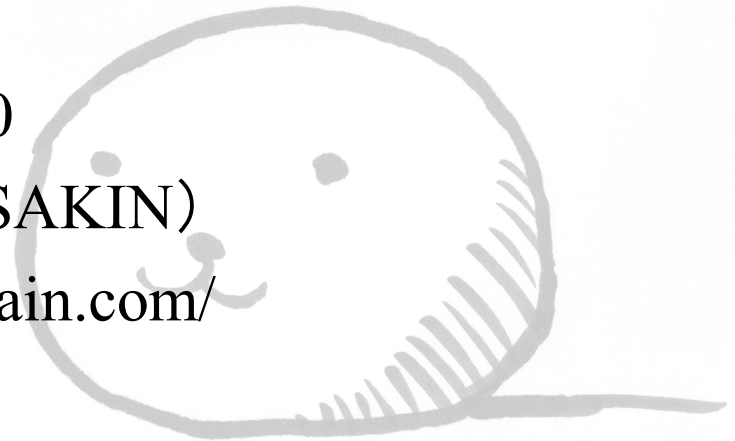

SITER SKAIN的 シューティングの実装について

神威・RefleX・ALLTYNEX Secondのお話

2010/02/20

じるるん(SITER SAKIN)

<http://www.siterskain.com/>



講演者

じるるん

サークルSITER SKAINの代表

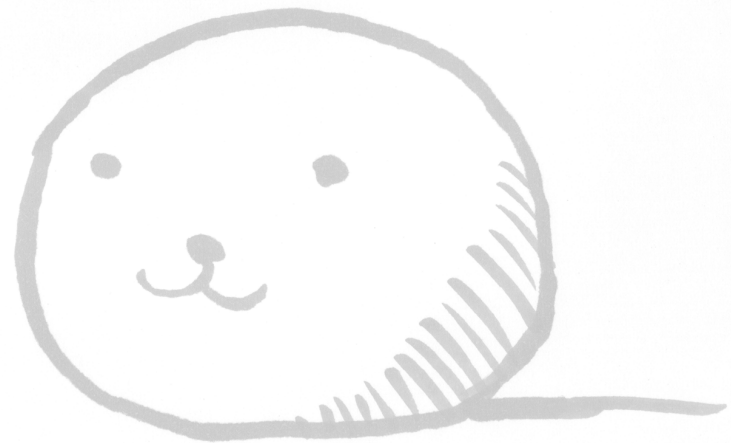
(一応)プログラム担当

その他、グラフィック、音楽、モデリング
デザイン、雑用、何でも担当

趣味でゲームを作り続けています。



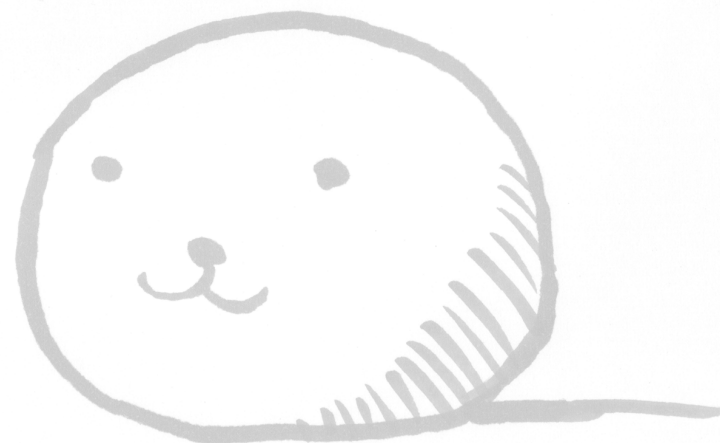
1. SITER SKAINの紹介



シタースケイン
SITER SKAIN

- 大学時代の友人で1999年に結成
- ゲーム企画者が自分でプログラム
- 個人製作の延長的な開発体制
- お手伝い感覚でなんとなくな作業分担
- メンバーは現在3名。全員プログラムがメイン
 - じるるん・・・ 神威、ALLTYNEX Secondの全般
RefleXの絵と音楽
 - ヤスウェア・・・RefleXのゲームデザイン・プログラム・絵
 - バーナス・・・ プログラムの技術的なサポート

2. SITER SKAINの ゲームの紹介



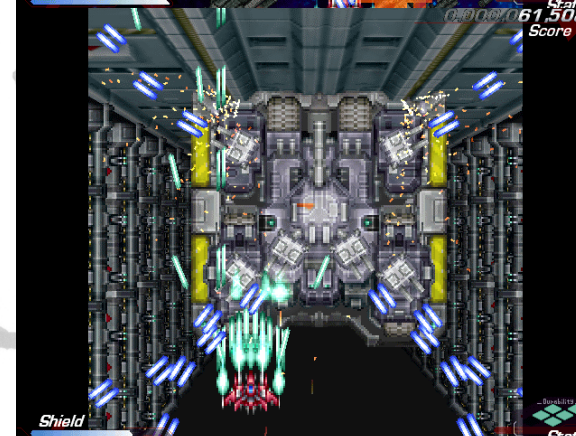
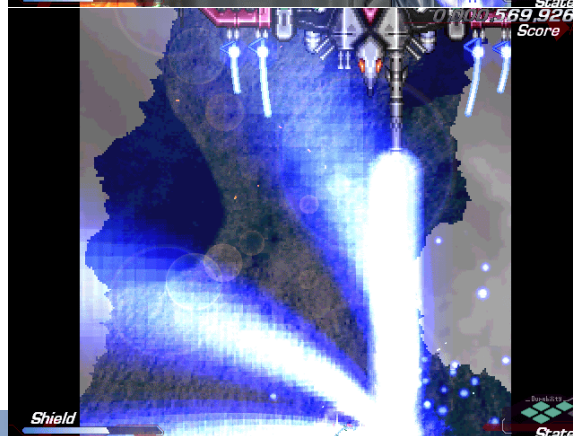
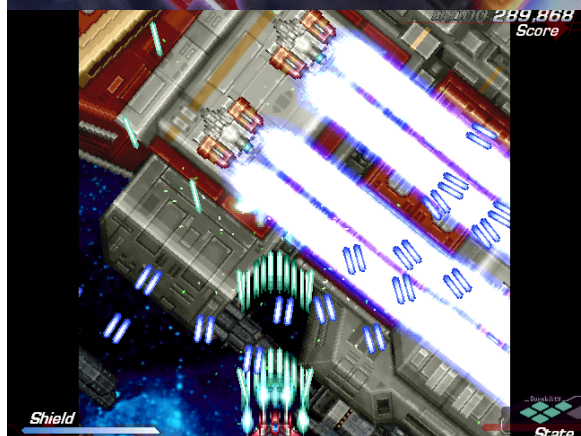
カムイ
神威(1999年12月完成)

- 拡大縮小描画により高低差の表現
- 低高度の敵機を自動照準する「雷撃」による倍率システム
- 徹底した作り込みと多彩な演出で展開する全6ステージ



リフレクス RefleX (2008年8月完成)

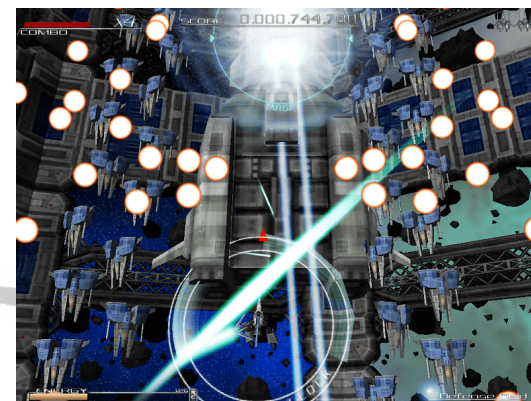
- 反射シールドシステムの先駆けReflection(1997)の正統続編
- 光学兵器を反射して攻撃する特殊シールドを実装
- ゲーム展開と連動した数々のシューティング的な演出



アルティネクス セカンド

ALLTYNEX Second(2010年8月完成予定)

- 初のフルポリゴン作品
- 2つの形態(アーマー形態・ファイター形態)
4つの武器(通常弾、ブレード、誘導レーザー、バスターライフル)
これらを駆使して戦う縦スクロールSTG



3. SITER SKAIN昔話

個人ゲーム製作開始～サークル結成まで
(1992年頃～1999年)



高校時代以前(～1994春)

ツクール作品「RAID WIND」作成

(開発:じるるん、その他)

- パソコンは持っていないがゲームは作りたい
- 友人宅のPC9801+バカスカウォーズで共同製作
- PCエンジンやMDのコンシューマSTG(移植含)の影響



- 演出重視、全9面
- ライバル機がしつこい
- 雑誌「LOGIN」の「未確認クリエイターズ」というコーナーに投稿
- 敵配置のテンポが評価
- 掲載されただけで賞とかは貰っていない

大学時代1年目~2年目前半

(1994年春~1995年秋)

プログラムの勉強

- ついにパソコンを購入(FM TOWNS)
- ゲームを作るためにプログラムを勉強
- 最初に覚えた言語は F-BASIC386
- 1年で小物の習作4本作成
 - ・2分間STGとか、そういった類の物
- 2年目にC言語を勉強
 - ・習作でまたしても2分間STGを1本作成

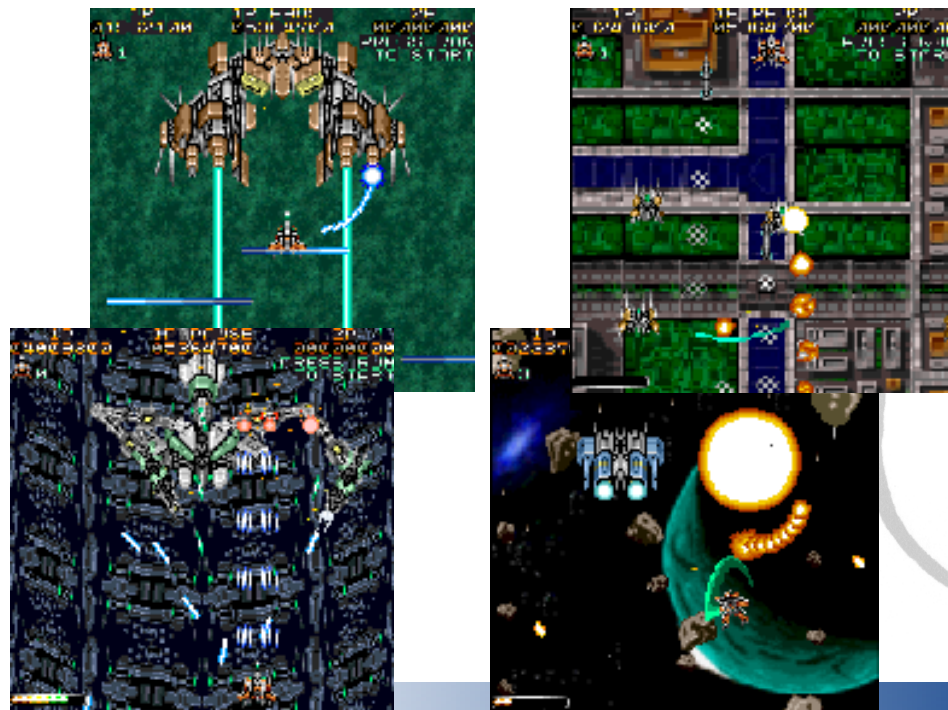


大学時代2年目～3年目

(1995年冬～1996年冬)

「ALLTYNEX」の開発期間(開発:じるるん)

- 業務用並を目標に「ALLTYNEX」開発開始
- 開発期間、約1年9ヶ月。C言語が約99%
- レイフォース・サイバトラー・ハイパーデュエル等の影響

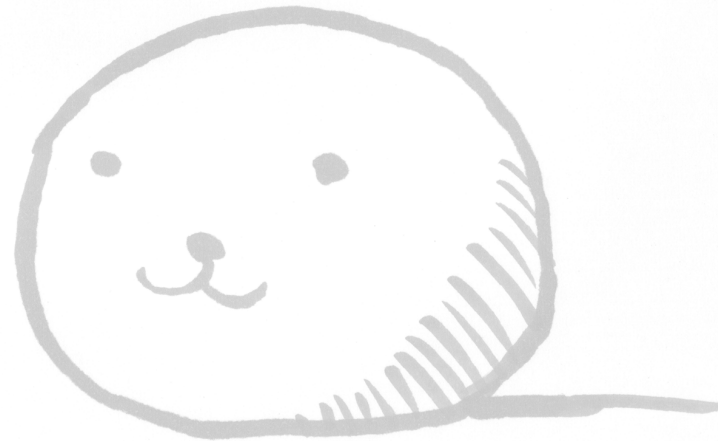


- 全5面の縦STG
- 2ボタンで4つの武器
- ビームサーベル超重要
- 業務用STGぽい緩急ある敵の動き
- 60FPS維持しつつ拡大多重スクロール
- 第2回Aコンで敢闘賞 12

大学時代3年目冬～ (1996年末～)

Windows環境への移行

- Windows95大人気
- ゲームも今後はWindows95で開発するべき
- 神威の開発開始(じるるん)
- Reflectionの開発開始(ヤスウェア)



大学時代3年目～4年目

(1996年末～1997年秋)

「Reflection」の開発期間(開発:ヤスウェア)

- 言語はDelphi2.0。DirectX未使用
- ギガウイングより1年以上前に敵弾反射シールド実装



- シールドを使った敵弾の反射システム
- 敵の攻撃がそのままはね返る点が重要
- 全3面+隠しラスボス
- ラスボス超弾幕は怒首領蜂を見て急遽追加
- エンターブレインのコンテストパークで受賞

当時の大学の友人達は
雑誌等への投稿がメイン

コミケに行った事もなかったため
同人ソフトという物をあまり知らなかった

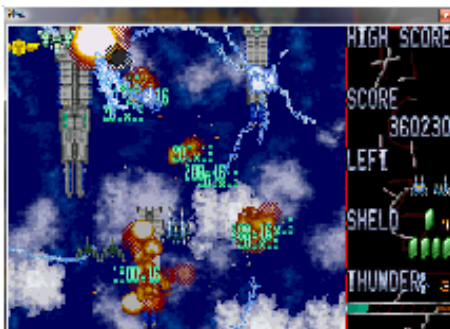


大学時代3年目冬～春

(1996年末～1997年春)

「神威」のプロトタイプ開発(開発:じるるん)

- 言語はDelphi2.0。DirectX未使用
- レイフォース・蒼穹紅蓮隊・サンダーフォースV等の影響
- C++Builder発売→ Delphi版神威の開発を中止



- 高低差の概念を採用
- 全画面に対する自動ロックオンシステム
- 拡大縮小描画が無いため、奥行にあわせて絵のパターンを4段階用意

大学時代4年目～社会人2年目

(1997年春～1999年冬)

「神威」の開発期間

- C++Builder発売
- 高速描画ライブラリWinGLとの出会い
→ Delphi2.0から移植作業
- 大学4年の夏に初めてコミケに一般参加
- 社会人になって開発ペースが極端に低下
- コツコツと開発を続けて、1999年冬、ようやく「神威」完成



1999年冬のコミケ57申込みにあわせて
サークル「SITER SKAIN」を結成



カムイ

4. 神威の開発



(1997年頃～1999年冬)
じるるん: 企画・プログラム・絵・音楽
OPとEDの絵は一部友人に依頼

神威(コンセプト)

- Windows95環境への移行
- 前作ALLTYNEXでは限界があった、2Dでの奥行き表現
- ゲーム内の世界観、空間など感じられるように
- 多パーツ、多関節、パーツ破壊、やりたい放題
- 業務用レベルの敵の動きを
- 既存作品とは別方向へのロックオンシステムの発展
- 演出は超重要
- 描画に関してはWinGLに全て任せる
(完成後に自力描画によるエフェクトを少し追加)

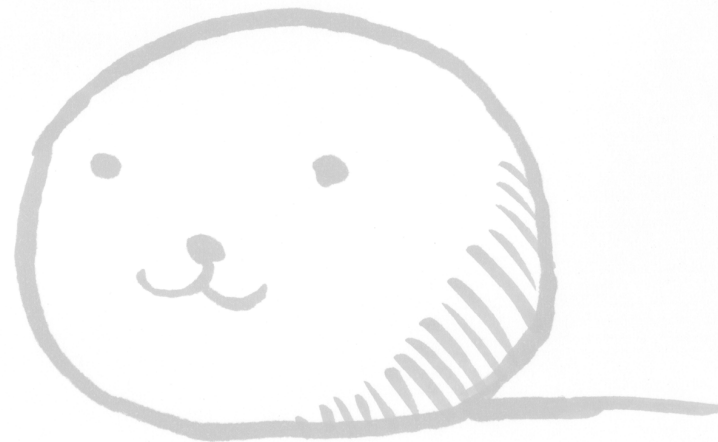
神威

仮想ハードウェアスペック

- マップチップ形式BGプレーン3枚
(拡大縮小には未対応)
- サイズ無制限スプライト(2048個)
- ウィンドウサイズ:640x480基準
- 内部描画解像度は240x240(速度重視)
- フレームレート可変方式
(但し2フレーム連続スキップはしない)
- 画面モードは256色

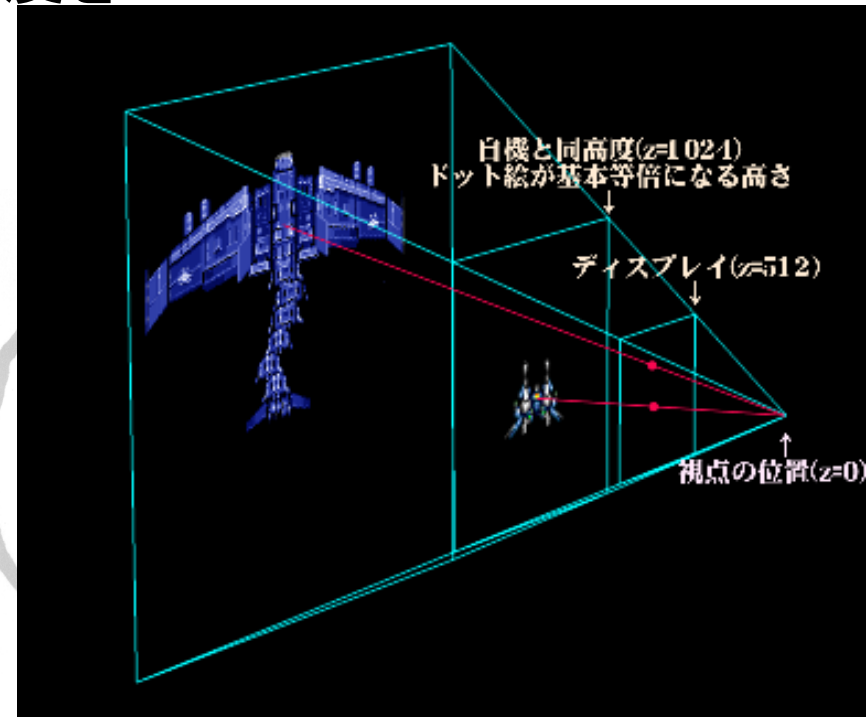


画面内に立体的な空間を作るために



神威(実装) 表示システムの構築

- スペースハリアー的ゲームが作れる、シンプルな拡大縮小描画使用の奥行表示システムを構築
- カメラ位置を $Z=0$ 、自機の高度を $Z=1024$
- カメラの向きは固定
- 奥に行くほど Z 値は増加
- 空間の回転には対応しない



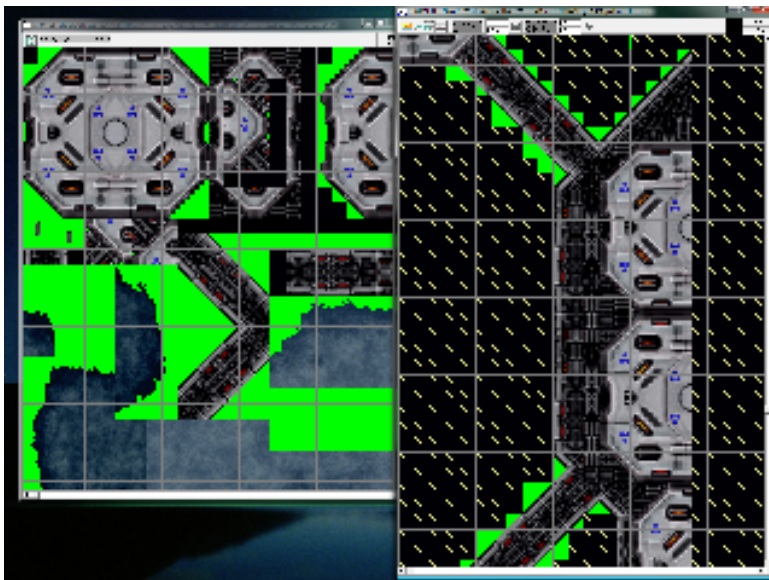
神威(実装) オブジェクト(敵キャラ)の管理方法

- 演出が重要なため、内部的にあらゆるオブジェクトが同列の扱い
- 自機、敵、敵弾、背景の厳密な区別はない
 - 例：自機は「操作可能な敵」
 - 背景は「当たり判定がない敵」
 - OPやEDの絵や文字も敵と同じ
- 無駄なループや判定は増加
- キャラ制御の自由度は上昇



神威(実装) マップの管理方法

- マップチップ形式のBGプレーン × 3枚
- 背景のループ処理、フラグ制御はプログラム側で制御
- マップエディタはマップチップの配列データ作成のみ
- 敵機が背景の下にもぐりこめるよう、自由にZソート可能



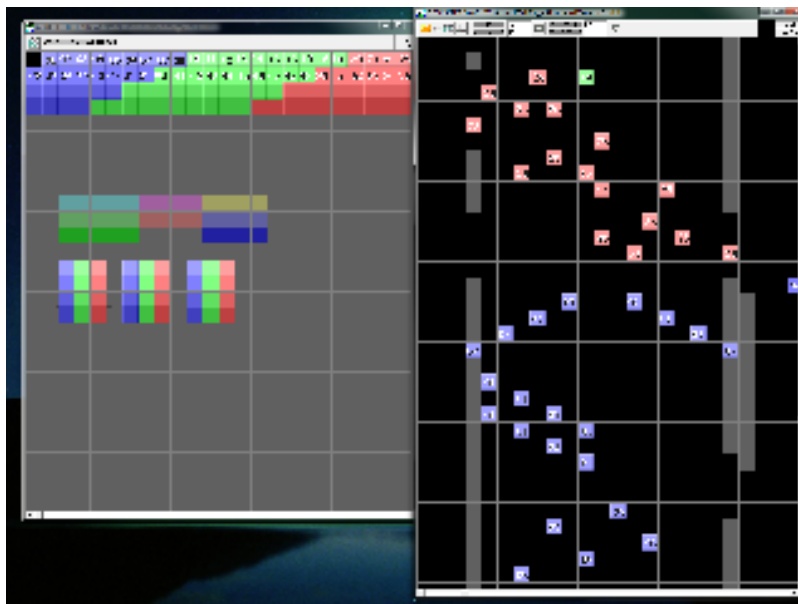
実際のゲーム画面



エディタ自体にスクロール制御の管理機能なし
基本的なチップ配置、範囲コピーペースト程度の機能
ツールのマップエディタの再現をイメージ

神威(実装) 敵の配置について

- 敵の配置にもマップエディタを使用
- 配置後、実際にゲームを動作させてタイミングを調整(ひたすらトライアンドエラー)
- 場合によっては、プログラム内部で敵を配置



- マップエディタで敵配置も行う
- 背景との連動なし。奥行きの設定などもナシ。
- プログラムでは一定カウント毎に次の列の配置データを取得

リフレクス

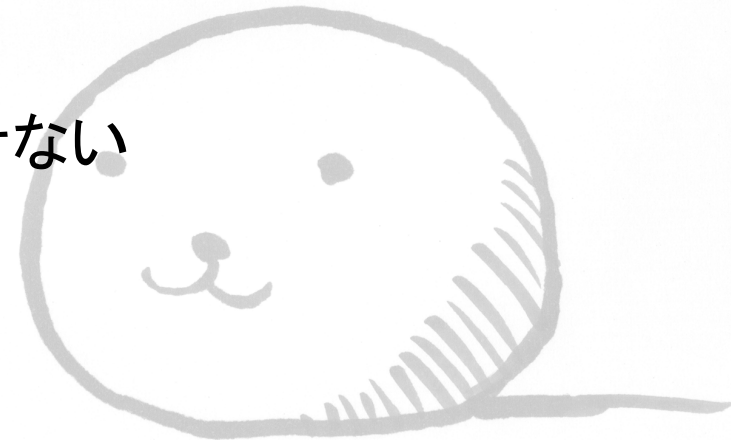
5. RefleXの開発



(1998年末頃～2008年夏)
ヤスウェア:企画・プログラム・グラフィック
じるるん:グラフィック・音楽

RefleX(初期コンセプト)

- DirectX使ってReflectionリメイク&続編
- シールドを使った敵弾反射システムは継承
- 前作と違い、シールド使用前提な敵の攻撃
- 弾幕の扱いは数ある攻撃方法の一種類
- 16bitカラーで256環境とサヨナラ
- 綺麗な半透明処理が可能に
- 背景演出とかもちろん欠かせない



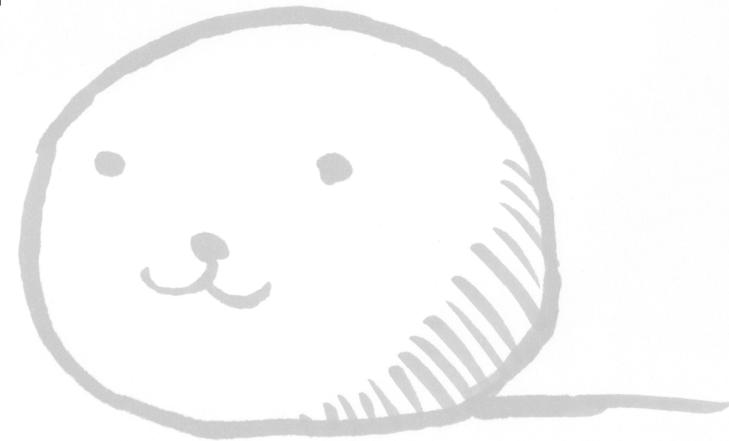
RefleX(方針転換と迷走～2002年頃)

- 開発当初(1998)、DirectDraw+自力半透明。超遅い
- 世間が徐々にDirect3Dスプライト描画に移行
- 1999年の最初の体験版でDirect3D使用に
- 世の中のPCスペック不足、特殊環境の問題
 - ・ビデオカード2枚刺しが普通にありえる
 - ・テクスチャサイズの制限が小さい
 - ・加算半透明がない、スペキュラがない
 - ・そもそも3D機能がない、等



RefleX(開発停滞～2004年頃)

- ハードウェア依存による動作環境の問題
→ゲーム本編より基礎部分の調査と修正が優先
- コミケの連続落選
- コミケに申込まなかった回も
→実質1年半ほど開発が停滞



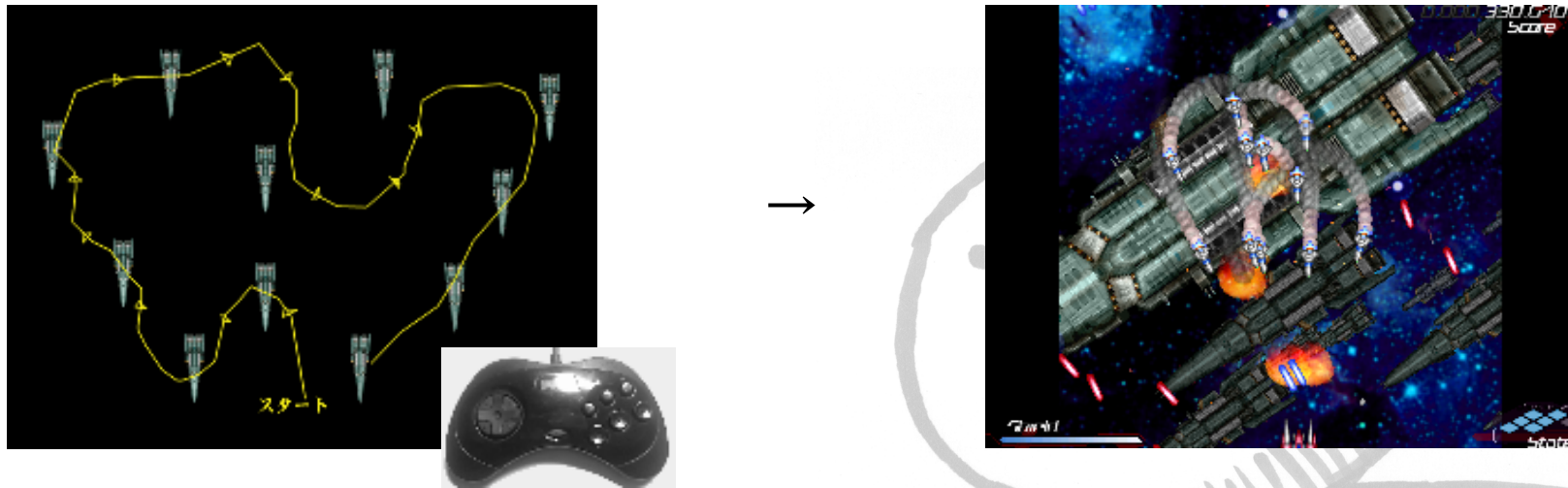
RefleX(～2008年完成まで)

- 飛躍的にPCのスペックが向上
- 2004年あたりで吹っ切れて、使用VRAM制限を64MBくらいにアップ
- テクスチャサイズも1024x1024まで許可
- レーザー等もパターン描画から頂点指定描画に
- 際限がなくなるため、作業にリミットを設定
 - ・3Dモデルデータは使用しない
 - ・アルファビット付テクスチャは使わない
- なんだかんだで2008年にやっと完成しました。

RefleX(実装)

Area3 艦隊戦ステージのスクロール制御

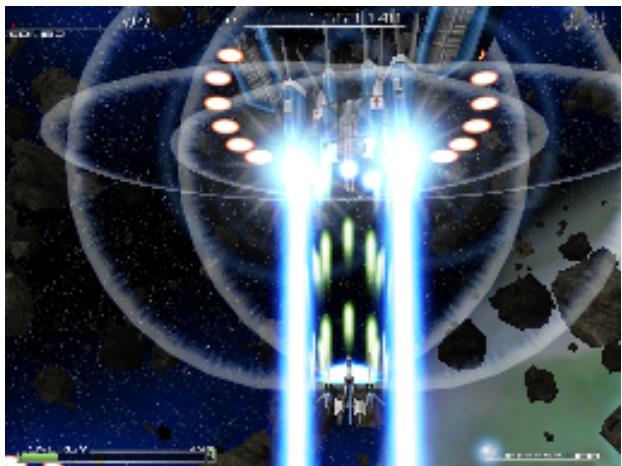
- 1. マップエディタで、広い宇宙空間に戦艦を配置
- 2. 空間を自由に移動できるプログラムを実装
- 3. その操作キー入力情報を保存
- 4. 保存した入力情報をリプレイ再生 → 背景スクロール



コントローラーで戦艦の配置された空間を移動

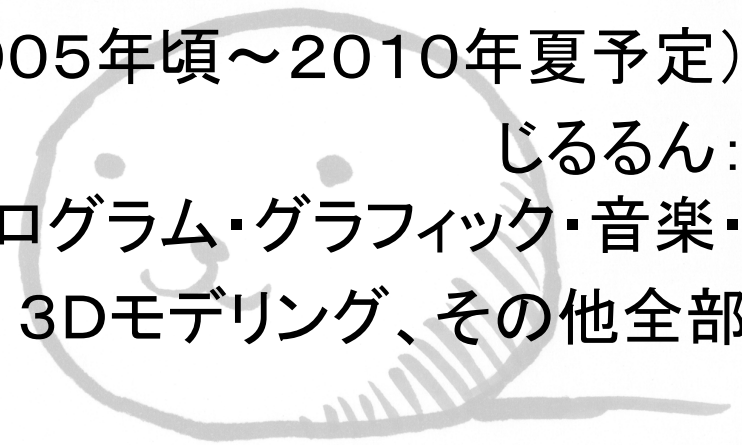
アルティネクス セカンド

6. ALLTYNEX Secondの開発



(2005年頃～2010年夏予定)

じるるん:
企画・プログラム・グラフィック・音楽・
3Dモデリング、その他全部



ALLTYNEX Second(コンセプト)

- FM TOWNS版ALLTYNEXの続編を3Dで
- ALLTYNEX 3部作(神威、RefleX、ALLTYNEX 2nd)をWindows上で完結させる
- 例によって業務用レベルのものを
- 3Dの勉強、C++の勉強、モデリングの勉強、色々勉強
- 2Dでは出来なかった敵の動き
- 途中から画面を横長に
→4種類の武器の使い分けをより有効に活用させる
- 3Dならではのリプレイ再生とか



ALLTYNEX Second (技術調査2000年～)

- 自分でDirectX使う(当時DirectX7)
- 3Dの基礎の勉強
- モデリングの練習
- 3Dゲーム用のフレームワーク構築
- 但し、RefleXの開発が優先
- DirectX7版を破棄し、DirectX8で書き換え
→現在はDirectX9を使用



ALLTYNEX Second (2005年、初公開)

- 他の同人ソフトサークルのレベルが上がりまくり
- なのに当サークルではまだRefleXが未完成
- 痺れを切らせ、RefleXを放っておいてALLTYNEX Secondのプロトタイプ版をお披露目(2005年夏)

→この時点では
物凄く惨い出来



ALLTYNEX Second (2006年頃)

- 内部処理の作り直し
 - ・3Dについて無知だったため、オブジェクトの回転をXYZのオイラー角指定でMATRIX合成していた
 - ・逆行列を自分で計算していたが、合成のタイミングを間違っており、板ポリで親子関係をつくと明後日の方向に
→思ったように制御できていない
- ヨー、ロール、ピッチでの回転指定
- D3DXの用意している関数を使用
→やっとまともに任意軸回転できるように

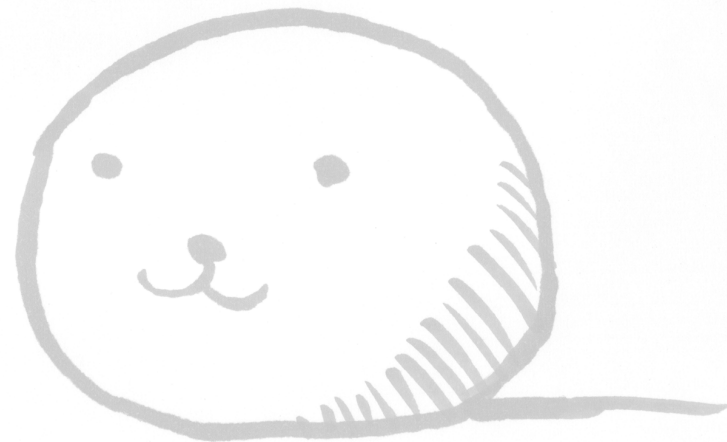


ALLTYNEX Second (2006年～2009年冬)

- 3D難しい……
- 流行の3D技術に全く付いていけないことを自覚
- とりあえず手の届く範囲の技術で完成を目指す
→技術的に現状以上のものは次回作に持ち越す
- 途中、RefleXの完成を本気で目指したため、開発を一旦停止
- RefleX完成後、開発再開、現在開発中

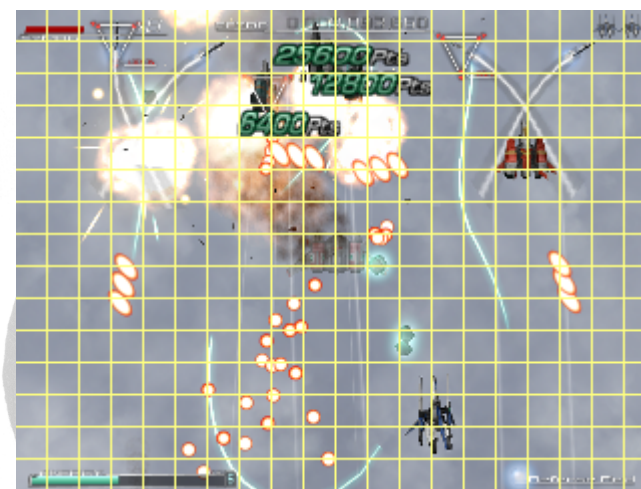
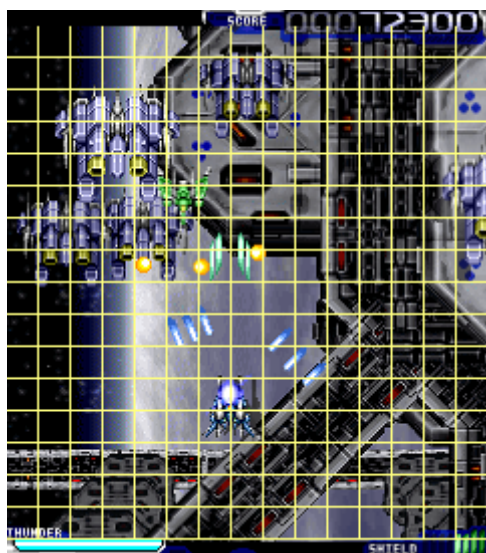


2Dから3Dへの変化による 内部的な考え方とか



ALLTYNEX Second(実装) 2D座標から3D座標へ

- 神威では、int型を8bitシフトした固定小数点演算
- ALLTYNEX Secondではfloat型を使用
- 神威の32ドット(マップチップ1つのサイズ)
→ALLTYNEXの1.0fのサイズになるようカメラを設置



サイズのルールを決める

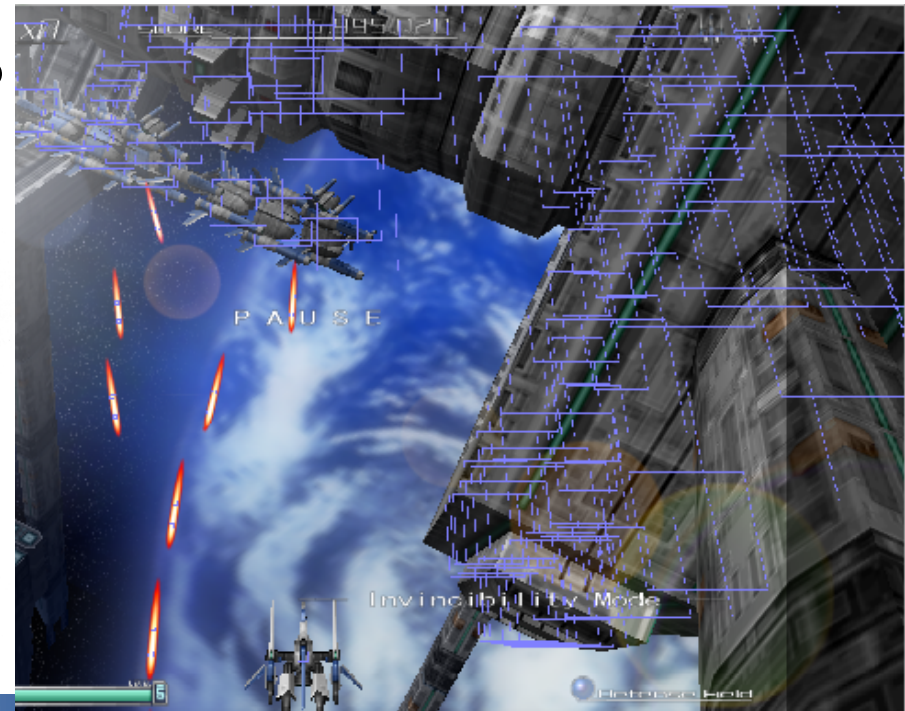
ALLTYNEX Second(実装)

データ手打ちモーション

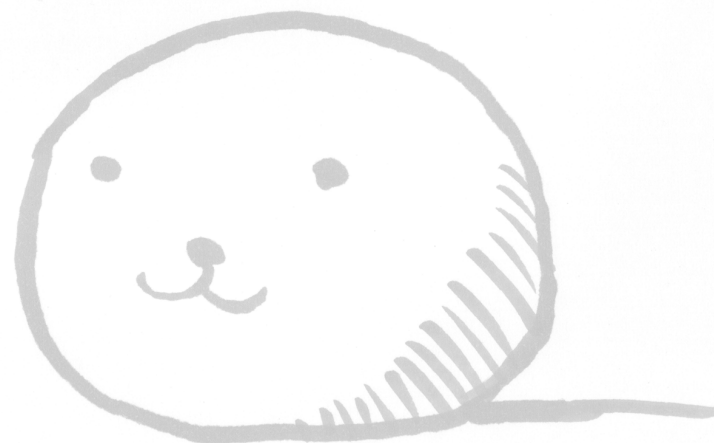
- モデリングにはメタセコイアを使用
- 2Dのアニメーションが3Dではモーションになる
- 3Dソフトでのモーションデータ作成は行わず、プログラム内でデータ手打ち
- モーションデータに中間キーフレームの概念は無い
- 始点と終点の単純な直行補間のみ
- あとは2Dの時と同じく、多パーツ、多関節処理
→3Dゲームの第一作目なのでこれ以上欲張らない

ALLTYNEX Second(実装) 当たり判定

- 当たり判定制御も2Dの延長
- 立方体、立方体同士の判定(単純な座標比較)
- 平面でない理由は、立体的な回転への対応のため
- 当たり判定データは手打ち



7. まとめ



SITER SKAINのゲーム開発姿勢

- 技術力が大したレベルでない事は承知の上
- そんな中でも工夫次第で案外いろいろ出来る
- 足りない技術は力技と作り込みでカバー
- 所詮アマチュア集団、ということで、多少クオリティが落ちても、絵も音楽もメンバー内で自分達が作りたいものを作る
- 作ることを楽しんでいればそれでいいじゃないか



ゲームを作りたいと思ってる人達へ

- 品質の善し悪しはともかく、自分で「プログラム」「絵」「音楽」をやってみる事をオススメ
- 「プロレベル」は無理でも、「同人としては十分」なレベルには大抵の人が到達可能
- 近年同人でもスクリプト化、ツール化など、開発環境が進んでいるが、それらは必須ではない。
- STGツールから入った人でもこのレベルくらいなら到達可能という1つの例



そんなわけで、
皆さんも、面白いシューティングゲームを
作ってください！！

ご清聴、ありがとうございました。

