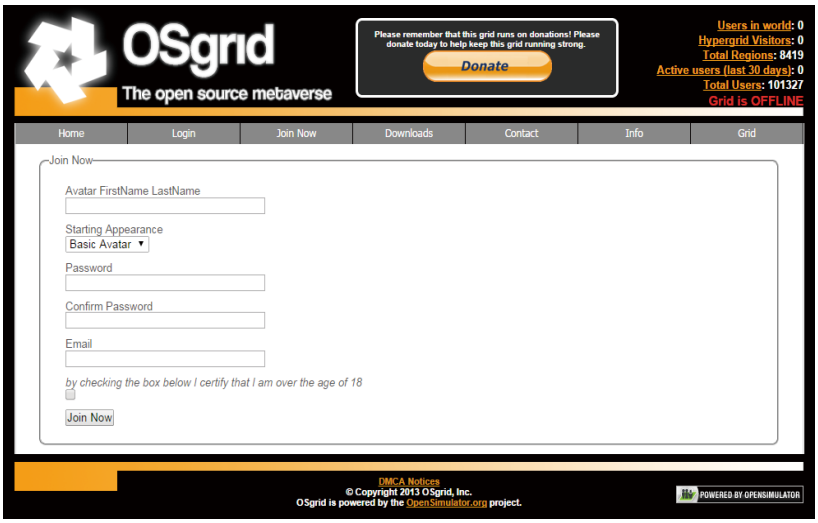




2.3 JOG 利用入門（by motoko Moonwall）

2.3.1 JOG ー ジャパンオープングリッドへの入り方　その1

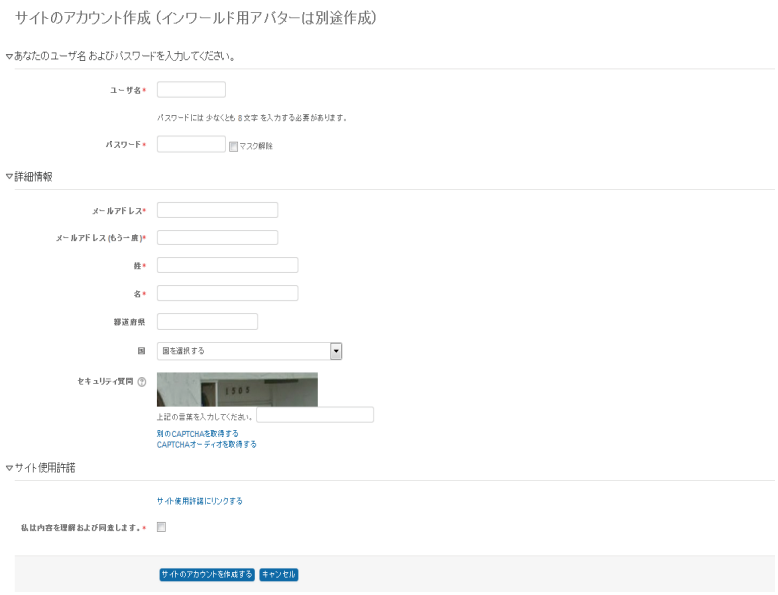
さてたとえば OSgrid の場合，ポータルページの「Join Now」をクリックするとこんな画面が開きます。



ここでアバター名とパスワードを登録することでアカウントがもらえます。この名前とパスワードはビューアーで OSgrid にログインするのに使えるし，またブラウザでウェブ上のこのサイトそのものにログインするときにも使えます。

これに慣れている人は JOG の最初のアカウント取得 ーーつまりこのページの右側の「ログイン→新しいアカウントを制作する」または「JOG に参加」で開くページーー を，ビューアーを使ってグリッドにログインするためのアバターの名前とパスワードであると錯覚してしまうかもしれません。





でもこれはあくまで「このサイト」にログインするための ID（名前）とパスワードなんです。

登録手続きするとメールアドレスに確認のメールが送られてくるので（ひょっとしたらブロックされてしまいスパムリストなどに入っているかもしれないのでもし届いてないという人はそっちを確かめてみてください），書かれた URL を開き登録手続きを完了します。

じつは JOG そのものにログインするためのアバター名とビューアーへのパスワードの登録は，あたらめてこの画面からはじめなければなりません。



サイトのアカウントをとったのですでにページの右上にあなたの名前がでているはずです。そのすぐ下，「JOGrid」のなかにある「アバター作成」をクリックしてください。



こんな画面が開くのでアバターの名前（セカンドネームはべつに「Resident」以外でも好きにつけられます）とパスワードを記入し，ホームとしてお好きな場所を選んでください。（もし特別な事情がないのであれば「Center Sim」にしておくのが無難です。）

「使用許諾に同意」に（ひととおり目を通した上で＾＾）チェックをいれ，「作成」ボタンを押してください。（「Sloodle との連携」は今は Sloodle は動いていませんので無視して結構です）

これでめでたくあなたのアバターが作られました。ビューアーにはこちらのほうの「名前」と「パスワード」をいれて JOG にログインしてください。

なんだかめんどーくさいなー，なんでサイトのアカウントと JOG そのものにログインするためのアカウントが別なの？ と思われる人もいるかもしれません。それは仕様なんです＾＾;

といっちゃうともともこもないんですけど＾＾ ーー JOG ではホームサイトでのウェブページの管理に Moodle というコースマネージメントシステムを使っているので自然にそうなってしまうんです。

もともと Moodle とは「世界中の教育・研修機関で急激に採用を増やし続けているオープンソースの学習管理システム（LMS）」です（http://moodlejapan.org/home/）。

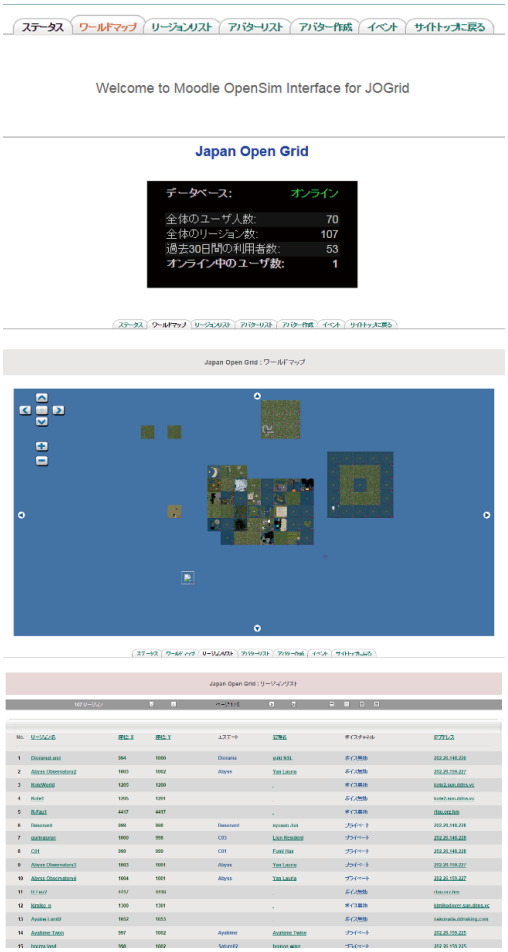
学習管理システムですから本来は OpenSim とはほとんど関係のない分野のアプリケーションです。もともと Moodle は主として学校の教師など教育関係の人たちが学生たちの学習を助けたりコンピューターを通じて学校生活の管理，あるいは成績管理，各種連絡などをスムースに行うためのツールです。

学生の学習管理のためのソフトだから生徒でない部外者が勝手にまぎれこむとややこしくなるのでコミュニティに入るときにいちいちアカウントを要求するように作られているわけです。

というわけで，とりあえず JOG のグリッドにビューアーでログインするための用意はととのいました。つぎは具体的なビューアーでの入り方ですが，せっかく JOG ホームサイトへログイン出来るようになったのでサイトのメニューについてちょっとだけ見ておきましょう。



グリッドそのものに関する情報はこの赤枠の部分に集中しています。じつは Moodle はブロックという単位でウェブページを制作編集するように作られていて，Moodle と OpenSim をむすびつけている Web インターフェイス modlos はブロックとしてこの位置に読み込まれるようになっているわけです。



アバター作成ページをふくめて OSgrid など他のグリッドサービスを利用したことがある人なら慣れ親しんだページが並んでいることがわかるでしょう。

JOG のホームサイトでもうひとつチェックしたほうがいいのはこのコースカテゴリの部分です。



教育支援用アプリケーションである Moodle でコースといえばふつうは学生が履修すべき学習コースですが，JOG の場合グリッド管理スタッフからのさまざまなお知らせやユーザー相互の情報交換などのための各ページにここからアクセスできるようになっています。

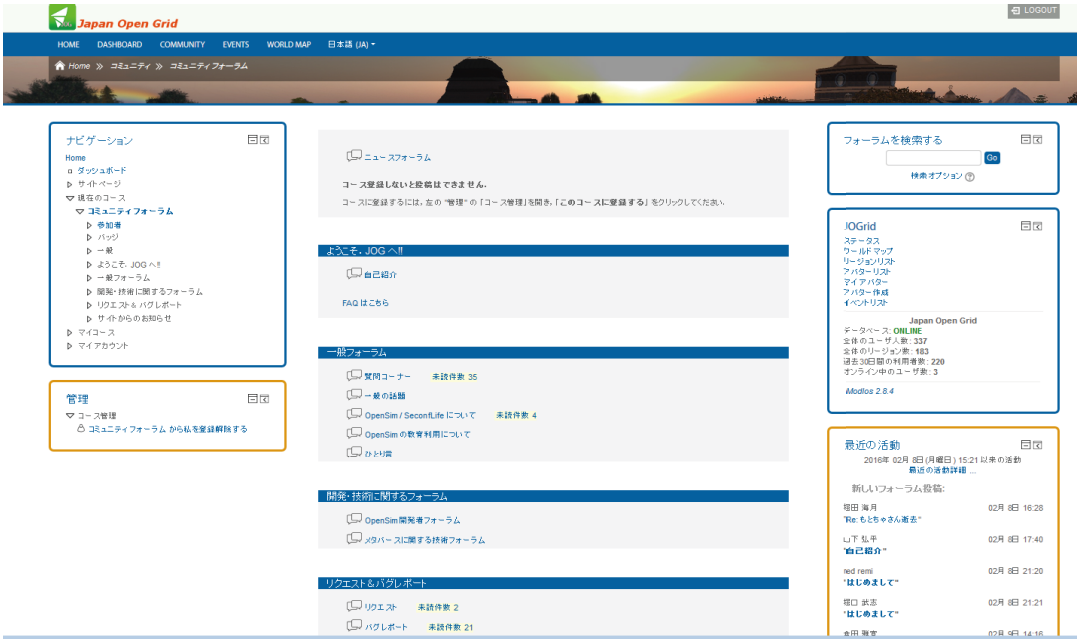
とくにコミュニティから開く「チュートリアル」と「コミュニティフォーラム」にはすくなくとも一度ぐらい目をとおしておいたほうがいいと思います。

利用開始時のとまどいや疑問点はこのチュートリアルを読めばたいてい解消するはずです。

気楽に日本語でフォーラムに読み書きできるグリッドサイトはネット広しといえどもここ JOG だけです＾＾。




Japan Open Grid
LOGOUT
HOME
DASHBOARD
COMMUNITY
EVENTS
WORLD MAP
日本語 (JA)
Home » コース » コミュニティ » チュートリアル
ナビゲーション
Home
ダッシュボード
サイトページ
現在のコース
チュートリアル
参加者
バッジ
一般
最初に
ビューアの設定
応用課
マイコース
マイアカウント
ニュースフォーラム
使用規約
サイト使用規約（2014 11/8 改正）
グリッドサービス使用規約（2014 11/8 改正）
SIM オーナサービス使用規約（2015 11/15 改正）
JOGへのリージョンサーバ接続サービス使用規約（2014 11/8 制定）
最初に
はじめに
アバターの作成方法
FAQ
ビューアの設定
JOG では Firestorm Viewer および Singularity Viewer を推奨ビューアとしています.
JOG へのログイン URL は http://jogrid.net:8002/ です.
JOG へのログイン方法（Firestorm Viewer）
JOG へのログイン方法（Singularity Viewer）
skins.zip: Singularity Viewer 用日本語化ファイル（一部のみ）
展開して, Cドライブの Program Files (x86)\Singularity にコピーしてください.


Japan Open Grid
LOGOUT
HOME
DASHBOARD
COMMUNITY
EVENTS
WORLD MAP
日本語 (JA)
Home » コミュニティ » コミュニティフォーラム
ナビゲーション
Home
ダッシュボード
サイトページ
現在のコース
コミュニティフォーラム
参加者
バッジ
一般
ようこそ、JOG へ!!
一般フォーラム
開発・技術に関するフォーラム
リクエスト& バグレポート
サイトからのお知らせ
マイコース
マイアカウント
管理
コース管理
コミュニティフォーラム から私を登録解除する
ニュースフォーラム
コースに登録しないと投稿はできません.
コースに登録するには, 左の "管理" の「コース管理」を開き,「このコースに登録する」をクリックしてください.
ようこそ、JOG へ!!
自己紹介
FAQ はこちら
一般フォーラム
質問コーナー 未読件数 35
一般の話題
OpenSim / SecondLife について 未読件数 4
OpenSim の教育利用について
ひとり言
開発・技術に関するフォーラム
OpenSim 開発者フォーラム
メタバースに関する技術フォーラム
リクエスト&バグレポート
リクエスト 未読件数 2
バグレポート 未読件数 21
フォーラムを検索する
Go
検索オプション
!OGrid
ステータス
ワールドマップ
リージョンリスト
アバターリスト
マイアバター
アバター作成
イベントリスト
Japan Open Grid
データベース: ONLINE
全体のユーザ人数: 337
全体のリージョン数: 183
過去30日間の利用者数: 220
オンライン中のユーザ数: 3
Modlos 2.8.4
最近の活動
2016年 02月 8日 (月曜日) 15:21 以来の活動
最近の活動詳細 ...
新しいフォーラム投稿:
堀田 海月 02月 8日 16:28
"Re: もとちゃさん逝去"
山下 弘平 02月 8日 17:40
"自己紹介"
red remi 02月 8日 21:20
"はじめまして"
堀口 武志 02月 8日 21:21
"はじめまして"

2.3.2 JOG ー ジャパンオープングリッドへの入り方　その2

【Singularity の場合】

OSgrid のサイトのダウンロードのページで各ビューアーの筆頭にあげられているのが Singularity です。他の各グリッドでもこれを愛用している方は多いでしょう。

http://www.osgrid.org/index.php/downloads

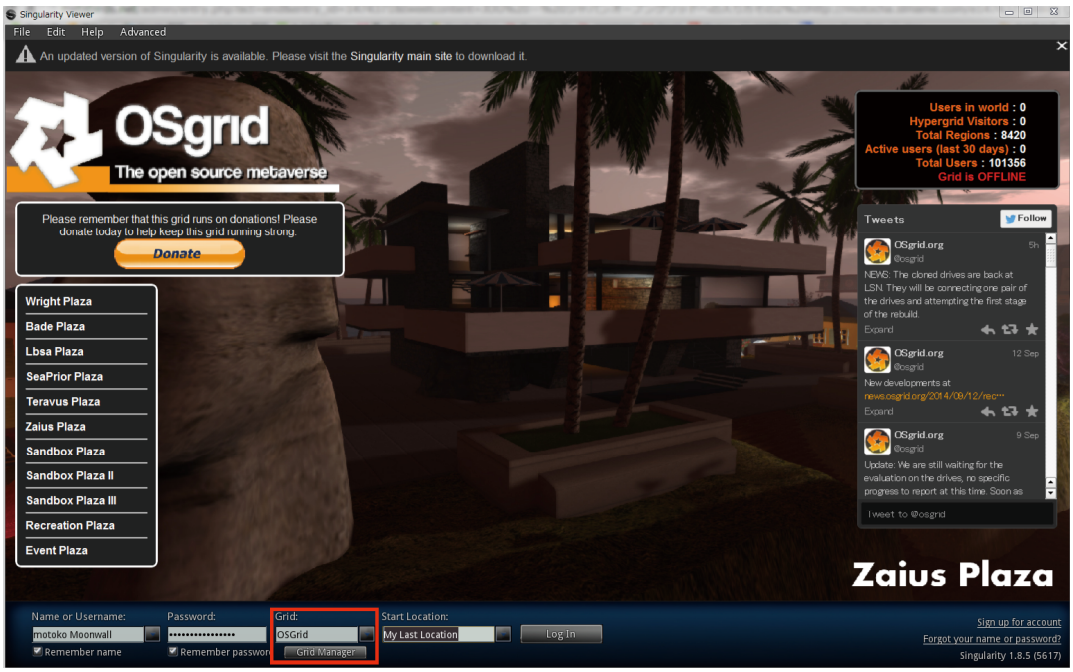
また，もしあなたが SL にいろいろ自作のオブジェクトを持っていて JOG に移植したいと望んでいる場合 XML ファイルを利用することができます。この XML のエクスポート，インポート機能をもっているのは Simgularity ビューアーです。(FireStorm の場合は Collada-Dae を使います)

たちあげると例えば「OSgrid 」にログインする設定になっているかもしれません。

Singulariy(18.6) の場合はこの赤く囲った部分にグリッドを選択する「グリッドリスト」のボックスとそれを編集する「グリッドマネージャー」を開くボタンがあります。

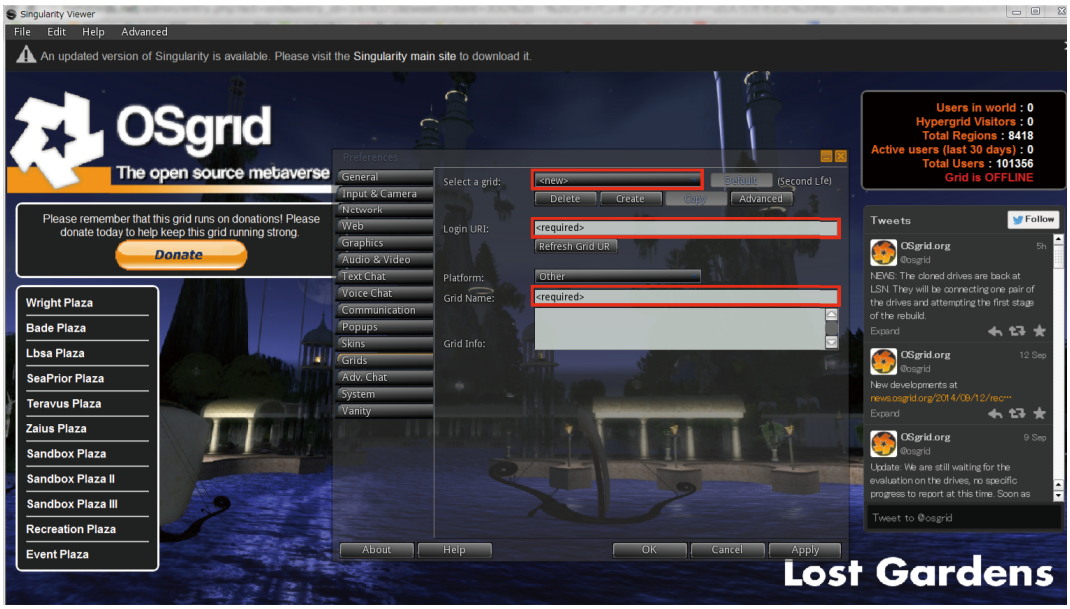
Singularity でないビューアーでもサードパーティのものはここに両者が表示されているかもしれません。あるいはひょっとして表示されていない場合でも「Ctrl + Shift + G キー」のショートカットで両方あるいはグリッドリストだけが表示されるようになる場合もあります。(CoolVL は後者です)

OSgrd や Avination や Inworldz のようなメジャーなグリッドサービスは最初からこのグリッドリストのなかに名前が乗っています。ですからリストから選択するだけでログインできるようになります。



しかし残念ながら今のところメジャーではない JOG はリストに名前がないはずです ^^;

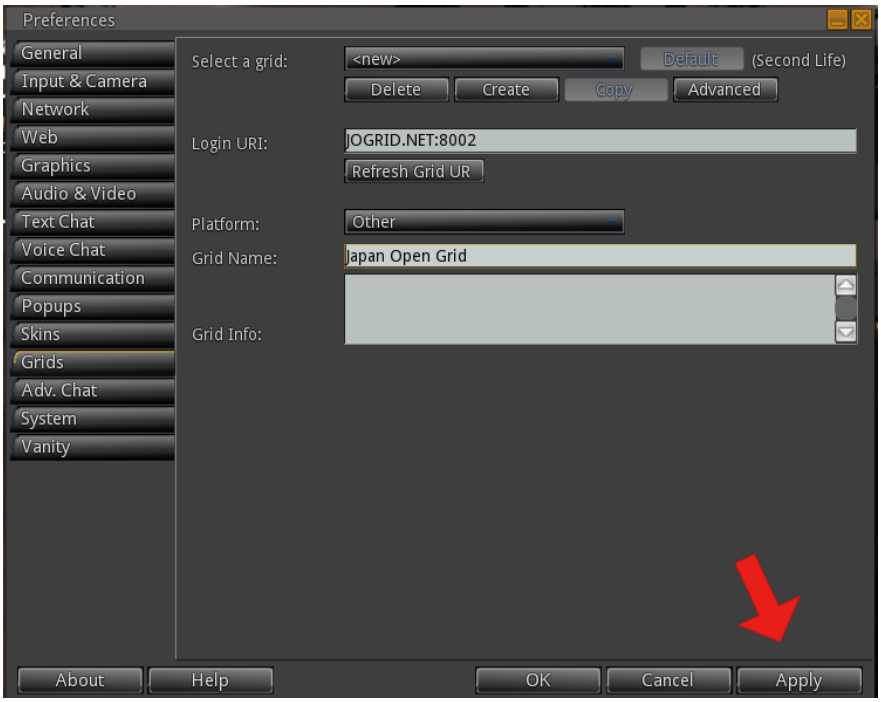
その場合はグリッドマネージャーでリストに JOG の名前を登録しなければなりません。Singularity のグリッドマネージャーボタンをクリックすると下のようなウインドウが開くはずです。



一番上のグリッドの名前のはいっている部分のすぐ下の「Create」を押すとこの画面のようになります。ここで入力しなければならないのは赤枠で囲ってある空白の二カ所です。

上の「Login URL :」とあるところには（JOG サイトのトップページに書かれている）「JOGRID.NET:8002」をいれてください。

下の「Grid Name :」には「Japan Open Grid」といれ,「Apply」を押し「OK」でウインドウを閉じます。



これでリストに Japan Open Grid の名が登録されるのでグリッドを選択してあとはふつうにログインしてください。

【FireStorm の場合】

公式についでSL のビューアーにFireStrom を使ってる人は多いんじゃないでしょうか。FireStrom は OpenSim のグリッドに入るときにも使えます。

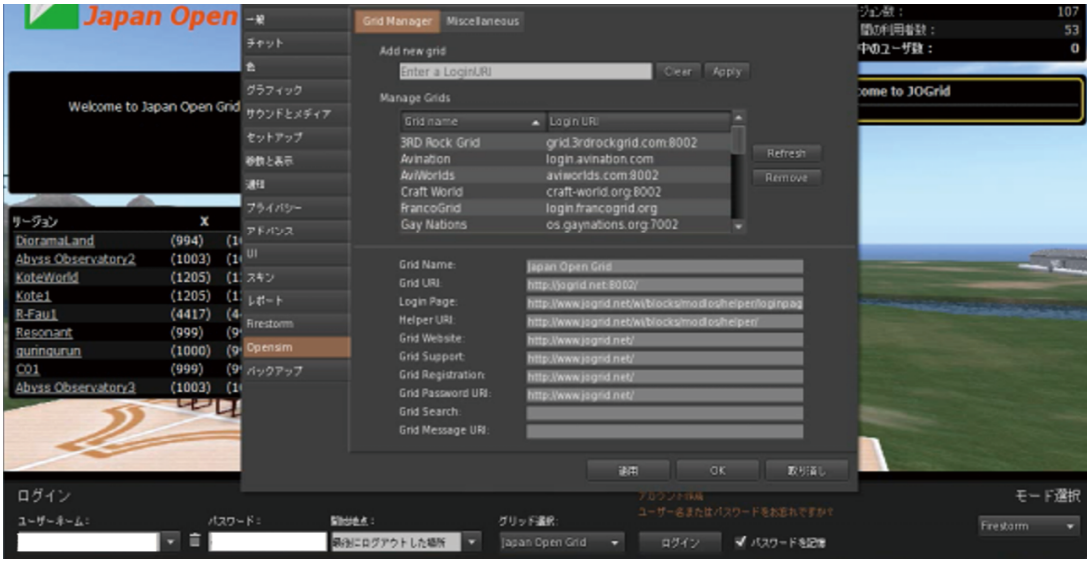
FireStorm は Singularity とちがって「グリッドマネージャー」のボタンがありません。

グリッドの選択は画面左上の「ビューワ」→「環境設定」→「Opensim」から行います。

グリッドマネージャーのウインドウがひらくので一番上の「Add New Grid」のところに「JOGRID.NET:8002」といれて「Apply」ボタンを押します。

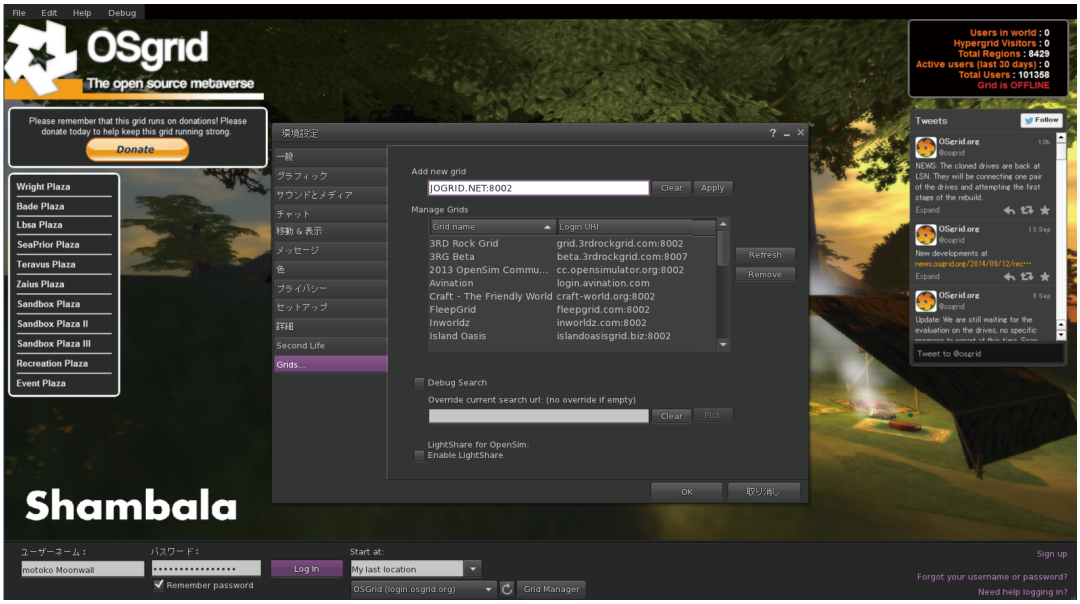
あとはそのままマネージャーウインドウのいちばん下の「適用」つづいて「OK」を押せば設定完了です。





【kokua の場合】

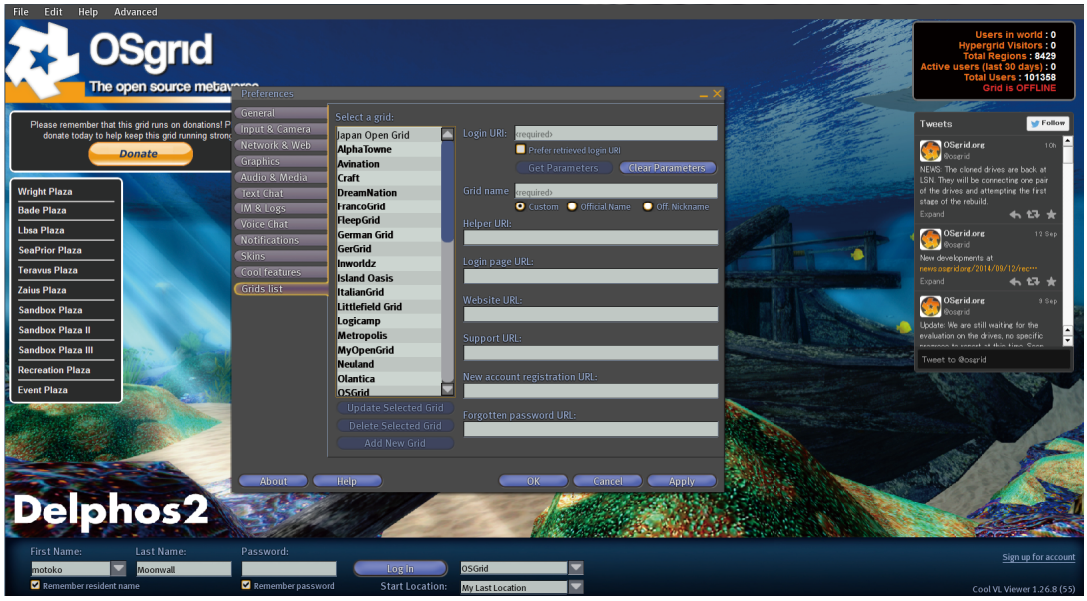
グリッドマネージャーボタンがあるので問題はないと思います。（グリッドリストのほうは Ctrl + Shift + G キーで出たりきえたりするようです＾＾;）



【CoolVL の場合】

使ってる人は少ないかもしれませんがグリッドマネージャーの開き方がちょっと変わっているのであえてとりあげておきます。

立ち上げたときの画面には FireStorm 同様ボタンはありません。Cool の場合は画面上の「Edit → Preferences 」で開く環境設定画面のなかの「Grid list」タブを選択するとグリッドマネージャー画面が開きます。設定そのものは他と同じなので特に問題はないと思います。



motoko Moonwall

3. ハイパーグリッドの紹介

ここでは，Yan Lauria さんによる教育用，または教育に役立つであろうと思われる SIM（JOG の SIM も含む）の紹介を行います.



JOG はハイパーグリッドモードで動いているよ！
このモードだと，他の OpenSim の世界（グリッド）に
そのままテレポートすることもできるんだよ！



すごい！
世界中の OpenSim と繋がっているんだね！



この章は Yan さんによる教育用の SIM（リージョン）の紹介だよ.
色んな SIM があるんだね.



やっぱり銀英伝はヤンが死んじゃった
後は後日談だよな～



何の話ですかー！

3.1 ハイパーグリッドの巡り方

3.1.1 グリッドのアドレス

OpenSim にはハイパーグリッドモードと呼ばれるモードがあり，このモードで動作している OpenSim 同士はお互いのグリッド間を同一のアバターで行き来することができます．ここでグリッドとは，それぞれの OpenSim のシステムが管理する「世界」のことを指します．

今いるグリッドから他のグリッド（の SIM）へ移動する場合は，移動先のグリッド（の SIM）のアドレスが必要になります．グリッド（の SIM）のアドレスは以下のように表記します．

サーバ名：ポート番号［：SIM 名］　　または

http:// サーバ名：ポート番号［/SIM 名］

ここで［ ］は省略可能を表します．もし SIM 名が分からない場合は ：SIM 名（/SIM 名） は省略できます．SIM 名を省略した場合は，そのグリッド内で予めデフォルトに設定されている SIM を指定したことになります．

例えば，次ページの 3.2.1 の The city of math の SIM のアドレスは以下のようになります．

oraft-world.org:8002:Mathland　　または

http://oraft-world.org:8002/Mathland

なお，3.2 節の紹介ページでの SIM 名 Mathland の後に続く（ ） は，その SIM 内での X,Y,Z 座標を表します（アドレスの指定時には SIM 内の座標は無視しても OK です）．

3.1.2 他のグリッドへのテレポート

実際に他のグリッドの SIM にテレポートする場合は，上記のアドレスを世界地図ウィンドウの検索の部分に入力し，検索を実行します（図 3.1.1 の黄緑の矢印）．もし検索対象の SIM が存在する場合はマップ表示部分に赤い〇が表示されます（図 3.1.1 の黄色の矢印）．この時，実際のテレポート先のマップは表示されませんので注意してください．

図3.1.1の状態で,テレポートボタンをクリックすれば,検索したグリッドにテレポートできます！

それでは，ハイパーグリッドの世界へレッツゴー！！

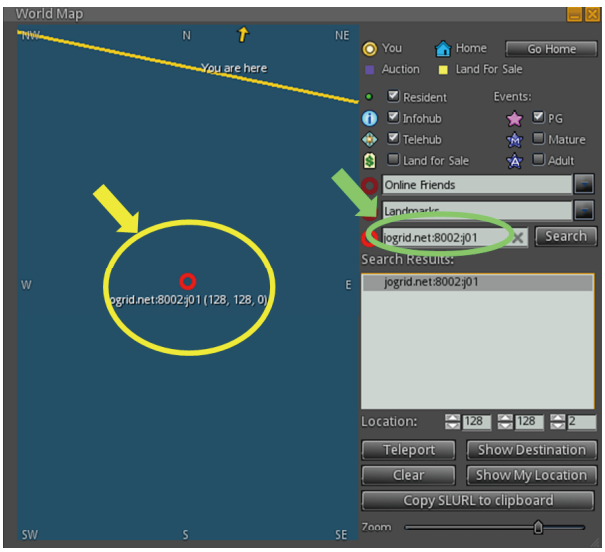


図 3.1.1　他のグリッドへのテレポート

3.2 教育用 SIM の紹介（by Yan Lauria）

3.2.1 The city of math, Mathland@Craft World

グリッド： craft-world.org:8002
SIM 名： Mathland (126,164,49)



イタリアの Michelle Tech による幾何学シム:等分、並行、面積、ピタゴラスの定理、回転体、体積。説明はイタリア語。

Virtual Science： http://www.virtualscience.it/mathland.html

3.2.2 Edu3d@Craft World

グリッド： craft-world.org:8002
SIM 名： Edu3d (104,131,24), Scriptlandia (97,127,23)



基本プリムを使ったアートなオブジェクト。シンガポールの Gardens by the bay の Supertree に似ている。パーティクル（ビーム、降雪、炎、泡、湯気、雨など)、動くテクステャー（スクロール、拡大・縮小、回転など)、サウンドほかのスクリプトのフリーライブラリもある。

3.2.3 DigiLandia@Craft World

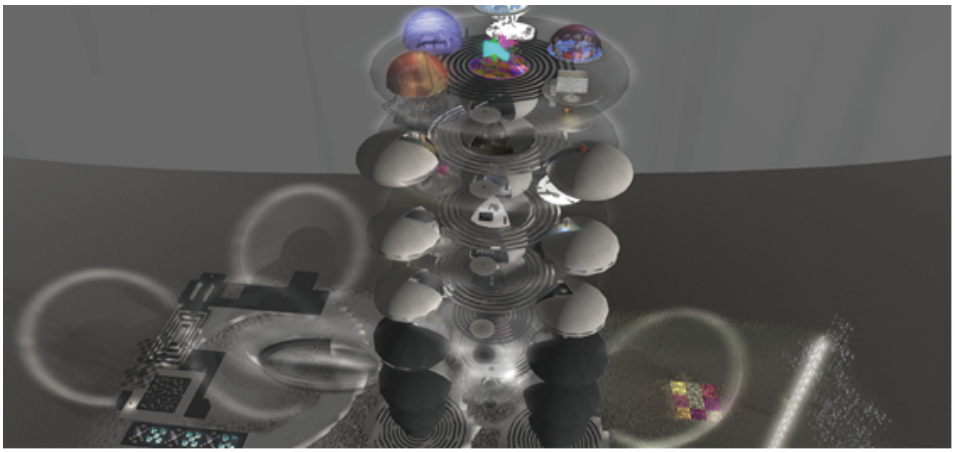
グリッド： craft-world.org:8002
SIM名： DigiLandia (155,146,22)： Michelangelo



Michelangelo Tricarico の作品。イタリアの歴史的建築物であるピサの斜塔、サン・ピエトロ大聖堂、ミラノのドゥオーモ。パリのエッフェル塔もある。中までは再現されていない。

3.2.4 Museo del Metaverso@Craft World

グリッド： craft-world.org:8002
SIM名： MdM (118,139,404)



現代アート美術館。

Museo del Metaverso Project Documentation： http://cargocollective.com/elifayiter_MdM
Museo del Metaverso： http://www.museodelmetaverso.org/

3.2.5 Micrographia/ 顕微鏡図譜 @Kitely

グリッド：grid.kitely.com:8002

SIM名：Micrographia (26,101,620)

Micrographia (178,223,22)：Microscope Gallery

Micrographia (159,207,22)：Yersinia pestism



Graham Mills による第一級の科学展示。弾性に関するフックの法則で知られるロバート・フックが 1665 年に出版した「顕微鏡図譜」には顕微鏡で観察したノミの精密なスケッチが載っている。その出版年にネズミにたかったノミが媒介したペストがロンドンで大流行し、約７万人が死亡。その翌年にロンドン大火が起きてペスト流行が終焉し、都市の再建にフックが活躍している。その偶然の一致に着目した博物館。

Kitely：https://www.kitely.com/virtual-world/Graham-Mills-2/Micrographia

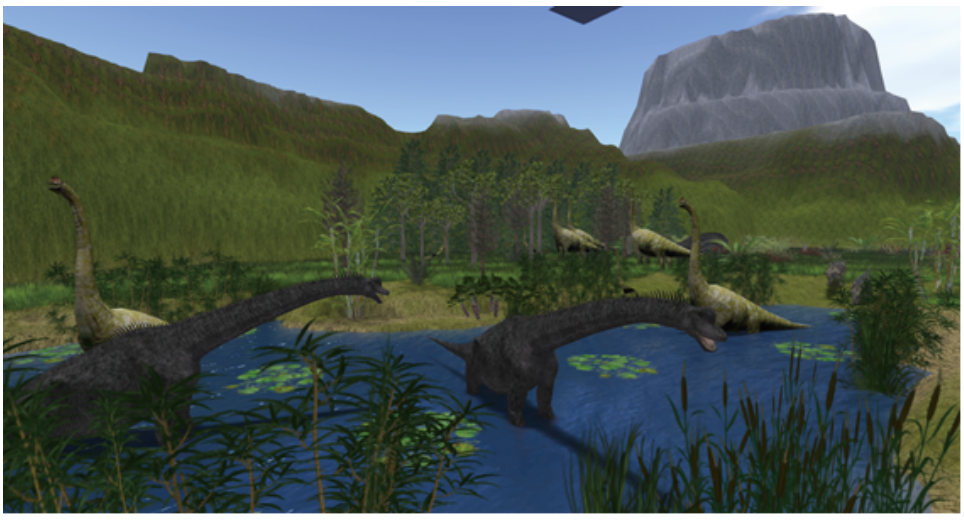
Virtual Outworlding：

http://virtualoutworlding.blogspot.co.uk/2015/06/2015-edu-micrographia-in-virtual-world.html

3.2.6 Jurassic Park@FrancoGrid

グリッド：hg.francogrid.org:80

SIM名：Jurassic Park (178,140,6)



Imperator Janus の作品。映画の舞台となった Isla Sorna 島を 1276m×1276m もの VarRegion で再現している。直接ゲートに TP できず、飛行も禁止されてるが、海中の着地点からカメラ移動でダブルクリック・テレポート（要設定）するといい。静態展示だがクオリティーは高い。

3.2.7 Moya 美術館 @FrancoGrid

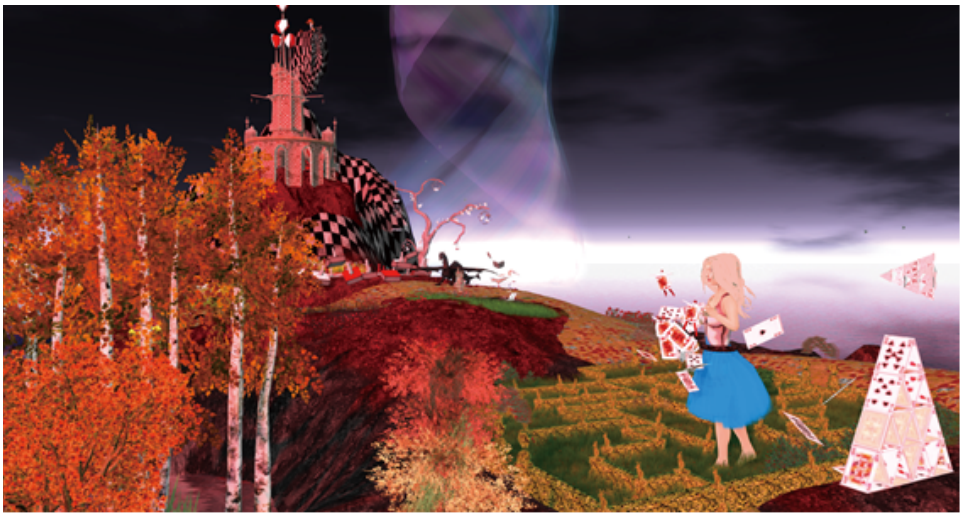
グリッド： hg.francogrid.org:80
SIM 名： moya (114, 122, 118)



Second Life にもあるフランス現代アートのアーティスト Patrick Moya による美術館。
Patrick Moya： http://www.moyapatrick.com/

3.2.8 Cherry Manga@FrancoGrid

グリッド： hg.francogrid.org:80
SIM 名： levillage (114,122,118), Terra-Mater (128,128,21), Aliceinwonderland (131,106,825)



Cherry Manga の高品質メッシュによるアート・インスタレーション。「不思議の国のアリス」では祠の穴を落下し、マウスポインタが椅子マークを表示する箇所をクリックしていくと物語が進行する。
YouTube： https://www.youtube.com/user/CherryMangaSL

3.2.9 Archaeology@University of StAndrews (Apollo Grid)

グリッド： apollo.cs.st-andrews.ac.uk:8002

SIM名： Partal: Central Hub (127,148,22), Cathedral 1 (128,128,40)



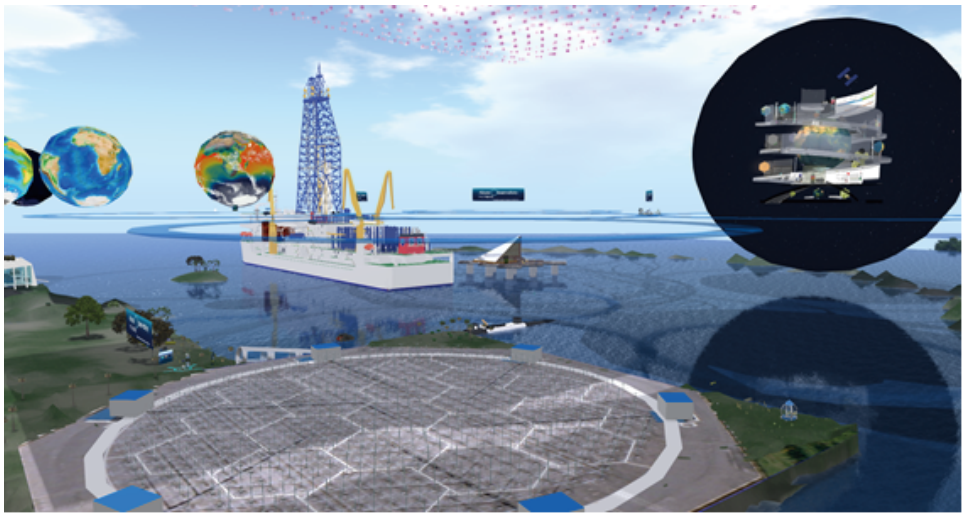
セント・アンドリュー大学によるセント・アンドリュー大聖堂（スコットランドのローマ・カトリック教会、1318年）の復元。

St Andrews Cathedral：
https://www.st-andrews.ac.uk/classics/research/projects/church-space/stas-cathedral/

3.2.10 Abyss Observatory/ アビス海文台 @Japan Open Grid

グリッド： jogrid.net:8002

SIM名： Abyss Observatory (190, 191, 141): Portal
Abyss Observatory (132, 466, 105): MU Radar
Abyss Observatory (493, 305, 139): Drill Ship
Abyss Observatory (386, 478, 107): Earth Simulator
Abyss Observatory (124, 116, 158): Earth & Life Evolution

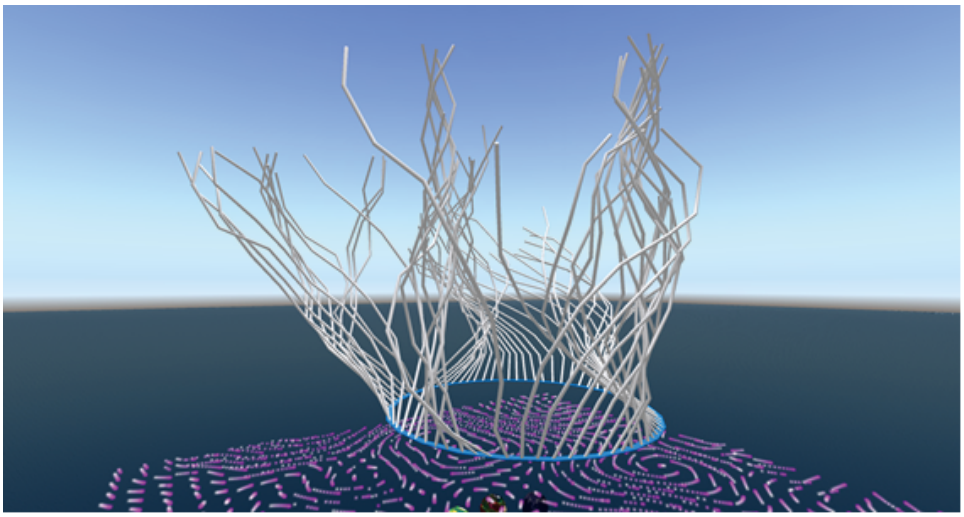


JAMSTECと東京情報大学の協力。地球深部探査船「ちきゅう」、地球シミュレータ（第三世代、NEC SX-ACE)、京都大学MUレーダー、地球と生命の進化、近代博物館。

3.2.11 "Wind Observatory" and "Wind Circle"/「風の展望台」と「風の円」@Japan Open Grid

グリッド： jogrid.net:8002

SIM 名： Abyss Observatory (9,9,401)



Jos Stan の２次元非圧縮性流体のリアルタイムシミュレーションを東京情報大学の井関研究室が VarRegion に実装。インワールドで強制力を加えたり流体のパラメータを変更すると、その場で可視化される。この風の場を用いた comet Morigi によるアート・インスタレーション。

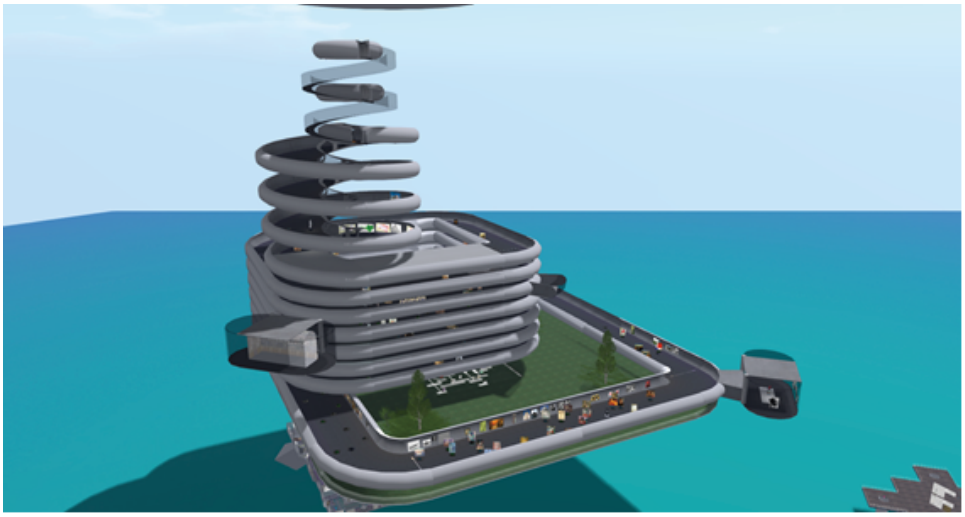
YouTube： https://www.youtube.com/watch?v=yfppzAzG3LO

3.2.12 The Modern Museum/ 近代博物館 @Japan Open Grid

グリッド： jogrid.net:8002

SIM 名： Abyss Observatory (381,127,3591): MM-I

Abyss Observatory (446,189,3785): MM-II



Hajime Nishimura、comet Morigi、motoko Moonwall による。絵画、彫刻、建築、音楽、科学技術工学上の発見・発明、社会経済イベントなどを時系列的に並べ、分野を越えた関係を発見するための協働プラットフォームとして作られた。

3.2.13 Paradise Island@Japan Open Grid

グリッド： jogrid.net:8002

SIM名： Paradise Island (59, 145, 21)



Tarsha による自然動物園。偶蹄目（ウシ、イノシシ、キリン、トナカイ、ヒツジ、ブタ、バッファロー)、ネコ科（ライオン、トラ)、ゾウ、クマ、イヌ、ウサギ、シマウマ、ウマ、ゴリラ、鳥類（カモメ、コクチョウ、ハクチョウ、コウノトリ)、海棲哺乳類（アザラシ、シャチ）がいる。

3.2.14 Le Voyage De Chihiro/Spirited Away/ 千と千尋の神隠し @Japan Open Grid

グリッド： jogrid.net:8002

SIM名： sen001 (148,235,33): Start point

sen003 (135,248,44): Abra-ya



DAIKI Aabye による教育目的のシム。3x3 の Varregion シムを 3 個使用し、「千と千尋の神隠し」の世界を全編にわたって忠実に再現している。

3.2.15 Shibuya and Osaka/ 渋谷と心斎橋 @Japan Open Grid

グリッド： jogrid.net:8002

SIM 名： Shibuya (49,196,22), Little_Osaka (108,108,24)



Kote2 による。外国人観光客に人気の高い渋谷の交差点と、グリコの看板で有名な心斎橋の道頓堀川に掛かるえびす橋。

3.2.16 Four Season Botanical Garden/ 四季の植物園 @Japan Open Grid

グリッド： jogrid.net:8002

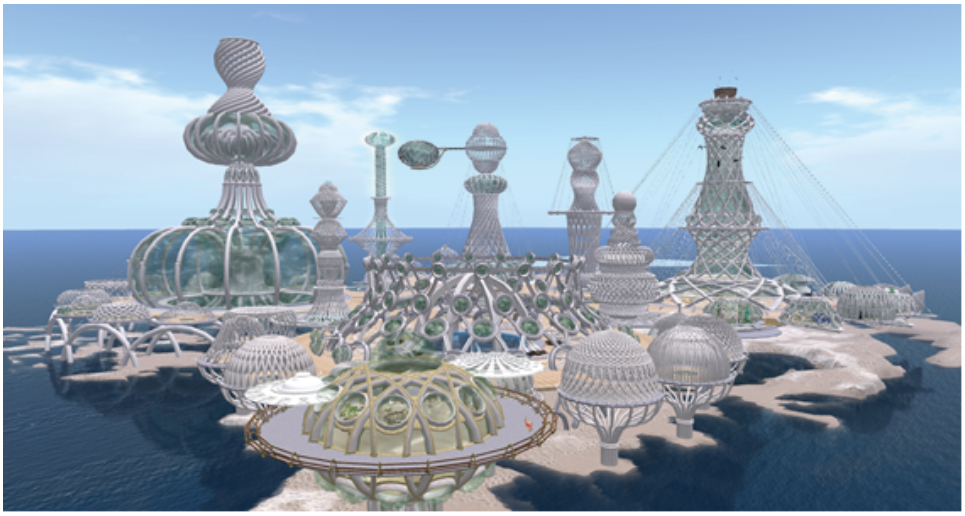
SIM 名： tim Land (48,169,21), vushnue organ (373,504,21)



timou Nishi による植物園。512mx512m の VarRegion を４分割して春夏秋冬を表現。春：サクラ、カンツバキ、ガマ、タケ、クロマツ、ナンテンほか、夏：ヤシ、サルスベリ、ノウゼンカズラ、co-coloco flower ほか、秋：カエデ、冬：シラカバほか。

3.2.17 Ivory City of Prims@Sanctuary

グリッド： sanctuary.homelinux.org:8012
SIM名： Ivory City 1 (41,127,42)



最近亡くなられたLumiere Noirによるプリムのチュートリアル。彼のパートナー Tosha Tyranが引き継いでいる。Second LifeのIvory Towerを圧倒する壮大な造形。

3.2.18 The Spirit of Arcadia@Sanctuary

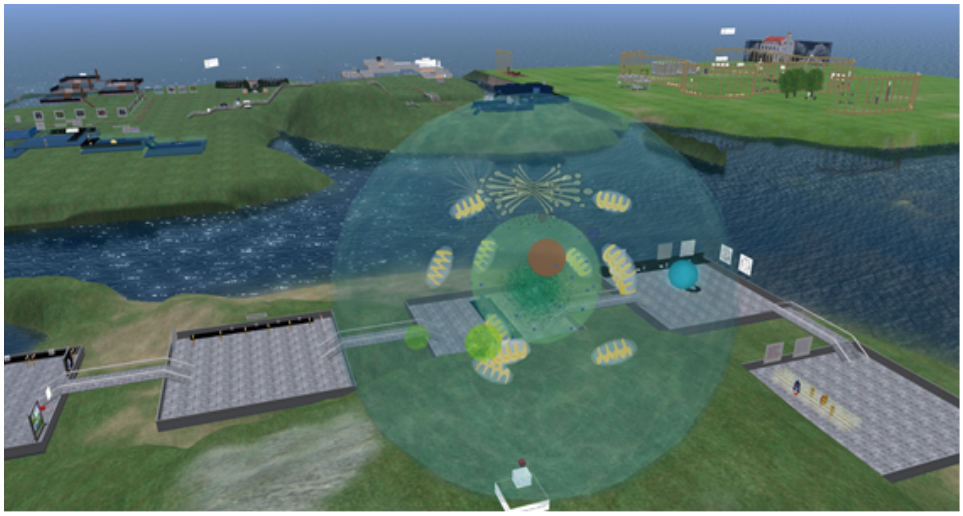
グリッド： sanctuary.homelinux.org:8012
SIM名： TSoA 11 (128,128,1)： Nemo's Underwater Lab@Sanctuary,
TSoA 1 (126,117,32)： Privateer Space
TSoA 9 (65,52,34)： Slums



Arcadia Ashylum/Aleyがフルパーミッションで提供してきた膨大なスカルプ作品のほぼすべてが展示されている。全部で建設途中のものも含めて１６シムある。

3.2.19 GENOME Island@VIBE

グリッド： genome.bio-se.info:9000
SIM名： Huxley (113,130,40)



テキサス・ウェズリアン大学の Mary Anne Clark が Second Life で作った GENOME Island の OpenSim 版。Huxle を到着点とし、Mendel（メンデルの法則）、Hooke（細胞）、Crick（ゲノム）の 3 リージョンに展示されている。膨大な数のノートカード。ノートカードのためのタッチポイントの統一表示が欲しいところ。宝探し的な効果もあると思うが。

Virtual Outworlding：
http://virtualoutworlding.blogspot.jp/2014/10/2014-edu-genome-islands-on-opensim.html

3.2.20 Nova Archaeology@VIBE

グリッド： genome.bio-se.info:9000
SIM名： Nova (150,48,24): Late Roman Village
Nova (44, 97,29): Maya
Altun Ha (174,233,21): Native American Cultural Periods
Allegheny (94,133,24): Lemon House



ペンシルベニア州立インディアナ大学出身の Nova Saunders をオーナーとする考古学シム。

The Hyperzette： http://the-hyperzette.blogspot.jp/2015/01/destination-education-nova.html

3.2.21 Africa Traveler@VIBE

グリッド： africatraveler.bio-se.info:9000
SIM名： AfricaTraveler (21,55,22)



西アフリカ旅行の際に必要となる風土病対策と異文化コミュニケーション・スキルを学ぶシム。
GlobalHealth3D Grid： https://sites.google.com/site/globalhealth3dgrid/

3.2.22 Pompeii@Metropolis

グリッド： hypergrid.org:8002
SIM名： Pompeii (111,175,28)



ヴェスヴィオ火山の大噴火による火砕流に埋もれたポンペイの復元。現在、２シムが公開されているが、最終的には計１６シムとなる予定で、OpenSimならではの大規模なもの。大学のプロジェクトではなく、Pixel Primほかボランティアベースなのも OpenSimらしい。
The project Pompeii reconstruction: https://hypergrid.org/metropolis/wiki/en/index.php?title=Pompeji

3.2.23 Gerry Anderson's World@AiLand

グリッド： ai.vue.ed.ac.uk:8002

SIM名： Black Rock (128,128,22): Supercar
Space City (93,116,22): Fireball XL-5
Marineville (46,9,21): Stingray



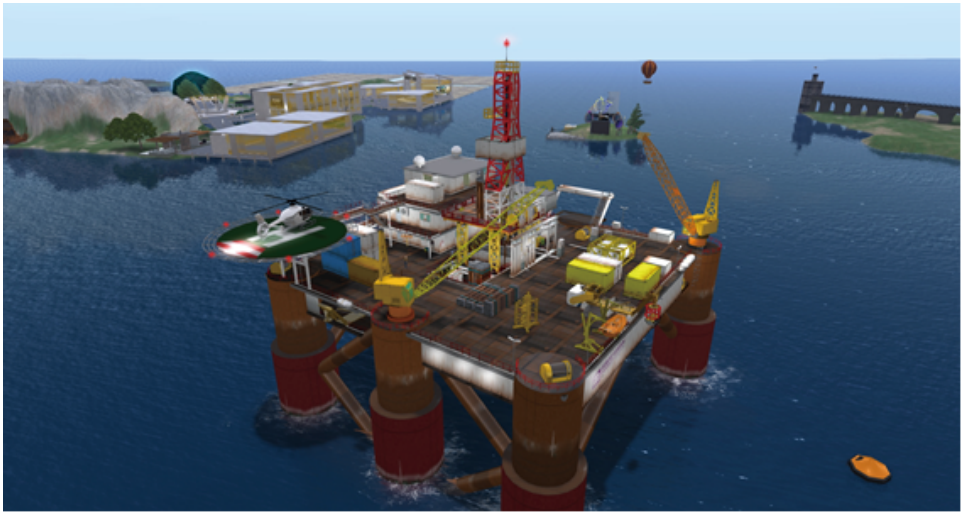
Ai Austinによる。ジェリー・アンダーソンによるスーパーマリオネーションの「スーパーカー」(1961)、「宇宙船 XL-5」(1962)、「海底大戦争 スティングレイ」(1964) の世界。

Ai Austin： http://www.aiai.ed.ac.uk/~ai/

3.2.24 Oil Rig@AiLand

グリッド： ai.vue.ed.ac.uk:8002

SIM名： Oil Rig (122,113,42)



Ai Austinによる非常に良くできたセミサブ型石油掘削リグ。掘削系では掘削やぐら、パワースイベル、ロータリーテーブル、ドローワークス、ドリラーズハウス（ドッグハウス）が、泥水系ではマッドポンプ、マッドタンク、ダイバーター、シェルシェーカー（掘削くず分離器)、チョーク＆キル・マニホールド、暴噴防止装置が、支援系ではアジマス・スラスター、遠隔海中作業ロボット、

ヘリポートがある。見当たらないのはライザー管、クラウンブロック(固定滑車)、トラベリングブロック（動滑車)、Crown-mounted Motion Compensator（ドリルストリングの上下動吸収装置)、ライザーテンショナーなど。

Yan Lauria

技術編

4. OpenSim のインストール

ここでは WindowsおよびLinux(CentOS6)への OpenSimのインストールの仕方を紹介します.なお,この章には前作「OpenSim サーバ構築入門」の焼き直し部分がかなりあります. また記述を省いている箇所もありますが,その部分は前作とあまり変わりがありませんので,是非,前作「OpenSim サーバ構築入門」も併せてご覧ください.

Windows ではバイナリコードをダウンロードして,そのまま起動ささせます. Windows ではグリッドモードの場合もデータベースに SQLite3 を使用し, さらに他のグリッドサーバ (Robust サーバ) に接続することを前提にしますので, 特に難しいことは何もありません.

一方 Linux では一部プログラムをコンパイルします(その方が最新のプログラムを使用できます). 具体的には, まず Cairo と libgdiplus, 次に mono をソースからコンパイルし, 続いて, OpenSim もソースコードからコンパイルします. CentOS6 を前提としていますが, 他の Linux ディストリビューションでもほぼ同じです. その後, tmux を使用した連続運転の手法も紹介します (前作では screen を使用していました).



やった！Windows 版も本章入りだね！



って, ほとんど書いてないじゃない !!



ううっ. すまん！Windows は実際そんなに書くネタがないだよ.
コンパイルもやらないし (やりたいけど ...). あっ, 7章も追加したよ.



先生って, 謝っているときも
表情変わらないね.



実は中身はロボットだったりして !!
オールドマンと同じようなシステムだという情報も ...

http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?Fumi.Hax#k11de5e4

4.1 Winows へのインストールとスタンドアローンモードでの起動

4.1.1 ファイルのダウンロード

プログラムのダウンロードページは http://opensimulator.org/ です．直接ダウンロードしたい場合は http://dist.opensimulator.org/ からダウンロードします．Windows で使用する場合は，拡張子が zip のものを選択してください．2016 年 3 月 15 日現在の最新バージョンのバイナリファイルは http://dist.opensimulator.org/opensim-0.8.2.1.zip です．

ダウンロードしたファイルを適当なディレクトリ（何処でも可）で解凍します．

4.1.2 スタンドアロンモードでの起動

バイナリファイルをダウンロードした場合，OpenSim を起動するための設定ファイルは，ほとんど既に用意されています．後は，bin¥OpenSim.exe をダブルクリックするだけで OpenSim がスタンドアロンモードで起動します．

4.1.3 初期設定

bin¥OpenSim.exe をダブルクリックすると，コンソール画面のウィンドウが表示され，OpenSim が起動します．通常起動後に 1 回だけ図 4.1.1 のようなファイアウォールに関するダイアログが表示されますが，この場合は「アクセスを許可する」をクリックします．

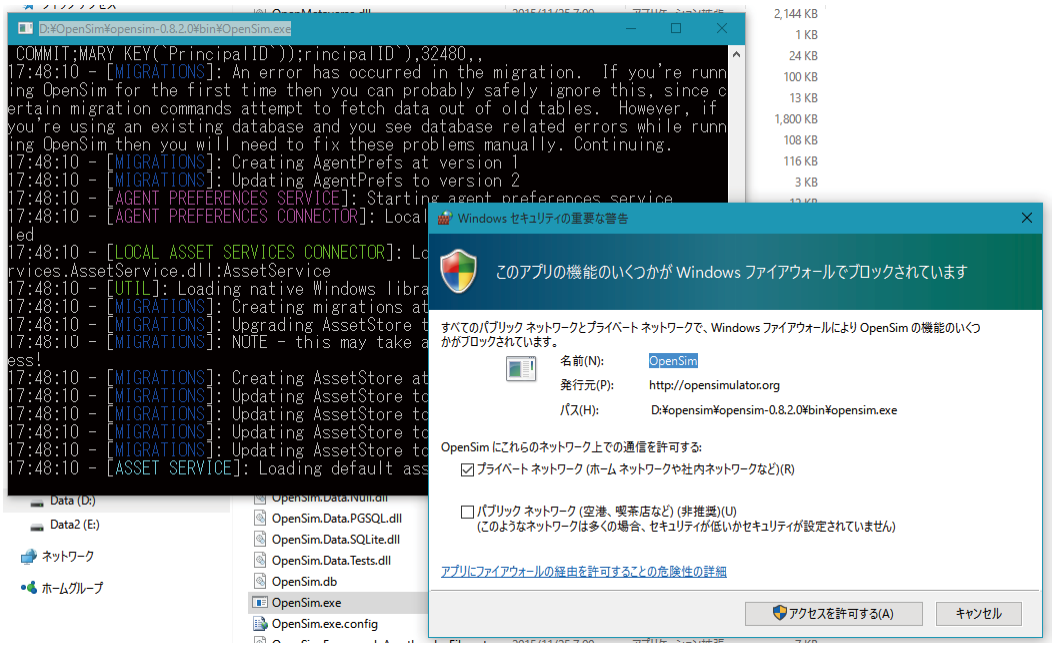


図 4.1.1 ファイアウォールに関するダイアログ

OpenSim のコンソール画面では，リージョン（SIM）の名前を入力するプロンプトが表示されますので，リージョンの名前を入力します（必須）．その後，RegionUUD，Location（位置座標），IP アドレス，ポートの設定（２種類）およびホスト名等を訪ねられますが，全てデフォルト（そのままエンターキー）で OK です（図 4.1.2）．

続いてエステート名を訪ねられますが，これもデフォルト（そのままエンターキー）で OK です．

最後に，アバター作成のための，アバター名（ファーストネームとラストネーム），パスワード（必須），メールアドレス，識別用の UUID も訪ねられます．アバター名とパスワードを入力し（パスワー

ドは画面に表示されない)，メールアドレス，UUID はデフォルトのままエンターキーを押します（図 4.1.3)．

OpenSim が正常に起動すれば，図 4.1.4 のような OpenSim のコマンドプロンプト（コマンド入力要求）が表示されます．



図 4.1.2 リージョンの設定画面



図 4.1.3 アバターの設定画面



図 4.1.4 OpenSim のコマンドプロンプト

4.1.4 Local サーバへのログイン

OpenSim を起動したPCと同じPCから OpenSim にログインする場合は,ビューアのグリッドマネージャで,LocalHost を選択し，先に設定したアバターのファーストネームとラストネームおよびパスワードを指定して OpsnSim にログインします．

上手くいけば,図 4.1.5 のようなピンヘッドの島（別名,たこ焼き島）にデフォルトアバター（ルース）としてログインすることができます．

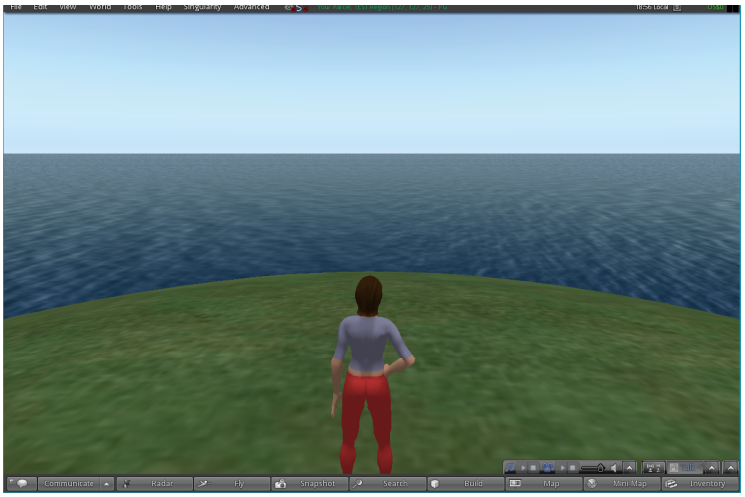


図 4.1.5 OpenSim への初めてのログイン

4.1.5 Local サーバの停止

OpenSim サーバを停止させる場合は，必ず quit または shutdown コマンドを用います．いきなりコンソールウィンドウを閉じると，変更した設定や作成したオブジェクトがサーバに保存されないことがあります．注意してください．

4.2 Linux へのインストール

4.2.1 開発環境のインストール

ここでは CentOS 自体のインストール方法は省きますが，インストールの仕方によっては開発環境幾がインストールされない場合があります．CentOS の場合は開発環境のインストール用スクリプトが http://www.nsl.tuis.ac.jp/DownLoad/SoftWare/Linux/centos-devel.sh に用意されていますので，それを利用して開発環境のインストールを行います（図 4.2.1）．他のディストリビューションについては，別途それぞれの説明サイトなどを参考にしてインストールしてください．

なおこの CentOS 用のスクリプトは， yum を使用する他の Linux ディストリビューションでも使用できる可能性があります （実行しても，パッケージがインストールされないということ以外の問題は発生しません）．

```
# wget http://www.nsl.tuis.ac.jp/DownLoad/SoftWare/Linux/centos-devel.sh
# bash centos-devel.sh
```

図 4.2.1 CentOS での開発環境のインストール

4.2.2 pkgconfig のライブラリディレクトリ

幾つかのソフトウェアではインストールの状態をシステムに通知するために pkgconfig を使用します．CentOS にインストールされている pkgconfig では /usr/lib/pkgconfig（または /usr/lib64/pkgconfig）に各ソフトウェアの情報を保存しますが，ソースコードからコンパイルしてインストールした場合には，/usr/local/lib/pkgconfig にインストール情報が保存される場合があります．

この不整合を解消するために，ソフトウェアをインストールする前に /usr/local/lib/pkgconfig から /usr/lib/pkgconfig にシンボリックリンクを張り，ソースコードからコンパイルした場合でもインストール情報が /usr/lib/pkgconfig に保存されるようにします（図 4.2.2）．

```
# ln -s /usr/lib/pkgconfig /usr/local/lib
```

図 4.2.2 pkgconfig へのシンボリックリンク

4.2.3 Mono のインストール

Mono は Linux/Unix 上で動作する MS .NET の実行環境です．MS Windows 用の C# などの中間コードを Unix/Linux 上で動作させる事が可能であり，OpenSim を Linux 上で実行されることができます．

Mono は各ディストリビューションで提供しているものを使用しても構いませんが，可能ならば最新バージョンをソースコードからコンパイルして使用した方が良いでしょう．

なお，Mono をコンパイルするには，Cairo，libgdiplus が必要です．前回では Glib2 も必要でしたが，今回は必要ありません．

【Cairo】

Cairo は 2D のグラフィックライブラリで libgdiplus で必要となります．図 4.2.3 に従って，コンパイルインストールを行います．2016 年 3 月 15 日現在の最新の安定バージョンのソースコードファイルは cairo-1.14.6.tar.xz です．

サイト URL:　　http://www.cairographics.org/
ダウンロード URL: http://www.cairographics.org/releases/

```
# wget www.cairographics.org/releases/cairo-1.14.6.tar.xz
# xzcat cairo-1.14.6.tar.xz |tar xfv -
# cd cairo-1.14.6
# ./configure
# make
# make install
# ldconfig
```

図 4.2.3 cairo のインストール手順 （cairo-1.14.6 の場合）

【libgdiplus】

libgdiplus は GDI(Graphic Device Interface: MS Windows のグラフィック用 API) の拡張ライブラリであり，Mono で必要となります.

図 4.2.4 に従ってコンパイル，インストールを行います. インストールコマンド終了後，/usr/local/lib にライブラリがインストールされます.

2016 年 3 月 15 日現在の最新バージョンのソースファイルは libgdiplus-4.2.tar.gz です.

サイト URL:　　http://www.mono-project.com/docs/gui/libgdiplus/
ダウンロード URL: http://download.mono-project.com/sources/libgdiplus/

```
# wget download.mono-project.com/sources/libgdiplus/libgdiplus-4.2.tar.gz
# tar zxfv libgdiplus-4.2.tar.gz
# cd libgdiplus-4.2
# ./configure
# make
# make install
# ldconfig
```

図 4.2.4 libgdiplus のインストール手順 （libgdiplus-4.2 の場合）

【Mono】

Cairo, lingdiplus をインストール後，Mono を図 4.2.5 に従ってインストールします. Mono のコマンドは /usr/local/bin に，ライブラリ関連のファイルは /usr/local/lib または /usr/local/lib/mono などにインストールされます. また，Mono のソフトウェア情報は /usr/lib/pkgconfig に保存されます. なお，Mono のコンパイルには非常に時間がかかります.

2016 年 3 月 15 日現在の最新バージョンのソースファイルは mono-4.3.2.467.tar.bz2 です.

サイト URL:　　http://www.mono-project.com/
ダウンロード URL: http://download.mono-project.com/sources/mono/

```
# wget download.mono-project.com/sources/mono/mono-4.3.2.467.tar.bz2
# tar jxfv mono-4.3.2.467.tar.bz2
# cd mono-4.3.2
# ./configure --with-libgdiplus=/usr/local/lib/libgdiplus.la
# make
# make install
```

図 4.2.5 Mono のインストール手順 （mono-4.3.2.467 の場合）

4.2.4 OpenSim のコンパイルとインストール

2016 年 3 月 15 日現在の OpenSim の最新リリースバージョンは 0.8.2.1 です．ソースファイルのダウンロード URL は以下の通りです．

```
http://opensimulator.org/dist/opensim-0.8.2.1-source.tar.gz     または
http://opensimulator.org/dist/opensim-0.8.2.1-source.zip
```

なお，図 4.2.6 のコマンド例で opensim-0.8.2.1-source を opensim に変更している理由は以後の説明上の都合によるもので，必ずしもディレクトリ名を変更する必要はありません．また展開するディレクトリも /usr/local 以外でもかまいません．ただし，以後の説明では OpenSim は /usr/local/opensim に展開されているものとします．

OpenSim のコンパイルでは以前は nant を使用していましたが，現在は **xbuild** を使用ます（図 4.2.7）．一度コンパイルして，再度コンパイルし直すには，xbuild を実行する前に **xbuild /target:CLean** を実行します．

```
# cd /usr/local
# wget opensimulator.org/dist/opensim-0.8.2.1-source.tar.gz
# tar zfxv opensim-0.8.2.1-source.tar.gz
# mv opensim-0.8.2.1-source opensim
# rm opensim-0.8.2.1-source.tar.gz
```

図 4.2.6 OpenSim のソースコードの展開（opensim-0.8.2.0-source.tar.gz の場合）

```
# cd /usr/local/opensim
# bash runprebuild.sh vs2010
# xbuild
```

図 4.2.7 OpenSim のコンパイル

4.2.5 tmux コマンド

前回 OpenSim をバックグラウンドで動かすツールとして screen コマンドを紹介しました．今回は **tmux** を紹介します．tmux は screen と同様に仮想コンソールを形成して，OpenSim をバックグラウンドで起動させることが可能です．

tmux をコンパイルするには，libevent の Version2 が必要ですが，ほとんどのディストリビューションには Version1 がインストールされているようです．もし tmux の configure で configure: error: "libevent not found" のエラーが出た場合は libevent の最新版をインストールしてから（図 4.2.8），次に tmux をコンパイル・インストールします（図 4.2.9）．

2016 年 3 月 15 日現在の libevent の最新バージョンのソースファイルは libevent-2.0.22-stable.tar.gz，tmux の最新バージョンのソースファイルは tmux-2.1.tar.gz です．

```
libevent サイト URL:   http://libevent.org/
tmux サイト URL:       https://tmux.github.io/
```

```
# wget downloads.sourceforge.net/project/levent/libevent/libevent-2.0/libevent-
2.0.22-stable.tar.gz
# zcat libevent-2.0.22-stable.tar.gz |tar xfv -
# cd libevent-2.0.22-stable
# ./configure
# make
# make install
# ldconfig
```

図 4.2.8 libevent のインストール手順（libevent-2.0.22-stable の場合）

```
# https://github.com/tmux/tmux/releases/download/2.1/tmux-2.1.tar.gz
# zcat tmux-2.1.tar.gz |tar xfv -
# cd tmux-2.1
# ./configure
# make
# make install
```

図 4.2.9 tmux のインストール手順 （tmux-2.1 の場合）

また，tmux を使用した OpenSim.exe の起動スクリプトの例を 図 4.2.10 に示します．screen では /etc/rc#.d からリンクを張って，OS の起動時に OpenSim.exe を自動起動しようとしても失敗することが多かったのですが，tmux では問題なく自動起動が可能です．

通常のコンソールから，OpenSim のコンソールに移動するには，**tmux a** と入力します．もし複数のプログラムを tmux で動かしているなら，**tmux a -t opensim_region** とします．tmux のコンソールから抜けるには **Ctrl+b** を入力してから **d** を入力します．

その他の tmux のコマンドについては，Web 上で検索するとたくさん出てきますのでここでは省略します．

```
#!/bin/bash
#
# opensim_region (OpenSim Region Server) using tmux
#                                                    by Fumi.Iseki

OPNSM=OpenSim.exe                  # 起動するプログラム
OSDIR=/usr/local/opensim           # OpenSim をインストールしたディレクトリを指定
PRGFL=/etc/init.d/opensim_region   # このスクリプトのフルパス
MONO=/usr/local/bin/mono           # mono へのパス
SLEEPTM=15                         # リスタート時のスリープ時間 (s)
CHKTM=30                      # 監視間隔 (s)
CNTMX=60                      # ハングアップ監視回数.判定時間は $CNTMX * $CHKTM (s)
#OPTION=" --server"

export PATH=/usr/local/bin:$PATH
export MONO_THREADS_PER_CPU=512

SCRNID=opensim_region
PIDFL=/var/run/opensim_region_check.pid

start() {
    echo "OpenSim Region Server Start."
    tmux new -d -s $SCRNID -c $OSDIR/bin "$MONO $OPTION $OPNSM"
}

stop() {
    tmux send-keys -t $SCRNID C-m "quit" C-m
    echo "OpenSim Region Server Stoped."
}

kill_check() {
    PID=`cat $PIDFL 2> /dev/null`
    if [ "$PID" != "" ]; then
       kill -9 $PID 2> /dev/null
       rm -f $PIDFL
    fi
}
```

```
start_check() {
    kill_check
    echo $$ >| $PIDFL
    PRVTM=" 0:00"
    COUNT=0
    #
    while [ "" = "" ]; do
        CHECK=`ps ax|grep tmux |grep $SCRNID`
        if [ "$CHECK" = "" ]; then
            start
            PRVTM=" 0:00"
            COUNT=0
        else
            CHECK=`ps ax|grep $MONO |grep $OPNSM |grep -v tmux`
            EXETM=`echo $CHECK | awk -F" " '{print $4}' `
            if [ "$PRVTM" = "$EXETM" ]; then
                COUNT=`expr $COUNT + 1`
                if [ $COUNT -ge $CNTMX ]; then
                    tmux send-keys -t $SCRNID C-m "quit" C-m
                    sleep $CHKTM
                    PID=`echo $CHECK | awk -F" " '{print $1}' `
                    kill -9 $PID
                else
                    tmux send-keys -t $SCRNID C-m "show stats" C-m
                fi
            else
                COUNT=0
            fi
            PRVTM=$EXETM
        fi
        #
        sleep $CHKTM
    done
}


case "$1" in
  start)
    start
    /bin/bash $PRGFL check &
    ;;
  stop)
    kill_check
    stop
    ;;
  restart|reload)
    kill_check
    stop
    sleep $SLEEPTM
    start
    /bin/bash $PRGFL check &
    ;;
  check)
    start_check
    ;;
  *)
    echo $" Usage: $0 {start|stop|restart|check}"
    exit 1
esac

exit $?
```

図 4.2.10 tmux を使用した OpenSim.exe の起動スクリプト例

応用編

5. Japan Open Grid へのグリッド接続

ここでは PC（主に Windows PC）で OpenSim を動かし，起動した SIM を JOG に接続する方法を紹介します．これにより，あなた専用の SIM を JOG の一部として運用することが可能になります．ただし，接続には若干のネットワークに関する専門的な知識が必要になりますが，この際ですのでネットワークの勉強も兼ねて頑張ってみましょう!!

なお，OpenSim を自宅などの NAT 内ネットワークから起動する場合は，通常は NAT ループバック（別名ヘアピン NAT）機能付きのブロードバンド（BB）ルータが必要となりますが，JOG 配布のバイナリは予め NAT 対応の修正が加えられているため，これを使用すれば通常のルータでも NAT 内で OpenSim を問題なく動作させることができます．ただし，ネットワークの構成が複雑になっている場合は，問題が発生する可能性があります．



自分専用の SIM を JOG に持てるんだね！

しかも NAT ループバックなしのルータでも
大丈夫なんでしょう!!
どうやったの？ すごいな〜!!

ところで NAT ループバック って何？





綺麗さっぱり忘れているみたいね．
前の巻を読もうね．
でも今回はそこは本当に気にしなくても
よいかも．

ふふふっ．私がちょっいと本気
を出せばこれくらいは ....
お茶の子さいさいなのだ ....
えへえへえへ





でもまだまだ面倒な設定は残っているよ．

それから，あんまり先生を褒めると直ぐに調子
に乗って使い物にならなくなるから，ほどほど
にね．

5.1 事前準備

まず，2.2 節を参考にして，JOG のサイト（https://www.jogrid.net）にユーザ登録し，続けてアバターを作成します．このアバターは後で作成する SIM のオーナーアバターとなります．

なお,JOG のサイトにユーザ登録した際には,折角ですので,是非とも「自己紹介」をお願いします．

次にサイトのチュートリアルのコース（https://www.jogrid.net/wi/course/view.php?id=4）から,JOG 専用の OpenSim のバイナリープログラムをダウンロードします．2016 年 3 月 16 日現在では,0.9.0Dev が最新バージョンです．ダウンロードしたファイル（zip ファイル）を適当なフォルダで展開すると，OpenSim-0.9.0Dev というフォルダが生成されます．

もし貴方の使用している BB ルータが NAT ループバック対応なら,5.3 節に進んでください．NAT ループバック対応でない場合は，このまま 5.2 節へ進みます．NAT ループバック対応について不明なルータの場合も，5.2 節です．

なお，NAT ループバック対応のルータとして有名なものは以下通りです（実際に NAT ループバック可能かどうかは確認していません．自己責任でお買い求めください）．

バッファロー　BBR-4HG，WZR-1750DHP2
YAMAHA　　RT-58i，RT107e，NVR500

5.2 IP アドレスの設定（ルータが NAT ループバック非対応の場合）

5.2.1 外部向けのネットワーク情報を得る

まず，サーバとして使用する PC が外部接続する場合に使用している**グローバルな IP アドレス**を調べます．″IP アドレス 確認″ などのキーワードで検索すれば，いくつかのサイトがヒットしますが，ここでは JOG で用意している **https://www.jogrid.net/addrtool.php** を使用します．

サーバとして使用する PC 上で Web ブラウザを起動し，https://www.jogrid.net/addrtool.php に接続すると，IP アドレス，ホスト名，さらに使用可能な UUID が表示されます（図 5.2.1）．ホスト名が IP アドレスと同じ場合（または表示されない場合）は残念ながらここで紹介する手順はそのままでは使用できません．別途 Dynamic DNS（DDNS）等を使用するか NAT ループバック対応のルータを購入する必要があります．

UUID は OpenSim のサーバを識別するためのコードです．これらの **IP アドレス**，**ホスト名**，**UUID** はメモを取るかコピペしておきます．

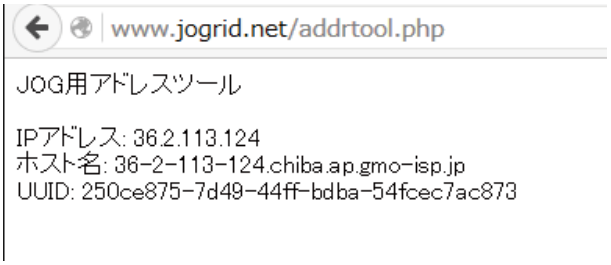


図 5.2.1 JOG 用アドレスツール

5.2.2 PC のローカル IP アドレスを得る

次にサーバとして使用する PC のローカルな IP アドレスを調べます．そのために，Windows のコマンドプロンプトを起動します．

Windows のバージョンによってコマンドプロンプトの起動の仕方は若干異なりますが，Windows7 の場合は，スタートメニューがら″**プログラムとファイルの検索**″に **cmd** と入力し実行します．

Windows10 の場合は，左下隅のウィンドウアイコン（Window7 のスタートメニューの位置にあるア

イコン）を右クリックし，″**ファイル名を指定して実行**″を選択します．表示されたダイアログの名前の部分に **cmd** と入力し，OK をクリックします．

コマンドプロンプトが起動したら，**ipconfig** と入力します．最初の方に表示される ″**イーサネット アダプター ローカル エリア接続**″ の **IPv4** の部分がローカル IP アドレスです．これもメモしておきます（図 5.2.2）．

一般家庭で使用する BB ルータの場合は，通常は 192.168.1.x などの値になるはずです（図 5.2.2 では 192.168.27.210）．

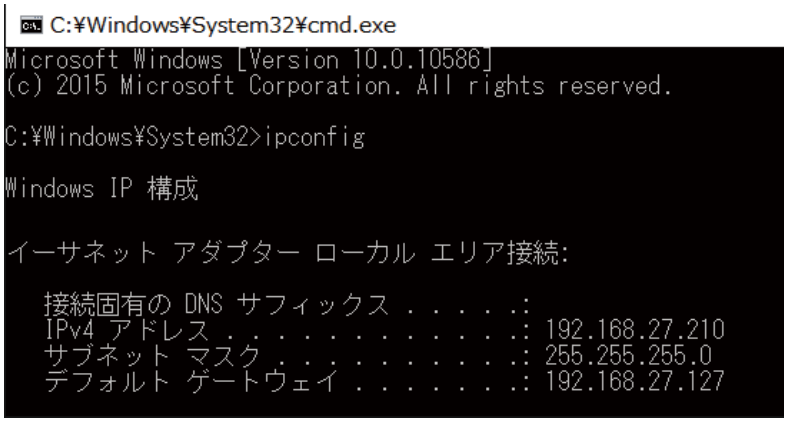


図 5.2.2 ローカル IP アドレスのチェック

5.2.3 hosts ファイルの書き換え

ルータが NAT ループバック対応でない場合，PC の **hosts** ファイルを書き換える必要があります．Windows の場合，hosts ファイルは **C:¥Windows¥System32¥drivers¥etc¥hosts** にあります．管理者の権限でテキストエディタを使用してこのファイルを編集し，一行追加します．

追加する行はの書式は以下の通りです．

5.2.2 での PC のローカル IP アドレス　　5.2.1 でのホスト名

例えば，図 5.2.1，図 5.2.2 の場合は以下のようになります．

192.168.27.210　　36-2-113-124.chiba.ap.gmo-isp.jp

5.3 IP アドレスの設定（ルータが NAT ループバック対応の場合）

5.3.1 外部向けのネットワーク情報を得る

まず，サーバとして使用する PC が外部接続する場合に使用している**グローバルな IP アドレス**を調べます．″IP アドレス 確認″ などのキーワードで検索すれば，いくつかのサイトがヒットしますが，ここでは JOG で用意している **https://www.jogrid.net/addrtool.php** を使用します．

サーバとして使用する PC 上で Web ブラウザを起動し，https://www.jogrid.net/addrtool.php に接続すると，IP アドレス，ホスト名，さらに使用可能な UUID が表示されます（図 5.2.1）．NAT ループバック対応のルータを使用している場合はホスト名を気にする必要はありません．

UUID は OpenSim のサーバを識別するためのコードです．これらの **IP アドレス**，**UUID** をメモを取るかコピペしておきます．

5.3.2 PC のローカル IP アドレスを得る

ルータが NAT ループバックをサポートしている場合も，5.4.4 項のポートフォワーディングを設定するために PC のローカル IP アドレスが必要になります．5.2.2 項と同様にして（図 5.2.2），PC

のローカル IP アドレスを得ます.

5.4 ファイアウォールの設定

5.4.4 ルータのファイアウォールの設定（ポートフォワーディング）

外部からルータに届いた信号を OpenSim のサーバが動作している PC へ転送するために，ルータの NAT 変換テーブルの設定を行います．この辺の用語や操作はルータによって違いますので，ルータのマニュアルをよく読んで，以下のように，ルータの **9000** 番ポートへの **TCP** と **UDP** の信号を PC の **9000** 番ポートに転送するように設定します（図 5.4.1）.

転送元 IP	転送元ポート	転送先 IP	転送先ポート	プロトコル
ルータの WAN 側 IP アドレス又は 5.2.1/5.3.1 での IP アドレス	9000	5.2.2/5.3.2 での PC のローカル IP アドレス	9000	TCP
ルータの WAN 側 IP アドレス又は 5.2.1/5.3.1 での IP アドレス	9000	5.2.2/5.3.2 での PC のローカル IP アドレス	9000	UDP

図 5.4.1 ルータのポートフォワーディングの設定

5.4.2 Windows のファイアウォールの設定

PC まで届いた信号を Windows 上のプロセス（OpenSim のプロセス）に届けるために，Windows のファイアウォールの設定の変更を行います.

Windows 側でも **9000** 番ポートの **TCP** と **UDP** の 2 種類に関してファイアウォールを解除します.

先ずコントロールパネルのカテゴリを〝大きいアイコン〟にし，〝**Windows ファイアウォール**〟を選択します．次に左サイドの〝**詳細設定**〟をクリックします．ファイアウォール設定用のウィンドが表示されますので，左サイドの〝**受信の規制**〟をクリックします.

右サイドの〝**新しい規制**〟をクリックすると（図 5.4.2），設定用ウィザードが起動します.

最初は **TCP** に関する設定です．設定用ウィザードで〝**ポート**〟を選択し，〝次へ〟をクリックします．次の画面で〝**TCP**〟，〝**特定のローカルポート**〟を選択し，ローカルポートの値として **9000** を入力し〝次へ〟をクリックします（図 5.4.3）.

次の画面では〝**接続を許可する**〟を選択し，そのまま〝次へ〟を 2 度クリックします．最後に追加したルールに名前を付けて〝完了〟です.

Windows のファイアウォールの画面に戻ったら，今度は同様にして **UDP** についての設定も追加します.

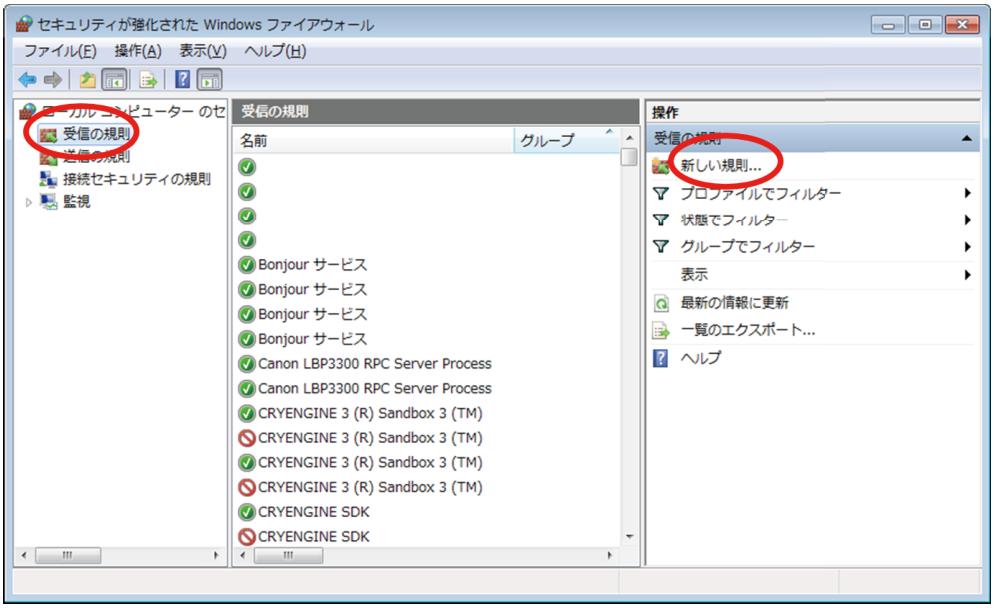


図 5.4.2 新しい規制の追加

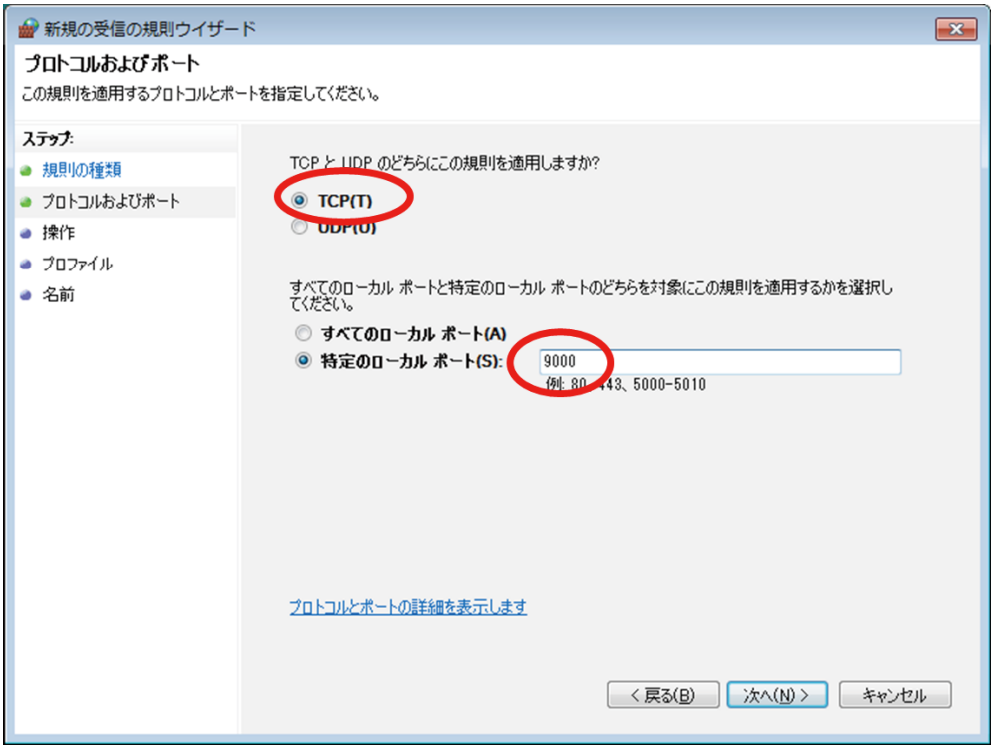


図 5.4.3 TCP の 900 番ポートを許可

5.5 OpenSim の起動

5.5.1 Regions.ini

今までのネットワークの設定を元に，OpenSim での SIM の設定ファイルである Regions.ini（実際は拡張子が .ini であれば，他の名前でも良い）をエディタで作成します.

Regions.ini を作成する際に必要になるパラメータは以下の通りです.

SIM 名 ： [ ] で括って最初の行に書く．既にあるもの以外であれば自由に設定可能
RegionUUID ： 5.2.1 または 5.3.1 での UUID
Locaction ： SIM の位置．JOG では (7985,7985) ～ (8015,8015) の正方形区画以外.
SizeX ： SIM の大きさ．256 の整数倍.
SizeY ： 必ず SizeX と同じ値にする.
InternalAddress ： 5.2.2 でのローカル IP アドレス
NAT ループバック対応のルータを使用している場合は 0.0.0.0
InternalPort ： 9000
ExternalHostName ： 5.2.1 でのホスト名
NAT ループバック対応のルータを使用している場合は 5.3.1 での IP アドレス
AllowAlternatePorts ： False

例えば 5.2 節の例（NAT ループバック非対応）の場合，Regions.ini は図 5.5.1 の様になります.

```
[MY SIM]
RegionUUID = 250ce875-7d49-44ff-bdba-54fcec7ac873
Location = 7800,7500
SizeX = 256
SizeY = 256
InternalAddress = 192.168.27.210
InternalPort = 9000
ExternalHostName = 36-2-113-124.chiba.ap.gmo-isp.jp
AllowAlternatePorts = False
```

図 5.5.1 NAT ループバック非対応の場合の Regions.ini の例

また 5.3 節の例（NAT ループバック対応）の場合の Regions.ini は図 5.5.2 の様になります,

```
[MY SIM]
RegionUUID = 250ce875-7d49-44ff-bdba-54fcec7ac873
Location = 7800,7500
SizeX = 256
SizeY = 256
InternalAddress = 0.0.0.0
InternalPort = 9000
ExternalHostName = 36.2.113.124
AllowAlternatePorts = False
```

図 5.5.2 NAT ループバック対応の場合の Regions.ini の例

このような内容のファイルをエディタで作成し bin¥Regions フォルダに Regions.ini として保存します.

5.5.2 OpenSim の起動

全ての準備が終了したら，bin¥OpenSim.exe をダブルクリックして OpenSim を起動します．OpenSim が起動すると，最初にエステート（地域）の名前を聞かれますが，デフォルトの My Estate のままで問題ありません（図 5.5.3）．つまりそのままエンターキーを押します．

次に SIM を管理するアバターの名前を入力します．ここでは 5.1 節で作成したアバターの名前を，ファーストネーム，セカンドネームの順で入力します（図 5.5.4）．

全て問題なく入力されれば，OpenSim のコマンドプロンプトが表示されます．OpenSim を終了させる場合は，このコマンドプロンプトから quit または shutdown と入力します．ウィンドウのタイトルバーの✖でクローズしてはいけません．

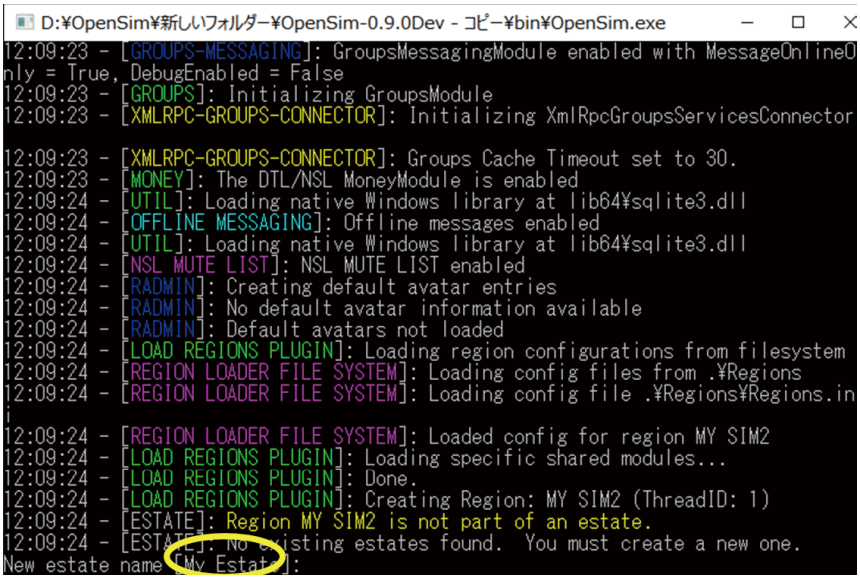


図 5.5.3 エステートの設定

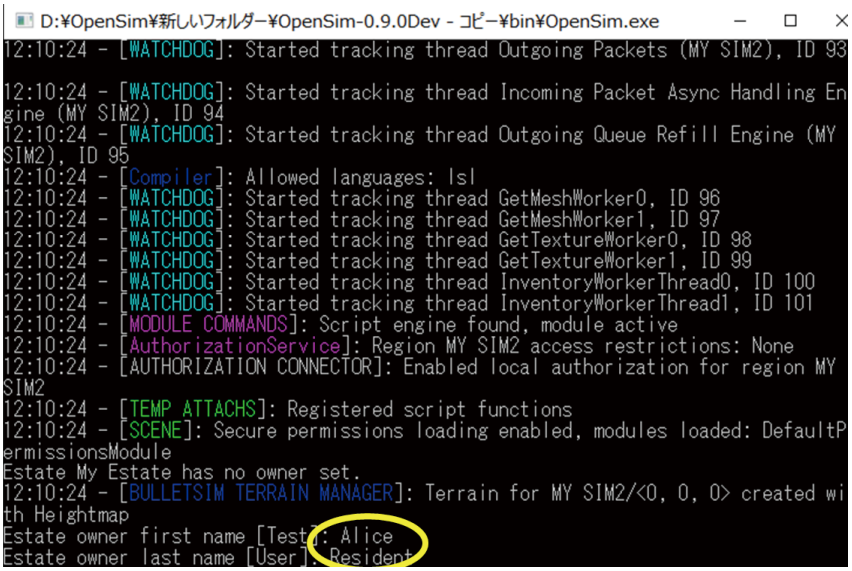


図 5.5.4 オーナーアバターの指定

5.5.3 自分の SIM への移動

自分の SIM へ移動するには，まず通常の方法で JOG へログインします．続いて世界地図で自分の SIM を検索します．すべて正常に設定されていれば，自分の SIM が地図上に現れるはずですので，そこにテレポートすれば OK です．

友達に SIM の名前を教えれば，友達を自分の SIM に呼ぶことができます．

もしどうしても上手くいかない場合は，JOG のコミュニティでどんどん質問してみましょう．色々な質問をおまちしています．;-)

5.6 より進んだネットワーク設定

実はこれまで行ったネットワーク設定では，場合によっては 5.2 または 5.3 節の設定をその都度やり直す必要があり，かなりの手間になります．もしこれらの問題を回避して，定常的に SIM を運営したいと思ったのなら，さらに若干の手間と知識とお金が必要になります．

5.6.1 ローカル IP アドレスの固定化

まず，サーバとなる PC の IP アドレスを固定化しましょう．恐らく現在は IP アドレスの設定は**″IP アドレスを自動的に取得する″**（DHCP）になってると思います．もし家庭内でネットワークに繋がる機器が多い場合は，IP アドレスが変化する可能性がありますので（実はそんなに頻繁には起こらないのですが），（念のため）これを固定 IP アドレスに変更します（図 5.6.1）．

固定 IP アドレスにするには，コントロールパネルの**″ネットワークと共有センター″**を使用しますが，具体的な方法は別途 Web ページを検索すれば分かると思いますので，ここでは省略します（設定する数値もルータの種類などによって変わります）．

もし，IP を固定化して通信ができなくなった場合は，設定を間違えていますので無理をせずに元の″IP アドレスを自動的に取得する″に戻しましょう．

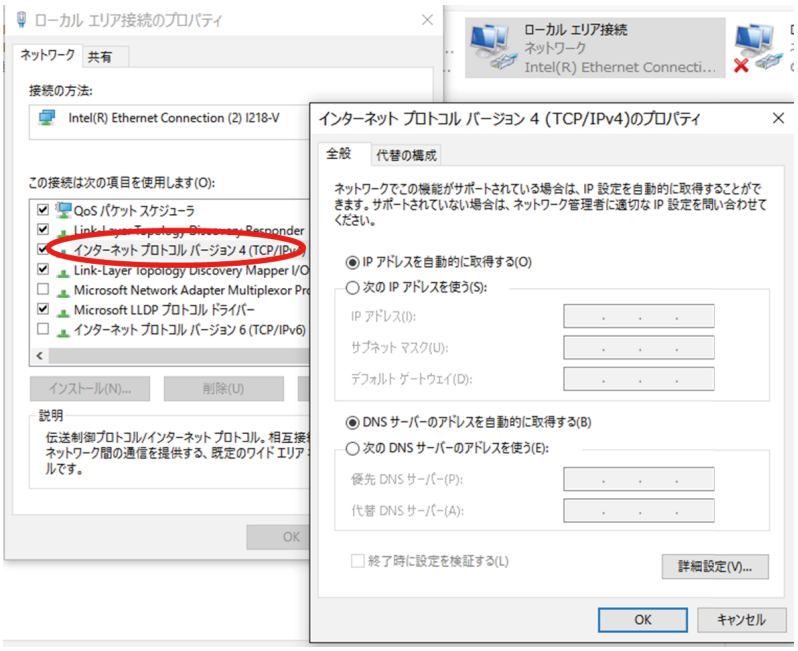


図 5.6.1 PC のローカル IP アドレスの設定

5.6.2 外向け（グローバルな）IP アドレスの固定化

インターネットに接続するためのルータの WAN 側には，自動的に IP アドレス（グローバル IP アドレス）が割り振られます．このアドレスは，ローカルな IP アドレスよりも頻繁に変更される可能性があります．

もし金銭的な余裕があれば，この IP アドレスを固定にすることもできます．ただしグローバル IP アドレスの固定化はプロバイダーによっては非常に高価になる場合もあります（非常に安いプロバイダーもあります）．

5.6.3 ドメイン名の取得

グローバルな IP アドレスが変化した場合，5.2（5.3）節のホスト名も変化します．NAT ループバック対応のルータを使用している場合は，グローバルな IP アドレスをそのままホスト名として使用しても構いませんが，その場合はグローバルな IP アドレスの固定化の問題が発生します．

ホスト名を固定化したい場合には，ドメイン名の取得を行います．ドメイン名の取得自体は，ドメインの種類によって安価なものもありますので（jp ドメインなどは高価），可能ならば取得をお勧めします．

ドメイン名は取得しただけでは意味がありません．そのドメイン名を有効にするために DNS（ドメインネームサービス）を設定しないといけません．通常，ドメイン名を販売するサイトでは DNS のサービスを行っていると思いますので，そのサービスを利用するのが良いでしょう．ただし，グローバル IP アドレスが固定化されていない場合は，Dynamic DNS と呼ばれるサービスを利用しなければなりません．

5.6.4 実際の運用

一般的には，固定 IP アドレスは高価ですので，**非固定の（動的な）グローバル IP アドレス＋ドメイン名＋ Dynamic DNS** を使用しているユーザが殆どだと思います．ただし，現時点の JOG ではグローバル IP アドレスが変化した場合に，Robust サーバが再起動されるまでテレポートに障害が発生するという問題があります（将来的には解消されるはずです．たぶん）．

また，サーバとして外部のホスティングサービスを利用するという手もあります．この場合，サーバのレンタル料がかかると言った問題もありますが，5.2 ～ 5.4 節の問題が一気に解決する可能性もあります．ただし，この場合は OS が Windows ではなく，Linux になるろ思われます．

いずれにせよ，ネットワークの設定に関しては，コストと手間とパフォーマンスを考慮して，どのようにするか決定することをお勧めします．

7. オールインワン ポータブルシステム (Windows 用)

4 章では Windows 上での OpenSim の起動方法を解説しました．しかし，それでも実用的な環境で OpenSim を運用しようとするとかなりの手間が掛かります．例えば教育機関などで OpenSim を利用した授業を行いたい場合などは，事前にそれなりの環境を用意しておく必要がありますが，これには OpenSim をある程度良く理解しておく必要があります．

そこで我々は，教育機関などで OpenSim の演習を行うことを想定して，誰でもそれなりの OpenSim 環境を Windows 上に非常に簡単に用意できるシステムの開発を行いました．その名も OpenSim-Moodle all-in-one Portable!! システムパッケージをダウンロードして解凍した後は，ファイルを３回クリック（初回以降は２回）するだけで OpenSim(Stand Alone モード）と管理用 Web インターフェイス（Moodle 上で Modlos を使用）を利用できます．

リージョンは 1 個のみですが,デフォルトでサイズが 768x768（通常リージョンの 9 個分）であり，リージョンの中央には各種アセットの販売モールも用意しています．

ハイパーグリッドの機能はサポートしていませんが（設定を手動で変えることによりハイパーグリッドモード対応可能），それ以外はほぼ OpenSim の全機能を利用可能です．是非この機会に OpenSim-Moodle all-in-one Portable をお試しください．

なお，次の 7.1 節でシステムの概要を解説していますが，今すぐ使いたい人は 7.1 を読み飛ばして 7.2 節から読んでも構いません．

情報：本システムの各管理ユーザのアカウント情報は以下の通りとなります．これらのパスワードはシステムのインストール後に速やかに変更することを**強くお勧め**します．(7.4 節参照)

OpenSim: ID admin user, Password adminpass

Moodle: ID admin, Password adminpass

MariaDB: ID root, Password rootpass



おおっ！ これは何やら簡単に使えそうなシステムだ！
やったね！



今までと比べると数段簡単だよ！
でも，ユーザとしては良いけど，サーバの勉強にはならないね．



でも OpenSim のしきいが低くなってユーザが増えるかも !!

7.1 システムの概要

本システムでは OpenSim を StandAlone モードで起動します．StandAlone モードは OpenSim.exe 単体で全てのサーバ機能を実現するモードです．StandAlone モードを使用するのは，OpenSim のシステムの複雑さを回避するためで，負荷分散と拡張（Grid 接続）という点を除けばほぼ Grid モードと同等の機能を実現できます．また，同時に通貨を管理する DTL/NSL マネーサーバ [1] も起動します．これにより OpenSim 上で通貨を使用することが可能となります．

ネットワーク上のサーバシステムとしては Xampp ポータブルを使用して Apache（Web サーバ，PHP 付き）と MariaDB（MySQL 互換の DB サーバ）を起動しており，さらにこの上で Moodle[2] と呼ばれる e-Learning システムを起動します．Moodle は世界中で使用されている教育用の Web システム（Course Managment System または Leaning Management System）です．この Moodle 上のプラグインとして OpenSim 用 Web インターフェイスである Modlos[3] を使用します．Modlos は Moodle の機能を利用して OpenSim 全体を管理します．ユーザ自らによるアバター作成はもとより，登録アバターの管理，リージョンの管理，マネー管理が可能です．

Moodle 上のプラグインとしては更に OpenSim/SecondLife 上のオブジェクトを Moodle 上から利用するための Sloodle[4] も同梱しています．また授業等で使用されることを想定して，おまけとして自動出席管理ブロック [5] および申請書類提出モジュール [6] も予めインストールされています．

つまり OpenSim-Moodle all-in-one Portable = OpenSim + Xammp + Apache + MariaDB + PHP + Moodle + Modlos + Sloodle です（図 7.1.1）．

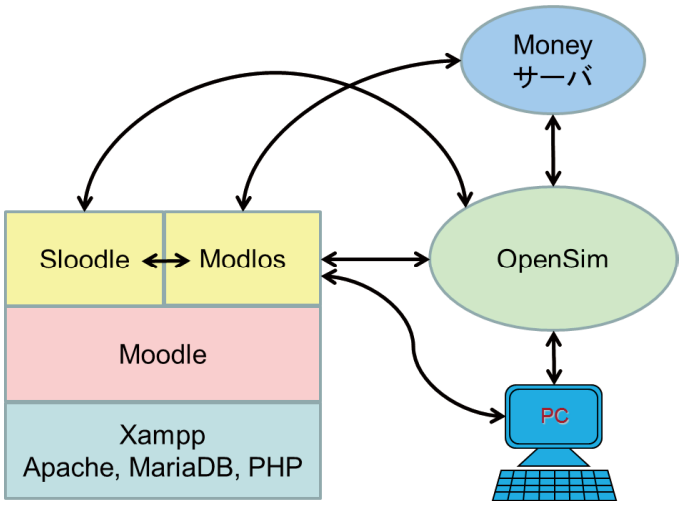


図 7.1.1 OpenSim-Moodle all-in-one Portable の各構成要素の関係

参照）

[1] DTL/NSL マネーサーバは東京情報大学 NSL 井関ゼミ（以降 NSL）で開発している OpenSim 用マネーサーバです．
http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?OpenSim%2FMoneyServer

[2] https://moodle.org

[3] Modlos は NSL で開発している Moodle 用 OpenSim Web インターフェイスです．
http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?Modlos

[4] 今回同梱している Sloodle 2.2.0 は Sloodle 2.1.0 を NSL で改良したものです．
http://www.sloodle.org, http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?Sloodle

[5] 自動出席管理ブロックは NSL で開発している Moodle 用出席管理システムです．
http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?autoattend

[6] 申請書類提出モジュールは NSL 開発している Moodle 用各種申請書類の提出用システムです．
http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?mod_apply

7.2 取りあえず OpenSim-Moodle all-in-one Portable を使う

7.2.1 パッケージのダウンロードと展開

先ずシステムパッケージを

http://www.nsl.tuis.ac.jp/xoops/modules/d3downloads/?page=singlefile&cid=8&lid=54

からダウンロードします．ただし，現在は Windows 用のみ用意されています．ダウンロード後，適当なフォルダにファイルを展開します．展開すると OpenSim_Moodle というフォルダが作成され，その内容は図 7.2.1 の様になります．



図 7.2.1 OpenSim-Moodle all-in-one Portable のファイル内容

7.2.2 初期設定ツールの起動

まず最初に，図 7.2.1 の tools フォルダの中の，初期設定ツールである Reset_Env.bat (*) を実行します．Reset_Env.bat を実行することにより，使用環境を自動的に判別し，各種設定ファイルを環境に合わせて書き換えます．Reset_Env.bat は最後に PC の IP アドレスを表示しますので表示された IP アドレスは必ずメモしておきます（図 7.2.2）．

Reset_Env.bat はプログラムの保存場所を変更した時に一度だけ実行すれば良く，毎回起動する必要はありません．

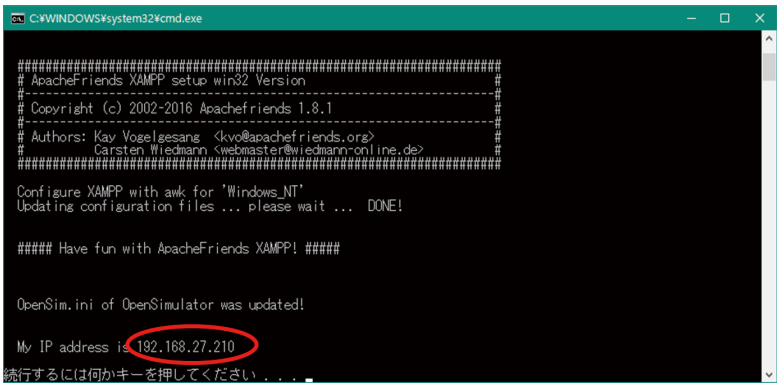


図 7.2.2 Reset_Env.bast の実行結果

7.2.3 Moodle（Apache と MariaDB）の起動

次に，図 7.2.1 の Start Moodle.exe(*) を実行します．これにより，Web サーバの Apache と DB サーバの MariaDB(mysqld) が起動し,Moodle にアクセス可能となります（図 7.2.3）．なお,このウィンドウは手動で閉じてはいけません．終了させる場合には Stop Moodle.bat(*) を起動します．

PC のブラウザから Moodle にアクセスする場合は，Reset_Env.bat 実行時に表示された IP アドレスを使用して http://IP アドレス / とします（図 7.2.2 の場合は http://192.168.27.210/）．

7.2.4 OpenSim の起動

続いて Start OpenSim.bat(*) をダブルクリックすれば，マネーサーバと OpenSim の本体が自動的に起動し，各種ビューアからアクセス可能となります．ただしリージョンの規模によってはコンソールにプロンプトが出た後も，ログインできるようになるまで暫く時間が掛かる場合があります．

なお，サーバ起動時にファイアウォール関連のダイアログが表示されるかもしれませんが，この場合は「アクセスを許可する」をクリックします．

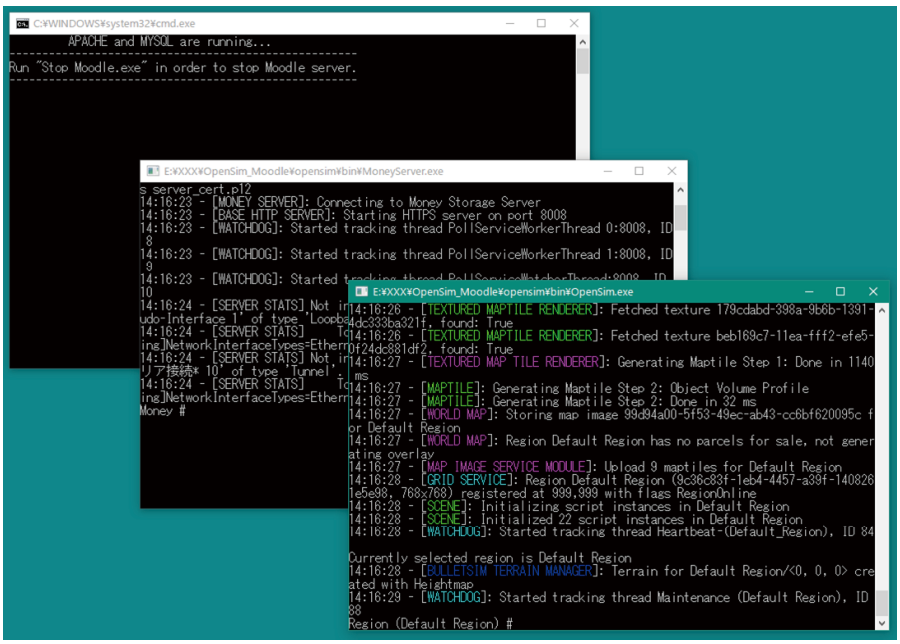


図 7.2.3 Start Moodle.exe と Start OpenSim.bat の実行結果

＊注）

Windows のデフォルト設定では，拡張子 .bat，.exe 等は表示されません．ただし，これはセキュリティ的には非常にまずい状態ですので，コントロールパネルの「フォルダオプション」(Windows7) または「エクスプローラのオプション」(Windows8/10) の「表示」タブで，「登録されている拡張子は表示しない」のチェックボックスを必ず外すように**強くお勧め**します．

7.2.5 システムの利用

図 7.2.3 の状態で OpenSim にアクセスすることが可能です．クライアントのビューアを起動してグリッドマネージャを開き Login URI を設定します．Login URI は具体的には 7.2.2 で表示された IP アドレスを使用して http://IP アドレス :9000/ とします (9000 は固定)．

図 7.2.4 に Singularity Viewer の場合の例を示します (IP アドレスは 192.168.27.210 とする)．

① Grid Manager のボタンをクリックします．
② Create ボタンをクリックする．
③ Login URI に http://192.168.27.210:9000/ と入力．
④ Advanced ボタンをクリックするとビューアの背景が変化し，⑤に各種の情報が表示される．
⑤ グリッドの名前は "My Grid" となる，なお，これらの情報中で最も重要なものは Helper URL である．
⑥ アバターの ID とパスワードを入力する．初期管理用として用意されているアバターの ID とパスワードはそれぞれ,admin user，adminpass である．(パスワードは後で Modlos のアバター編集画面から変更を行うこと)
⑦ LogIn ボタンを押し，OpenSim にログインする．

アバターの ID とパスワード（初期値はそれぞれ admin user, adminpass）を入力し，LogIn ボタンをクリックすると，デフォルトで図 7.2.5 のようなリージョンにログインできます．

注）設定は "My Grid" という名前で，Grid Manager に保存される．

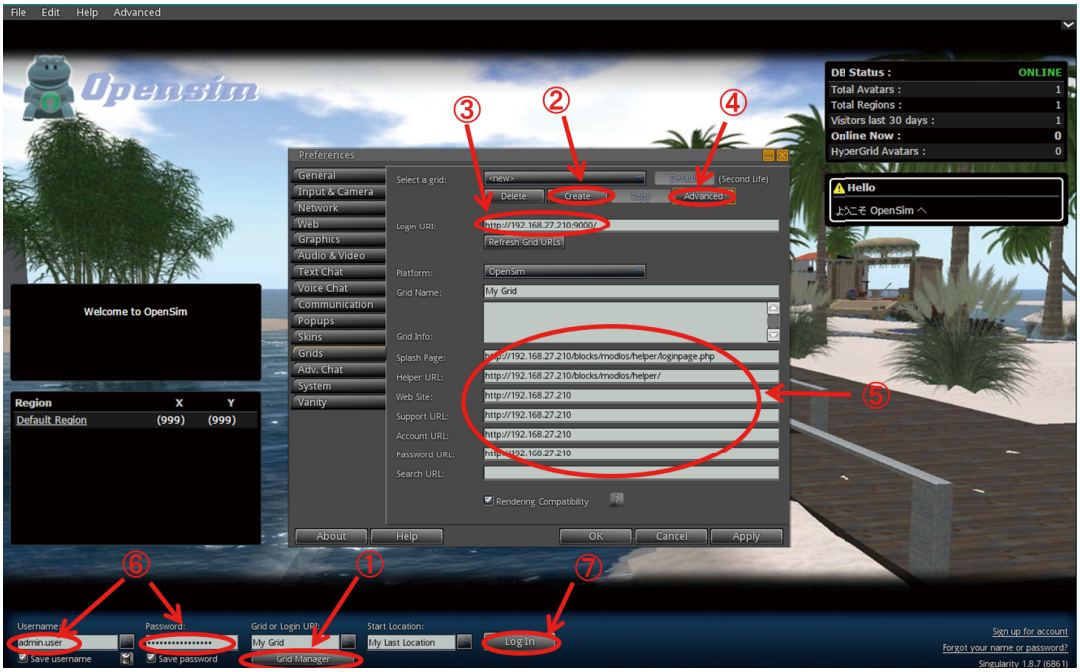


図 7.2.4 Singularity Viewer での設定

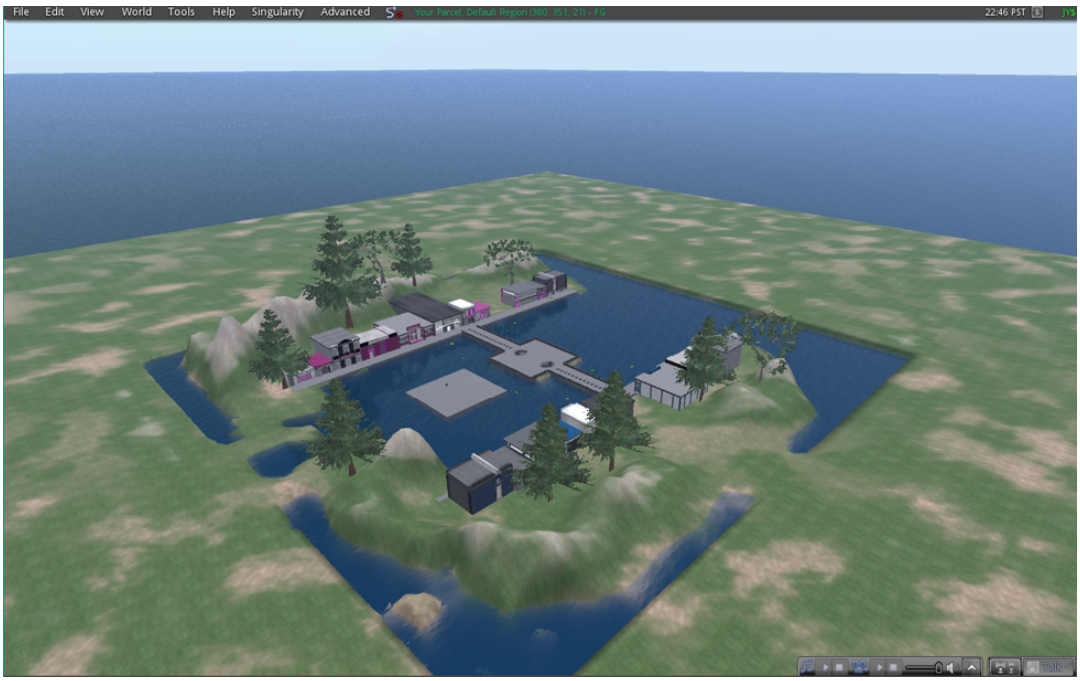


図 7.2.5 デフォルトの初期リージョン

7.2.6 システムの停止

システムを停止させる場合に各ウィンドウの「×」(閉じる)をクリックしてはいけません.「×」(閉じる)で強制終了させると,処理中のデータがDBに反映されなかったり,Webサーバや DBサーバが正しく終了せずにバックグラウンドで動き続ける場合があります.

システムを停止させる場合は,まずOpenSimのコンソールウィンドウの「Region (Defaut Region) #」プロンプトに「quit」と入力します.続いてMoneyサーバの「Money #」プロンプトにも「quit」と入力します.もし,コンソールにプロンプトが表示されていない場合は,数回エンターキーを押します.

OpenSimとマネーサーバのコンソールウィンドウが閉じたら,Stop Moodle.exe を起動します.何も問題なければ,先に Satrt Moodle.exe で起動したウィンドウも自動的に終了します.

7.3 トラブルシューティング

7.3.1 Start Moodle.exe が失敗する

起動手順で最も問題が発生すると思われるのが,Start Moodle.exe の失敗です.これは大抵の場合,ApacheやMariaDB(mysqld)が使用するはずのネットワークソケットを他のプロセスが既に使用していることに起因します.

インストール後,始めから動ない場合は Apacheが使用するはずの80番ポートのネットワークソケットをシステムプロセスやSkypeなどのプロセスが既に使用している可能性があります.80番ポートを使用しているプロセスを特定するために,Windosのスタートメニューで cmd を検索して起動します.コンソール画面が表示されたら netstat -nao |grep :80|grep LISTEN を実行します.

0.0.0.0:80 や [::]:80 が表示された場合は(図7.3.1),80番ポートを使用しているプロセスが存在しますので,その行の一番右の番号(プロセスID)をメモして,タスクマネージャーでプロセスを確認して,(可能なら)そのプロセスを停止させます.

筆者の場合もWindows7のPCで問題が発生し,上記コマンドで調べた結果,プロセスIDが4のプロセスが原因と分かりました.ただし,4はWindowsのシステムプロセスで実際にはプロセスの中のサービスを特定しなければならず,「コントロールパネル」→「管理ツール」→「サービス」で不明なサービスを一個ずつ一時停止させた結果「World Wide Web 発行サービス」というサービスが原因であることが分かりました(筆者の例です).

もし,以前は動いていたという場合は,前回の Apache または MariaDB (mysqld) が正常終了せずにバッググラウンドで動いている可能性があります.タスクマネージャーでタスク(プロセス)を表示させ,「Apache HTTP Server」または「mysqld.exe」があるかどうかを確認し,もし存在すればそのタスク(プロセス)は終了してください.

7.3.2 Reset_Env.bat で表示される IP アドレスがおかしい

ネットワークインターフェイスを2個以上搭載している場合,スクリプトがIPアドレスを正しく決定できない可能性があります.opensim¥bin¥OpenSim.iniのBaseIP および server¥moodle¥config.php の $CFG->wwwroot を手動で正しく設定してください.

```
C:¥Windows¥System32>
C:¥Windows¥System32>netstat -nao|grep LISTEN |grep :80
  TCP         0.0.0.0:80             0.0.0.0:0              LISTENING       14468
  TCP         0.0.0.0:8008           0.0.0.0:0              LISTENING       12556
  TCP         [::]:80                [::]:0                 LISTENING       14468

C:¥Windows¥System32>
```

図 7.3.1 netstat コマンドで 80 番ポートを使用しているプロセスを探す

7.4 各種パスワードの変更

7.4.1 Moolde

Moodle にログインしたら，右上にある自分の画像をクリックして「プロファイル」を選択します．プロファイル画面になったら，中央の「ユーザ詳細」の中に「プロファイルを編集する」のリンクがありますので，それをクリックすればパスワードを変更できるページに移動できます．

7.4.2 OpenSim

OpenSim のアバターのパスワードは Modlos から変更します．まず，Moodle にログインしたら右側にある「Moodle OpenSim」ブロックの「マイアバター」リンクをクリックします．するとあなたが管理するアバター一覧が表示されますので，変更したいアバターの EDIT ボタンをクリックしてパスワードの更新を行います．

7.4.3 MariaDB（mysqld）の管理（root）ユーザ

tools フォルダの中にある mysql_pass.bat を起動し，現在のパスワードと新しいパスワードを入力します．特に何もメッセージが出力されなければ，パスワードの変更は成功したことになります．

7.5 Xampp

本システムでは Apache と MariaDB の起動と停止に Start Moodle.exe と Stop Moodle.exe を使用していますが，本来のツールである XAMPP Control Panel も使用できます．XAMPP Control Panel を起動するには，tools フォルダの中の xampp-control.bat または server フォルダの xampp-control.exe をダブルクリックします（図 7.5.1）．ただし，XAMPP Control Panel を終了する場合はご注意が必要です．XAMPP Control Panel を終了させるには必ず「Quit」ボタンをクリックします．Quit ボタンをクリックせずに，パネルの右上の「×」をクリックしてもプログラムはバックグラウンドで走り続けています．

止めたと思って XAMPP Control Panel の起動を繰り返していると，ツールバーのインジケータが大変なことになります．（^^；

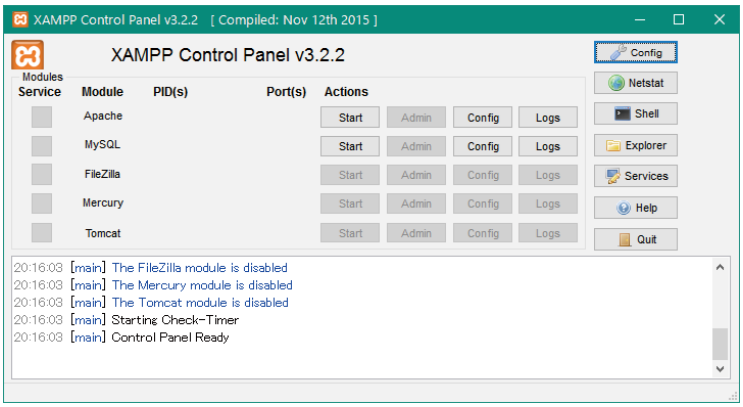
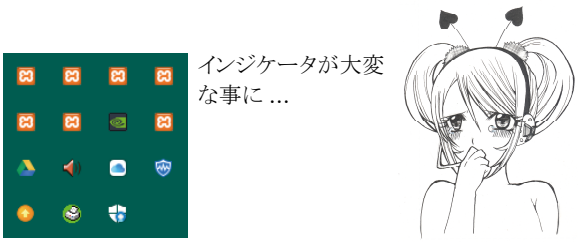


図 7.5.1 Xampp Control Panel



7.6 Modlos の設定

ここでは，Modlos-3.3.1 の簡単な解説を行います．

7.6.1 一般メニュー

オンライン アバター

現在オンライン中のアバターの一覧を表示します．

ワールドマップ

ワールドマップをリアルタイムに表示します．ビューアに付属しているワールドマップはデータが強力にキャッシュされていますが，このマップデータはキャッシュされないので，現在落ちている SIM（リージョン）などをリアルタイムで知る時とができます．ただし，頻繁にマップ画像を更新するとサーバから DOS アタックだと誤解されて，一定期間サーバにアクセスできなくなります（画像が表示されなくなります）．マップ画像の拡大・縮小に一々データを取りに行っているところが欠点です．

アバターリスト

登録されている全アバターの一覧を OpenSim へのログイン日付順で表示します．

マイ アバター

自分が管理するアバターの一覧を表示します．Modlos では，OpenSim のアバターを Moodle のアカウントとは別に作成します．ただし，両者は紐づけされて管理されます．

アバター名の先頭にある EDIT リンクをクリックすると，アバターを削除したりパスワードを変更することが可能です．

リージョンリスト

リージョン（SIM）の一覧を表示します．ただし今回は StandAlone モードなので，作成できるリージョンの数はそんなに多数にはなりません．

マイリージョン

自分のアバターが管理するリージョンの一覧を表示します．

アバター作成

アバターを新規に作成します．一人の Moolde アカウントで何名のアバターを作成可能かは，Modlos の全般設定で指定することが可能です．管理者がアバターのテンプレート（ひな型）を用意すれば，ユーザはアバターの初期の容姿をそのテンプレート（ひな型）の中から選ぶことができます．

イベントリスト

インワールドでのイベントのリストを表示します．または，イベントを登録することも可能です．登録されたイベントはインワールドで検索可能です．

管理メニュー

管理用メニューに移動します．

7.6.2 管理メニュー

一般メニュー

一般メニューに戻ります．

管理コマンド

管理用コマンドを実行します．管理コマンドしては以下の物があります．

外部 DB との同期

Moodle の DB と OpenSim の DB のアバター情報の同期を取ります．また，Sloodle がインストールされている場合は，Sloodle のアバター情報とも同期をとります．通常は実行する必要はなく，何らかの障害でアバターデータに不都合が生じた場合に実行します．

テクスチャのキャッシュデータをクリア

OpenSim 内の JPEG2000 のデータの表示用キャッシュをクリアします．画像の表示がおかしくなった場合に使用します．

ログイン情報テーブルをクリア

OpenSim が何らかの原因でクラッシュして，その後回復した場合，クラッシュ前にログインしていたアバターの情報が DB 上に残ったままになることがあります（ログインした状態になっている）．この場合にログインテーブルの内容をクリアするためのコマンドです．

データベースのクリーンアップ

リージョン（SIM）の画像の一部は，定期的に生成されますが，その後は再利用されることなく DB のアセットテーブルに蓄積されていきます．このコマンドはこの使用されない画像データを DB から削除するコマンドです．システムが混んでいない時を見計らって定期的に実行すると良いでしょう．

エステート

エステートの管理用メニューです．エステートは土地の管理単位であり，ある土地のオーナを変更する場合などに使用します．このメニューでエステートの定義とそのエステートのオーナを指定することができます．

ログインスクリーン

ビューアのログイン画面で，右側に表示される情報ボックスを定義します．

マネー管理

OpenSim 内のマネーを管理します．全アバターや特定アバターへの送金，または集金が可能です．アバター間のお金の移動もできます．また，売り上げデータベースで賞品の売上高ランキング等の表示もできます．

HG アバター（HG モードのときのみ）

OpenSim が HG モードをサポートしている場合，外部からの SIM に訪れた HG アバターの一覧を表示します．

初期アバター

ユーザがアバターを新規作成する場合のテンプレートとなるアバター（テンプレートアバター）を設定します．テンプレートアバターを設定するには，まずテンプレートとなるアバターをインワールドで実際に作成します．作成が終わったら，アバターのスナップショットを取り，そのアバターの UUID と共にここに登録します．

アバターの新規作成時には，テンプレートアバターの UUID を元に（名前以外の）全てのアバター情報が新規のアバターにコピーされることになります．従って，テンプレートアバターのインベントリはなるべく少なくした方が良いでしょう．テンプレートアバターが大量のインベントリを持っていると，新規のアバターの作成（コピー）時に非常に時間が掛かることになります．

全般設定

Modlos の全般的な設定を行います．詳細は次の 7.6.3 項を参照してください．

7.6.3 全般設定

ここでは全般設定のうち，ページ内での説明では分かりにくいものについてのみ解説します．

データベースへの接続

データベースに障害があった場合や停止している場合，Modlos はデータベース接続待ちの状態になりレスポンスが帰ってこなくなり（または非常に遅くなり），Moodle 全体に影響が出ます．このような状況を防止するため，データベースが何らかの理由により停止している場合は（OpenSim のインストール前などの場合も）ここのチェックをはずします．

MySQLi インターフェイスの使用

最近のデータベースは全て MySQLi か PDO インターフェイスを使用していますので，通常はチェック入れて置きます．チェックを外すのは MySQLi をサポートしていない古いデータベースを使用している時のみです．

ベース アバターの GUID

データ作成時にテンプレートアバターを使用しない場合の，テンプレートとなるアバターの UUID を指定する．新規アバターには，このアバターの名前以外の全ての情報がコピーされる．この場合，新規アバターは一種類のみとなるので，特に理由がないならテンプレートアバターの使用をお勧めする．

画像処理用プログラム

Windows の場合は「opj_decompress （Windows)」を選択してください．これは OpenJPEG が提供している JPEG2000 のデコードプログラムです（Modlos に同梱）．他の選択肢は Linux 用です．

Profile, Search, Message, Group の各モジュールの指定

OpenSim-Moodle all-in-one Portable ではデフォルトの ... (on Moodle DB) のままにして置いてください．この設定で問題なく作動するように設定済みです．

Flotsam グループデータベース読み込みキー / 書き込みキー

OpenSim.ini と一致済みです．変更しないでください．

Viewer と Money スクリプトとのアクセスキー

MoneyServer.ini と一致済みです．変更しないでください．

Money スクリプトの処理を行うリージョンサーバ

OpenSim-Moodle all-in-one Portable の場合は Moodle と OpenSim.exe が同じ PC で動いているはずですので，http://127.0.0.1:9000/ で問題ありません．

7.7 OpenSim の追加モジュールとヘルパースクリプト

Moldos では，OpenSim の追加モジュールをサポートするためのヘルパースクリプトを内蔵しています．ヘルパースクリプトがサポートする OpenSim の追加モジュールは，OpenSimProfile，OpenSimSearch，NSL Mute Message，Floatsam Group(改)，DTL/NSL Money です．これらのモジュールはヘルパースクリプトを通して，XMLRPC で通信を行います．

ヘルパースクリプトは moodle¥blocks¥modlos¥helper にありますが，これがビューアのグリッ

ドマネージャで Helper URI として正しく認識されていないと（図 7.2.4 の⑤），追加モジュールは OpenSim 上で正しく動作しません．

また，moodle 全体が https で通信を行う場合も問題があります．OpenSim の XMLRPC が https をサポートしていないからです．この場合は，ヘルパースクリプトを Moodle の外（https の外）に出さなければいけません．これには幾つかの手法がありますが，もっとも簡単な手法は単体のヘルパースクリプトをダウンロードして，新たに http で通信可能なように設置し直すことです．

単体のヘルパースクリプトは

http://www.nsl.tuis.ac.jp/xoops/modules/d3downloads/?page=singlefile&cid=8&lid=52

からダウンロードできます．

マニュアルも以下にありますが，本文は（中学生）英語ですので，もし良く分からなければ，JOG のフォーラムでどんどん質問してください．

http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?OpenSim%2FMoneyServer%2FHelper%20Script

6. Unity3D への Data 変換

Unity は非常に有名なゲームエンジンです．幾つもの代表的なゲームタイトルが Unity を使用して開発されてきました．ただし，Unity では一つのシーンに対して複数の開発者が同時に作業すると言ったことが苦手です（やろうと思えばやれないこともないけど，全然スマートじゃない）．

一方 OpenSim では一つの世界を複数の開発者達が協力して作り上げていくことができます．ただし，OpenSim 内に作られたものは，OpenSim に入らなければ見ることはできません．OpenSim のクライアントソフト（ビューア）の操作方法も覚えなければいけません．そんなこんなで，OpenSim に作成したものを一般に人に見てもらうのは非常に面倒です．

それでは，OpenSim で作成したものを Unity に読み込んで，Web アプリケーションまたは通常のアプリとして出力したらどうでしょう....　うん．これなら皆に見てもらえるんじゃないかな！ ということで開発したのが OpenSim の OAR データを Unity に読み込ませるため OAR Converter です．

OpenSim のデータを Unity に変換するサービスとしては，以前 Tipodean 社がそれを行っていましたが，価格が高い上に手法もクローズドでした．今では Web ページもリンク切れのようです．一方我々の OAR Converter はオープンソースで無料です（寄付等は大歓迎）．

これで OpenSim がもっと（マインクラフトのように）流行ればうれしいなっと！



Unity に変換できれば，また色々できるね！

私も Unity ちゃんみたいに有名になりたいな！





また訳の分からんことを！

まあ，夢を持つのは自由さ．
ふっ．



6.1 OAR Converter の概略

OAR Converter (http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?OAR%20Converter) はまず Linux 上で C++ により実装されました．OAR Converter は OpenSim のデータアーカイブである OAR (OpenSim ARchives) ファイルを解析して，OpenSim のプリム，スカルプテッドプリムおよびリンデンメッシュの各データを Collada 形式の dae ファイルとして出力します．Collada は 3D データの交換用フォーマットで，テキスト形式の単純な XML で記述されており，解析が容易であるという特徴を持っています．

出力された Collada 形式の dae ファイルは，SelectOARShader と呼ばれる Editor 拡張を通して Unity に読み込まれます．SelectOARShader は OpenSim のオブジェクトのマテリアルのパラメータを参照して，より良いと思われる Unity のシェーダを選択して適用する機能を持っています（ただしベストな選択ではない）（図 6.1.1）．

勿論全ての OpenSim データを Unity 用に変換できる訳でありません．スクリプトは当然の事ながら無理です．その他の変換の可否を図 6.1.2 に示します．

また OAR Converter を用いた変換の様子は https://www.youtube.com/watch?v=LUhqNJHLCuM で見ることができます．

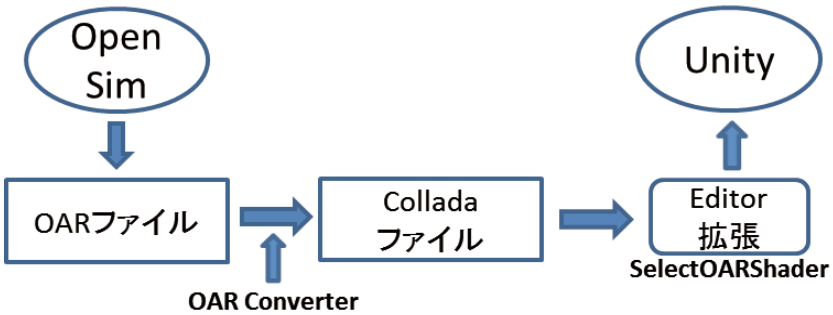


図 6.1.1 OAR Converter の構成

変換対象	変換	注意事項
Prim	○	
Sculpted Prim	○	
Mesh	○	
Tree/Grass	△	Linden Treeの生成アルゴリズムは使用しない
Flexible Prim	△	柔軟性のパラメータはサポートしない
Terrain	△	テクスチャの解像度が低い
Avatar/NPC	×	
Water	×	
Script	×	

図 6.1.2 データ変換の可否

6.2 OARConvWin

OAR Converter は Linux 上のコマンドとして実装されたため，Linux の知識がないと中々使い辛いシステムでした．そこで，これを Windows 用の UI でラップしたのが OARConvWin (http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?OARConvWin) です．最新版は v1.0.7 で

http://www.nsl.tuis.ac.jp/xoops/modules/d3downloads/?page=singlefile&cid=7&lid=47

からダウンロードできます．また簡単な操作方法についても上記 Web ページ上で解説しています．

6.3 データ変換の手順

6.3.1 OAR ファイルの作成

OAR ファイルを作成するには，OpenSim のコンソール画面のプロンプトから［save oar ファイル名 .oar］のコマンドを入力します．作業メッセージの出力後，OpenSim の bin フォルダにファイルが生成されます（図 6.3.1）

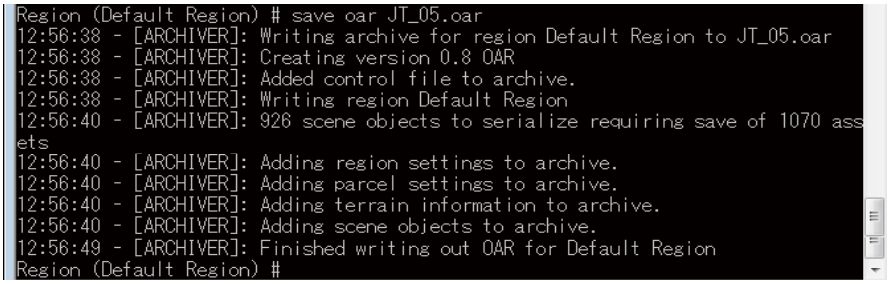


図 6.3.1 save oar コマンド

6.3.2 OAR ファイルの読み込み

OARConvWin を起動し，ファイルメニューから「OAR ファイルの読み込み」を選択して（図 6.3.2），OAR ファイルを読み込みます．OAR ファイルを読み込むと，自働的にファイルの展開が始まり，展開結果を OAR のファイルのファイル名の先頭に OAR_ の文字を付けたフォルダに保存し，その後リージョンの情報ダイアログを表示します（図 6.3.3）．例えば，JT_05.oar と言う名前の OAR ファイルを読み込んだ場合は，展開結果は OAR_JT_05 と言う名前のフォルダに保存されます．

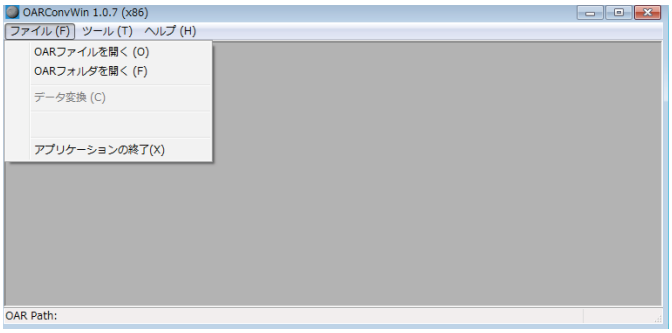


図 6.3.2 OAR ファイルの読み込み

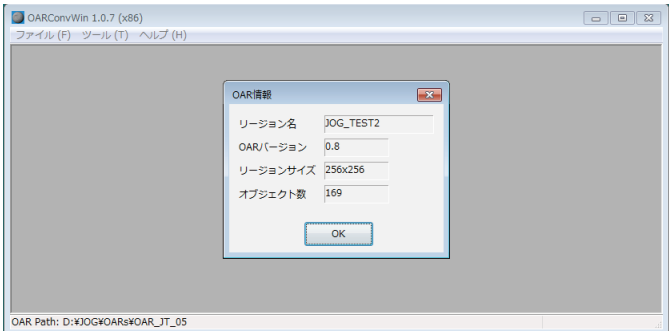


図 6.3.3 OAR ファイルの読み込み完了
ステータスラインに，展開先のフォルダ名が表示される

6.3.3 Collada 形式への変換

OAR ファイルの読み込みが完了すると，ファイルメニューの「データ変換」がアクティブになりますので，これを選択します．なお，変換前にツールメニューから「ログウィンドウ」を選択し，さらに「設定」から設定ダイアログを表示させて「デバッグモード」にチェックを入れておくと，データ変換の進捗状況がログウィンドウに表示されます（図 6.3.4）．

変換結果は OAR のファイルのファイル名の先頭に DAE_ の文字を付けたフォルダに保存されます．例えば，JT_05.oar と言う名前の OAR ファイルを読み込んだ場合は，変換結果は DAE_JT_05 と言う名前のフォルダに保存されます．

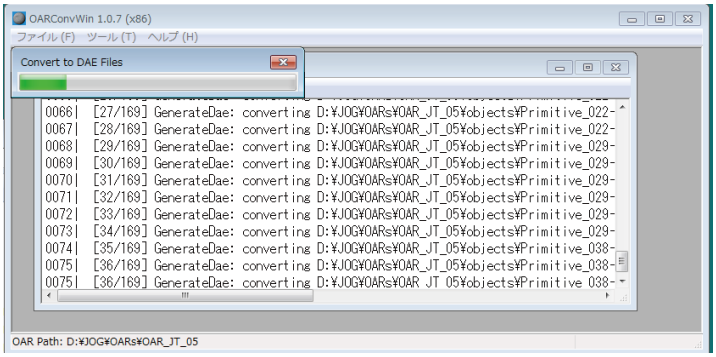


図 6.3.4 Collada 形式へのデータ変換

6.3.4 Unity でのデータの読み込み

Uniy を起動し，空のプロジェクトを作成します．OARConvWin と一緒に配布しているデータのうち，forUnity3D フォルダの中にある Editor フォルダを，Unity の Project View にドラッグ＆ドロップします．次に OARConvWin で作成した DAE_... フォルダ（例えば DAE_JT_05）も Unity の Project View にドラッグ＆ドロップします(大抵の場合，この処理にはかなりの時間が掛かります)．

Unity への DAE_ ファルダのコピーが終了したら，コピーした DAE_ フォルダを開いて中の（フォルダ以外の）ファイルを全選択し，Hierarchy View へドラッグ＆ドロップします（図 6.3.5）．

続いて，DAE_... フォルダの中にある Phantoms フォルダを開いて，同様にしてファイルを全選択し，Hierarchy View ドラッグ＆ドロップします．

これで，基本的なデータ変換作業は終了です．

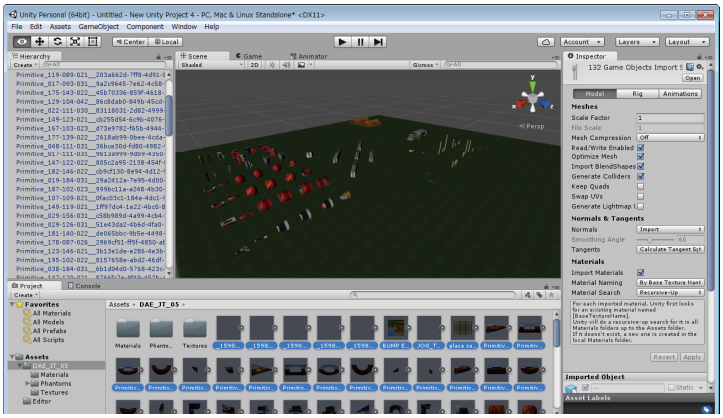


図 6.3.5 Hierarchy View へのドロップ＆ドラッグ

6.4 変換結果

図 6.4.1, 6.4.2 に OARConvWin による変換例のスナップショットを示します. また JOG の Web サイトである https://www.jogrid.net のトップページには,「Unity ちゃんと JOG を巡ろう」と題して Unity の Web Player 用変換したデータへのリンクを用意しています. 是非ご覧ください.



図 6.4.1 OpenSim 内での東京情報大学



図 6.4.2 Unity 上に読み込んだ結果

8. 終わりに

力及ばずと言うか予想通りと言うか，結局予定していた内容の半分くらいの内容になってしまいました．予定通りに進まなかった部分は当然の如く技術編と応用編であり，この本がそれなりに成立しているのは，ひとえに JOG のユーザのみなさんによるご投稿のおかげであります．原稿をご投稿頂きました JOG ユーザの皆さま，誠にありがとうございました．m(__)m

特に motoko さんには体調不良の中，ご推敲をお願いしていたのかもしれません．改めて感謝の意を表すると共にご冥福をお祈りいたします．

さて，この本の続編があるかどうかわかりませんが，続編があることを期待（？）しつつ，いつかまたお会いしましょう．(^^;



え〜．出足りないよう！



ちょっと中途半端じゃない！



今度のまたコミケに当選したら，きっとすぐに改訂・増補版が出るよ！
今度は印刷本だね！
先生！ 頑張ろうね !!

いや，実際第 2 版はコミケ本なんだが ...

OLD Programer はオーバーヒートで既に機能不全に陥っています！



OpenSim 入門 その2 第2版

2016 年 12 月 29 日　第 2.0.0 版

日本メタバース技術研究会
Japan Open Grid
JOG ユーザ会
東京情報大学 ネットワークシステム研究室 井関ゼミ
キャラ原案・イラストレーション 「サミッティー」

連絡先
http://www.nsl.tuis.ac.jp
http://www.jogrid.net