



自作ゲーム『ラビリントス』の紹介

白石 玉斗

<https://gyokutosiraisi.github.io/my-site/index.html>

公開方法

現在（2025.2）

- ▷友人にテストプレイを依頼中
- ▷完成度90%程度
- ▷難易度調整などを行いリリースを目指す

ネット上で公開予定

- ▷2025年夏ごろを目途に完成版を公開予定
- ▷「ふりーむ」などに投稿
- ▷第17回ウディターコンテストにも登録予定

世界観・ストーリー

p.2

ゲームシステム

p.8

イチオシ！ 様々なゲーム性

p.17

快適&奥深い！ ゲーム体験

p.28

制作の経緯やツール、評価

p.41

※各項目、ハイパーリンクが機能しています。

世界観・ストーリー

p.2

ゲームシステム

p.8

イチオシ！ 様々なゲーム性

p.17

快適&奥深い！ ゲーム体験

p.28

制作の経緯やツール、評価

p.41

タイトル画面



主人公

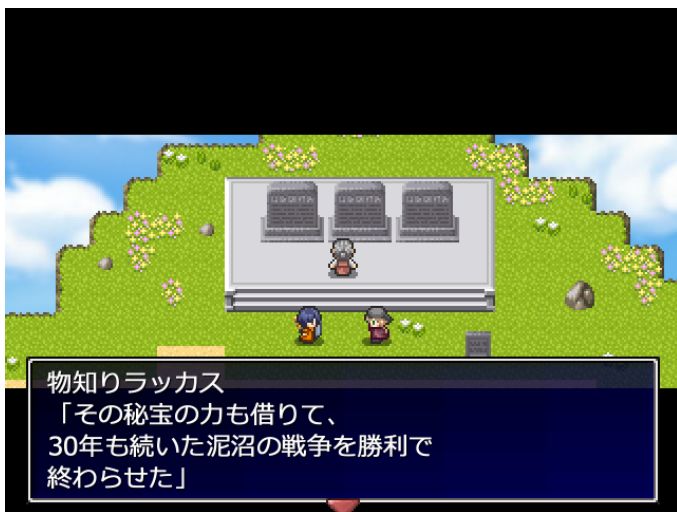


男女を選ぶことができる。名前も自由。



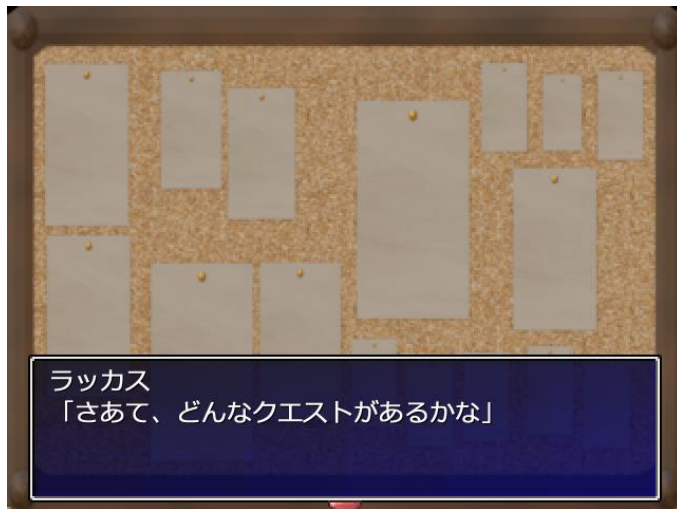
主人公の境遇

- ▷ 主人公は母親と小さな村で暮らす**青年**。
- ▷ **冒険者**になる為、幼馴染**ラッカス**と共に、ギルドに向かうところから物語が始まる。



歴史

- ▷ この国は数年前まで**戦争**をしていた。
- ▷ 突如、王が**謎の秘宝**を持ち出し、その力で戦争に**勝利**した。
- ▷ 戦争の**傷跡**はまだ各地に残っている。



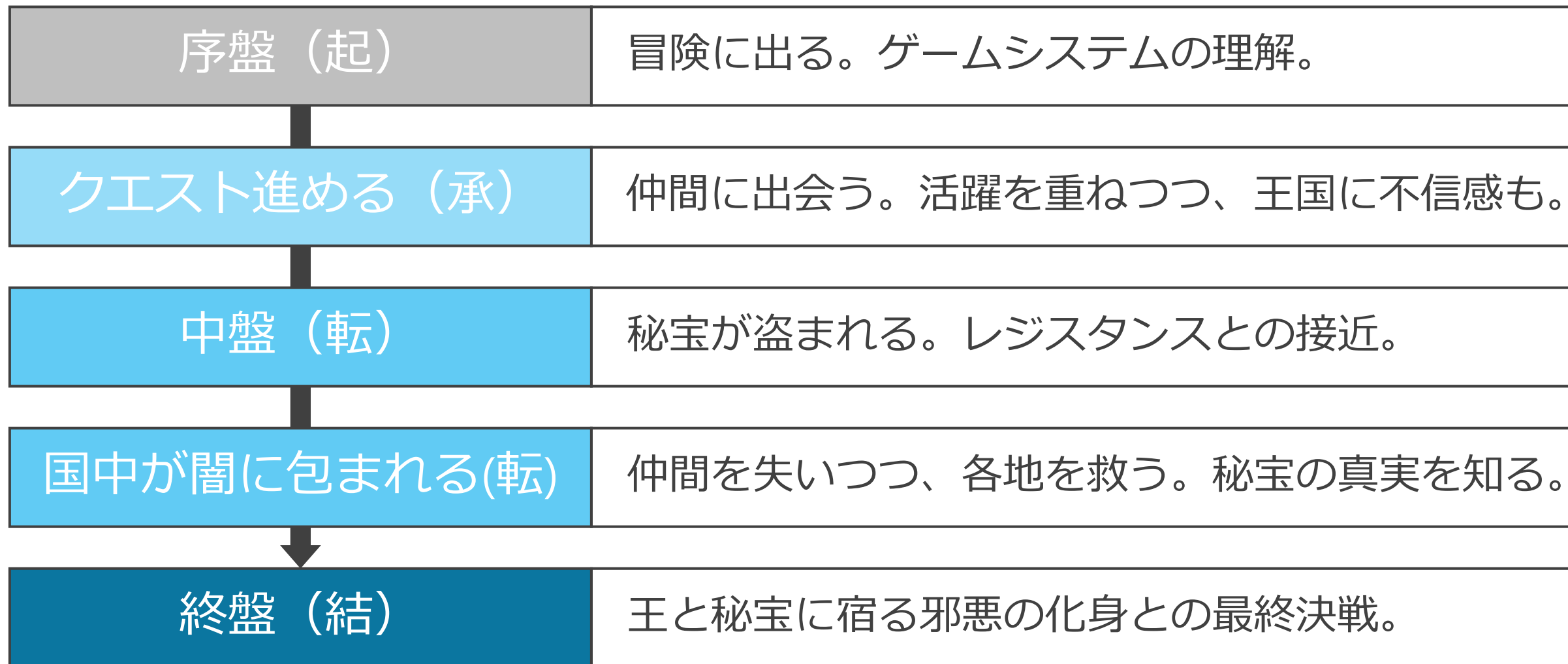
冒険の目的

- ▷ 冒険者の醍醐味は**クエスト**！
- ▷ **仲間**を集めつつ、広い世界を旅する。



ストーリー

- ▷ クエストをこなす内、**王国の秘密**に近づく。
- ▷ 王国に反抗する**レジスタンス**との出会い。
- ▷ やがて主人公らは禁足地「**ラビュリントス**」に向かうことに……。



世界観・ストーリー

p.2

ゲームシステム

p.8

イチオシ！ 様々なゲーム性

p.17

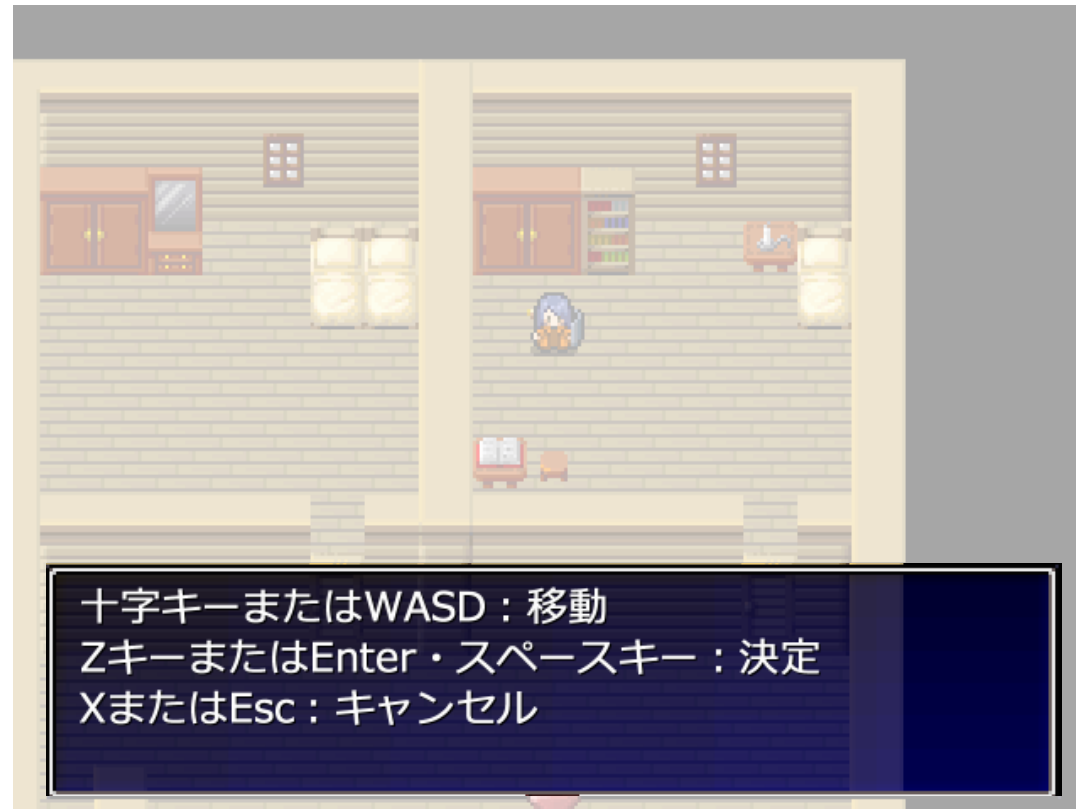
快適&奥深い！ ゲーム体験

p.28

制作の経緯やツール、評価

p.41

基本操作

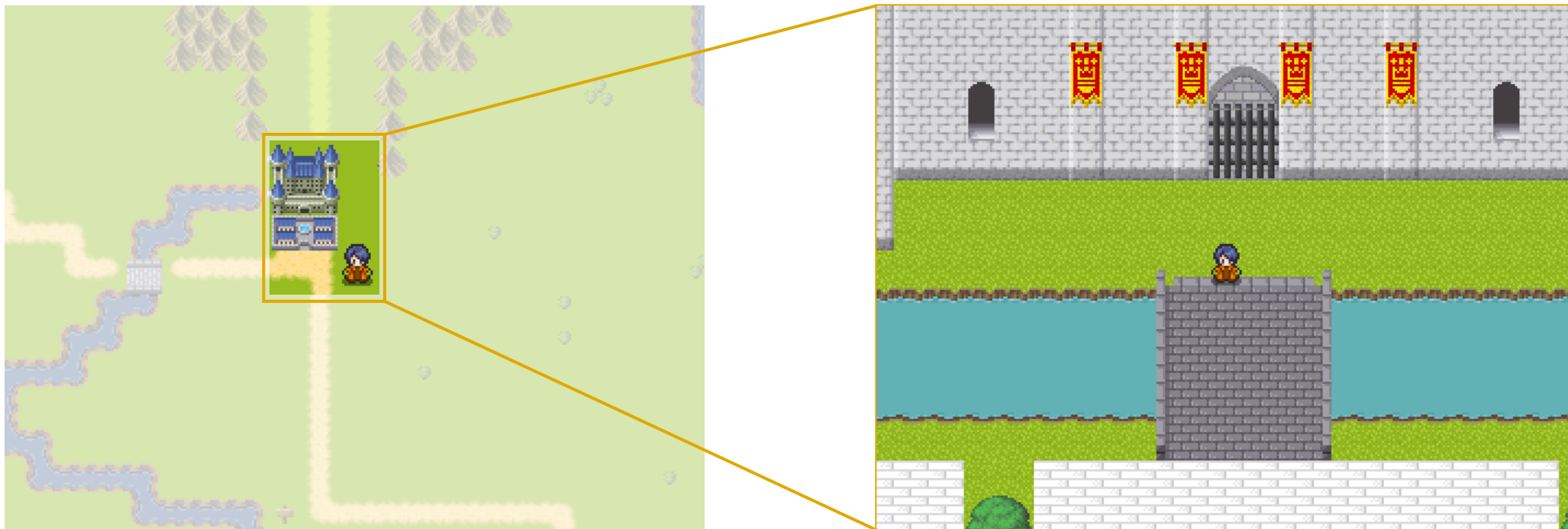


バトルシステム



古典的なコマンドバトルを採用

2Dフィールドマップ



フィールドマップから町やダンジョンに遷移する。

昼夜システム

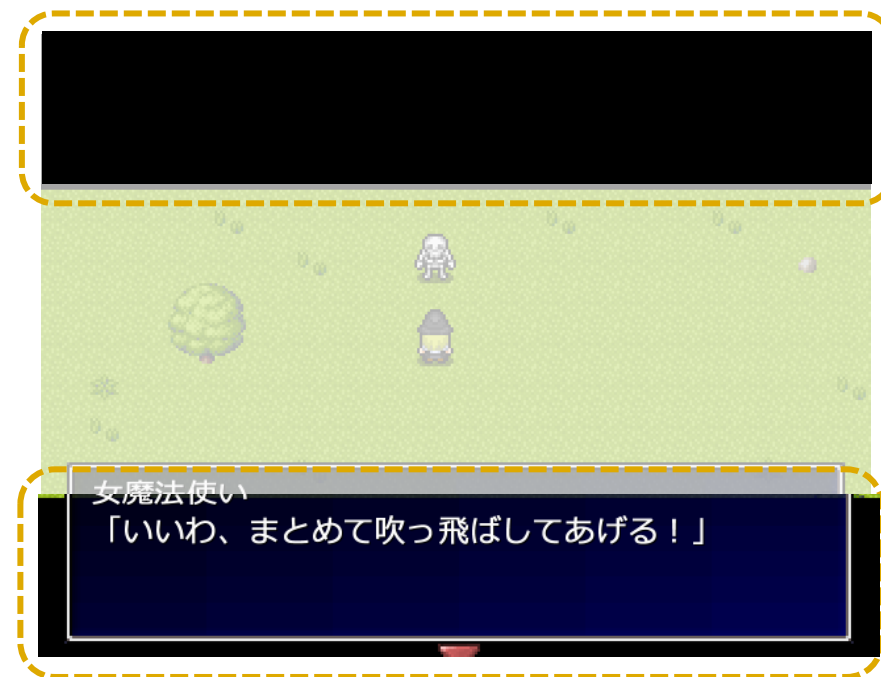


フィールドでの活動時間に応じて、昼夜が入れ替わる
夜しか発生しないイベントや敵なども

イベント



一部イベントは**イラスト**に。



黒幕が降りて、臨場感を演出。

ゲーム内イベント・ムービー



10秒～5分のムービーを豊富に用意！
単なるテキストではなく、**視覚的演出**も

コアループ

クエストを受注する

ダンジョン等を攻略する

達成度に応じた報酬

ストーリーイベント

ゲームの中核となる繰り返し
行われるプレイヤーの行動

探索&戦闘

重厚なストーリーと
ゲーム内ムービー

各種データ

文字数

総計**10万字**

キャラ

プレイアブルは**6人**

マップ

28（階層含め**50以上**）

イベント

メイン・サブ合わせて**50**

敵キャラ・技名

それぞれ**50種類**程度

プレイ時間

平均**11時間**（n=12）

世界観・ストーリー

p.2

ゲームシステム

p.8

イチオシ！ 様々なゲーム性

p.17

快適&奥深い！ ゲーム体験

p.28

制作の経緯やツール、評価

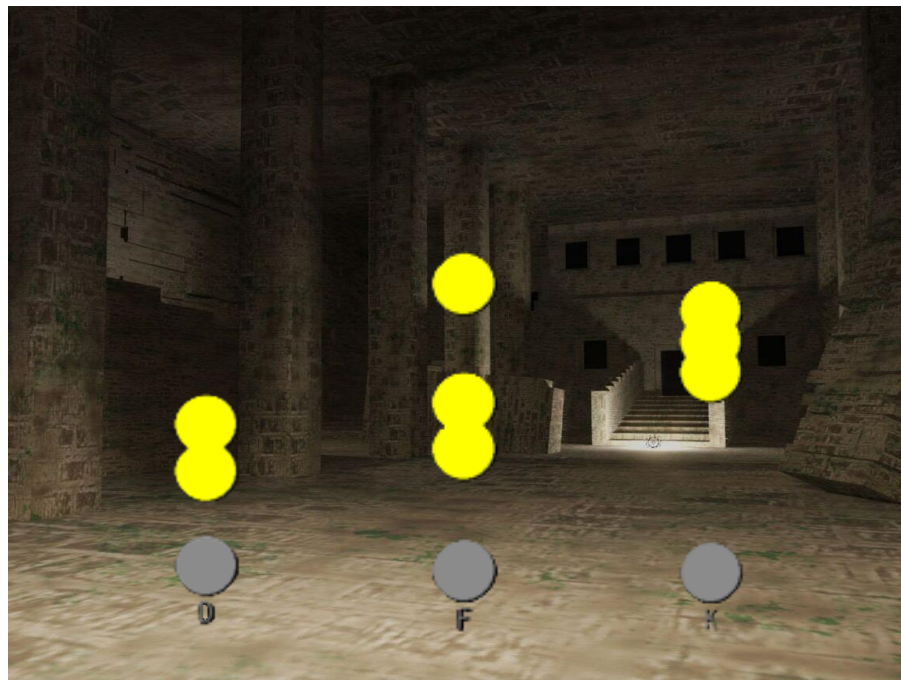
p.41

縦スクロールの障害物回避



中盤のボスである**ドラゴン**が突然町に現れ、主人公たちを襲う。
ここでは**町からの脱出**を目指し、ひたすら下に**逃げる**ことになる。

リズムゲーム



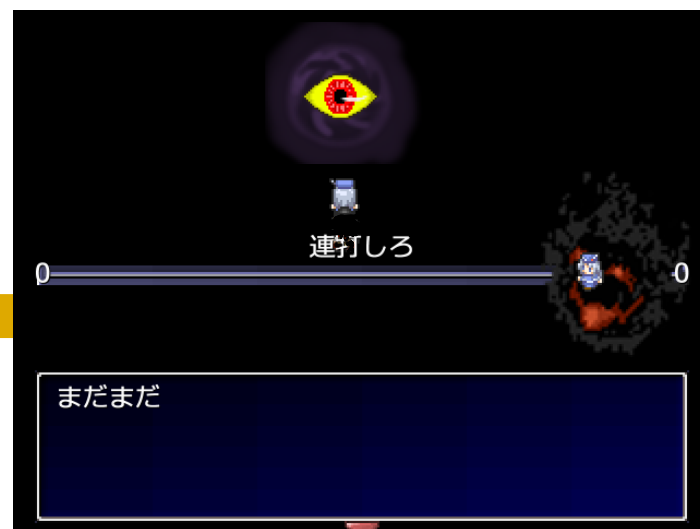
ダンジョン内のギミックのひとつ。
タイミングよく各キーを押すと、成功

QTE



妖精が仕掛けたトラップを回避するQTE。
赤カーソルが真ん中の時に指定された**キーを押す**と成功

連打アクション①



決定キー**連打**で**ゲージが左右**に
左に追い込まれると負け
右に追い切ると勝ち

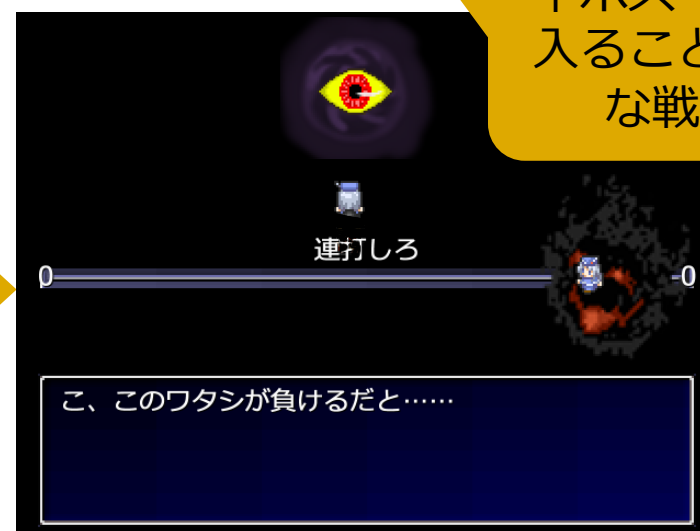
ゲージを押す強さが設定済
一度**追い詰めても**.....

敵の押す強さが数倍に
逆に押し込まれ、**ピンチを演出**

連打アクション②



主人公が押し込まれると、
ゲージを押す強さが**アップ**



中ボス・ラスボスでこの演出が入ることで、プレイヤーに重要な戦闘という認識も促す

接戦になるよう設計し、
プレイヤーに**緊張&満足感**

自由度の高い攻略



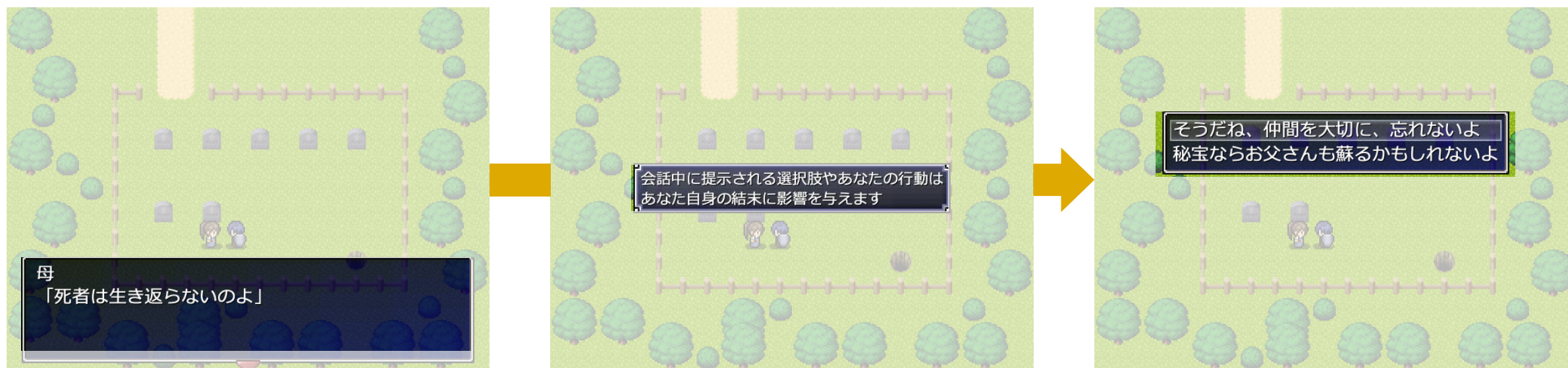
クエスト達成に際しては、必ず複数の**攻略法**！
現場や町であらかじめ**ヒント**の提示も

攻略度に応じた演出・報酬



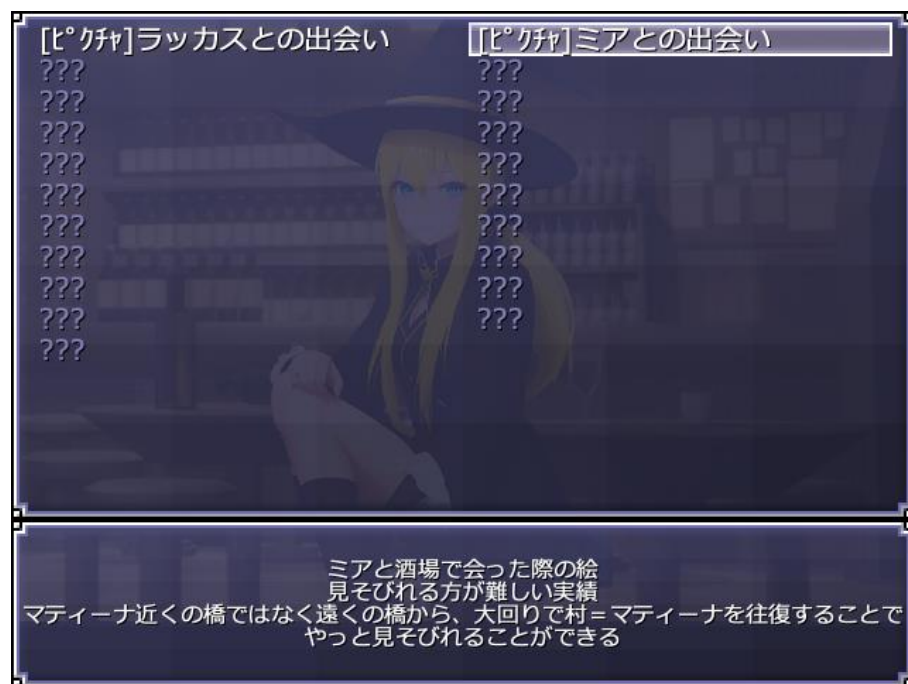
ストーリーや報酬が変化！
こうした成果はエンディングにも影響（次スライド）

マルチエンディング



主人公の選択によって内部数値（カルマ値）が変化
数値次第で**エンディングが分岐**する

イベントスチル



解放したイベント絵をメニュー画面から確認できる

収集要素



マップ中にある33個のハニワを集める**隠し要素**
全て集めると**アイテムが入手**できる

世界観・ストーリー

p.2

ゲームシステム

p.8

イチオシ！ 様々なゲーム性

p.17

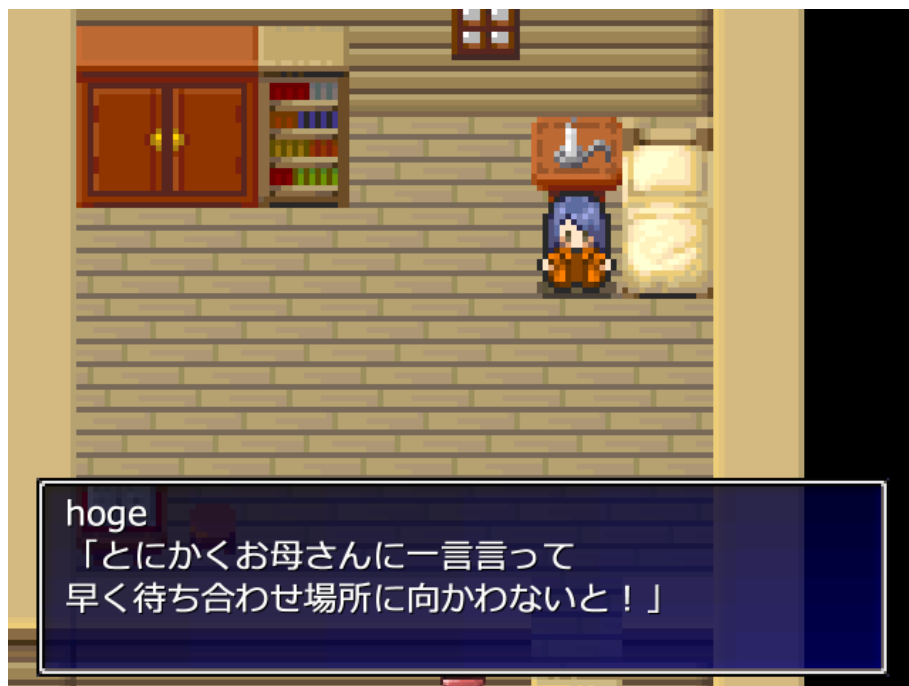
快適&奥深い！ ゲーム体験

p.28

制作の経緯やツール、評価

p.41

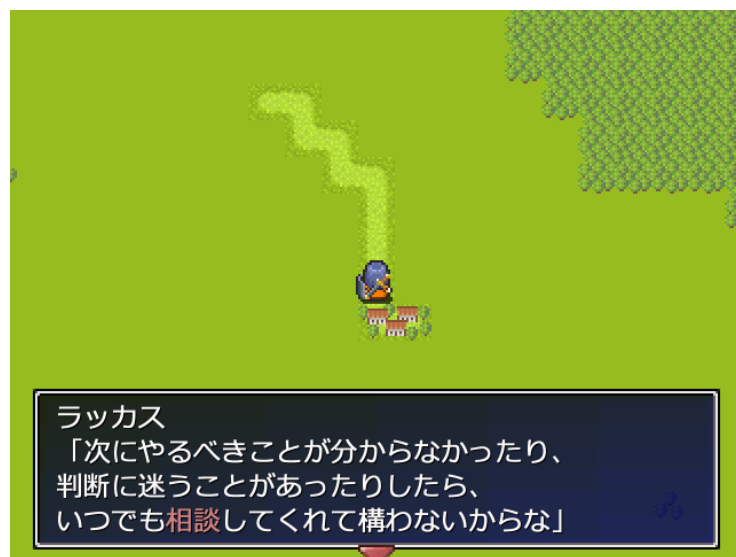
時間経過で行き先のリマインド



1分経過で
リマインド



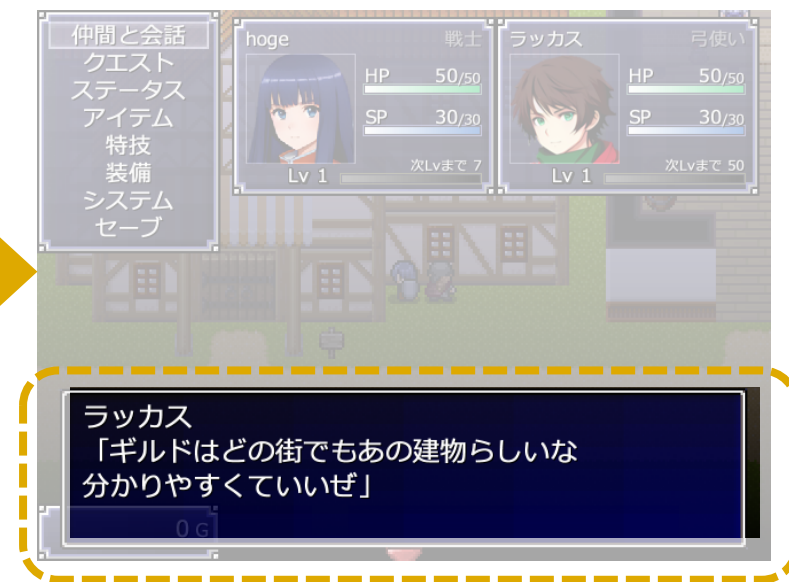
相談システム



相談コマンドで**いつでも**
相談ができる



目的地の位置や攻略のヒント、
ギャグ、豆知識など



相談内容はきめ細かく用意、
状況に応じた台詞に
(フラグ数は約**450**！)

状態異常「泥酔」の説明



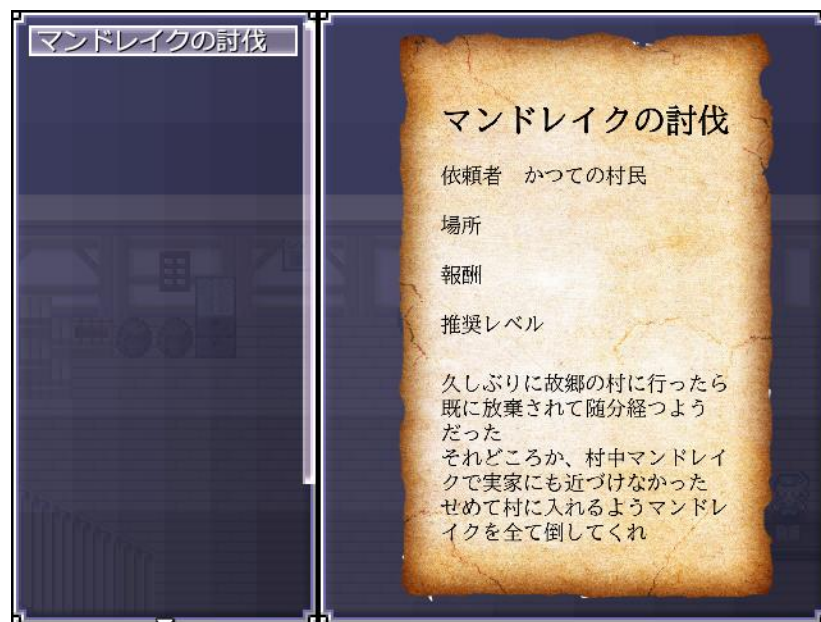
女冒険者
「まあ、攻撃されれば
酔いが醒めることもあるんだがな」

女冒険者
「この子、この前酔ったまま
魔物と戦って攻撃を外しまくって
いたのだけど」

女冒険者
「攻撃の威力自体は普段より
高かったのよね
不思議だわ」

ゲームの仕様はなるべく**キャラが説明**
世界観を大事に！

リアリティの工夫



無味乾燥な文字だけでなく、
紙っぽくこだわる



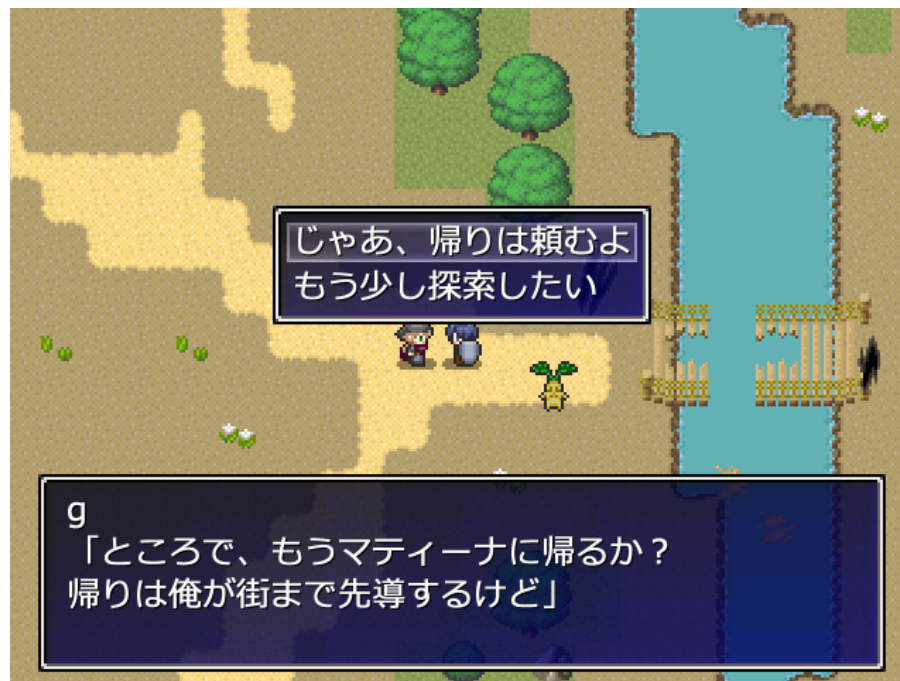
ほんの数秒だが、
宿泊時には**着替え**の演出

細かい演出



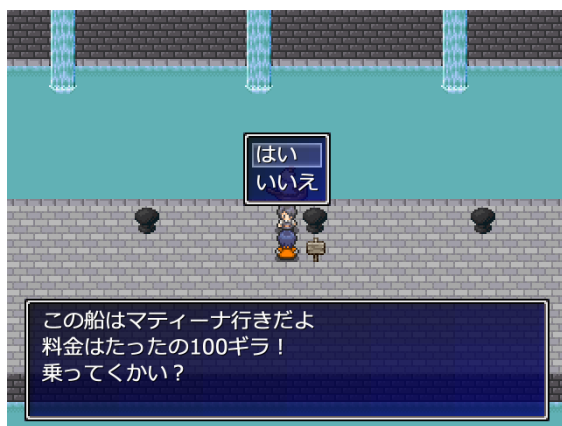
1秒にも満たない演出だが、扉のグラを複数用意し、
リアリティを

一部の行動を任意でスキップ



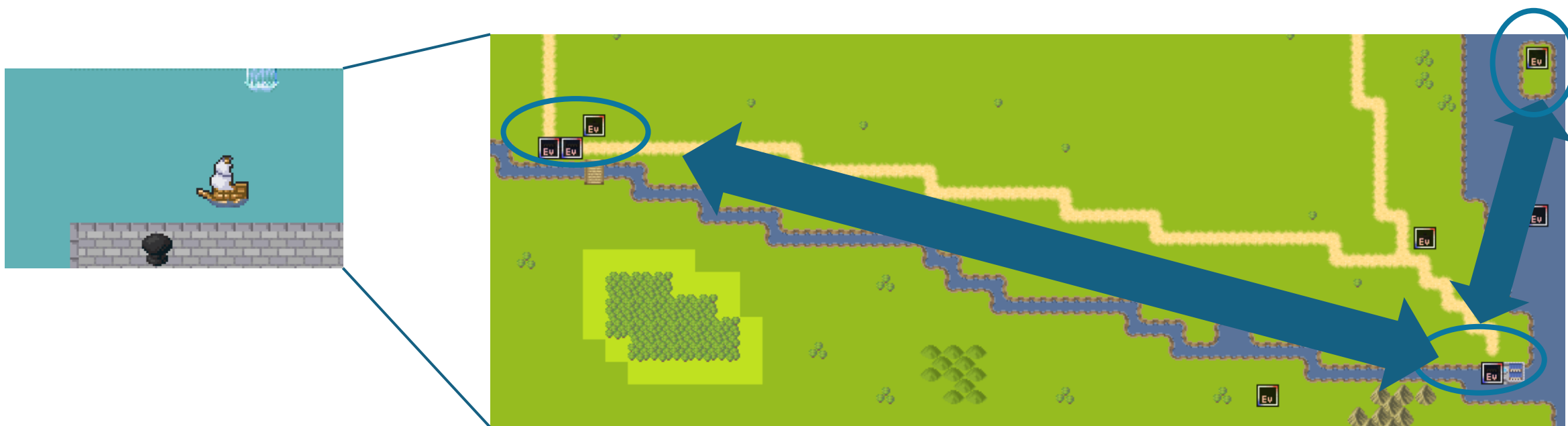
繰り返しになるクエストの行き帰りは**スキップ可能**
すぐに町に戻ることができる

世界観に合った移動手段①



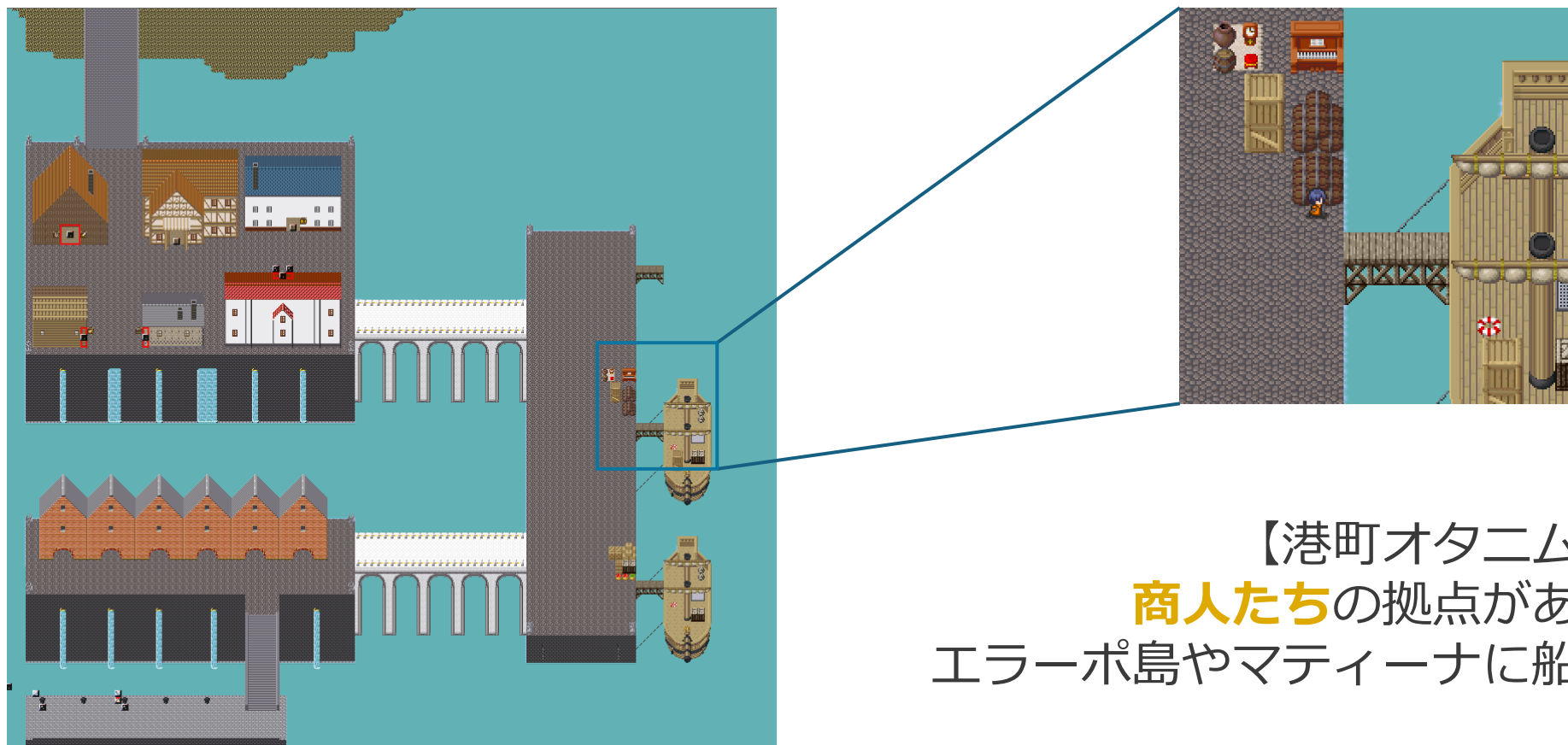
海路で繋がる町は**船で**、陸路で繋がる町は**馬車で**スキップ

世界観に合った移動手段②



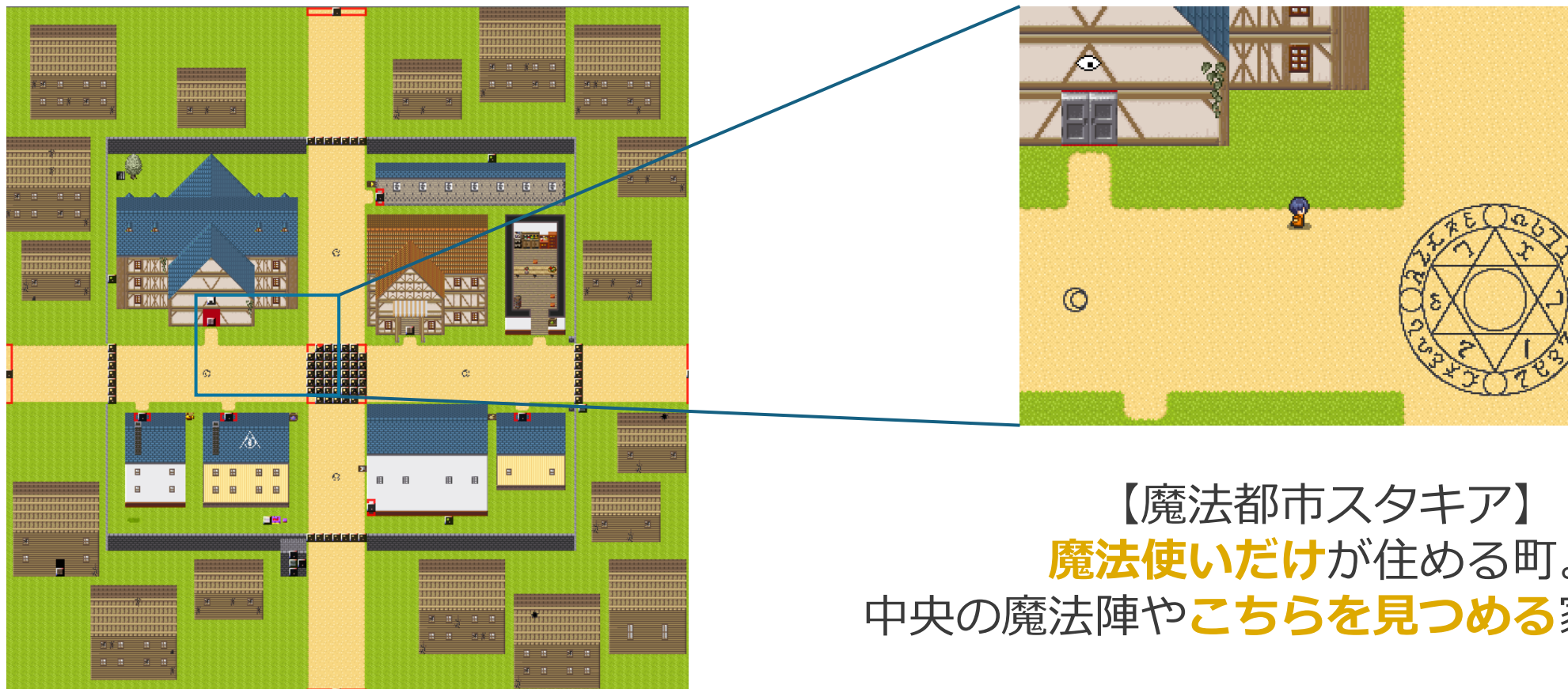
川や海で繋がる町は**船**で移動可能

マップ紹介①



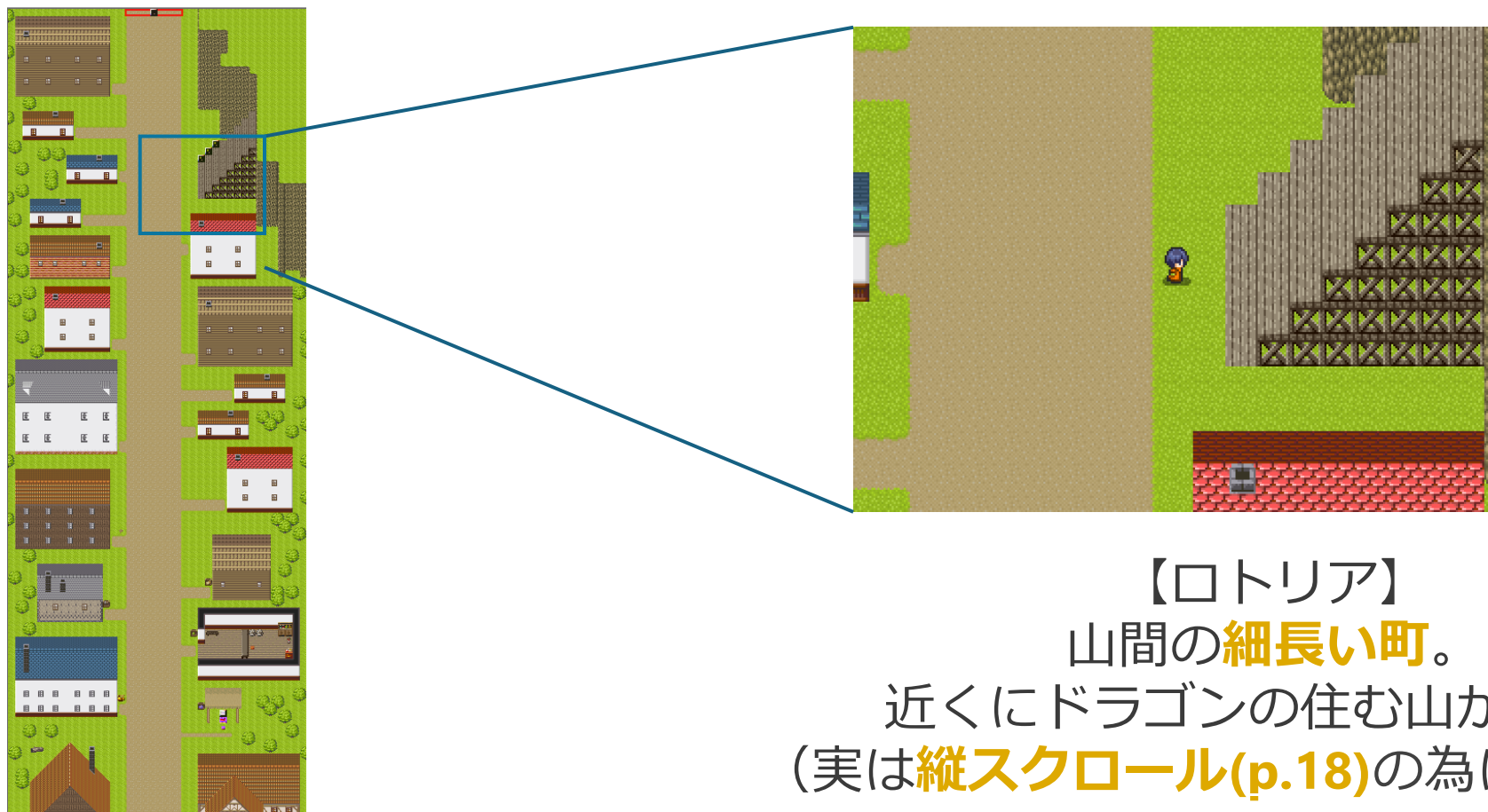
【港町オタニム】
商人たちの拠点がある町。
エラーポ島やマティーナに船で移動できる。

マップ紹介②



【魔法都市スタキア】
魔法使いだけが住める町。
中央の魔法陣や**こちらを見つめる**家が特徴。

マップ紹介③



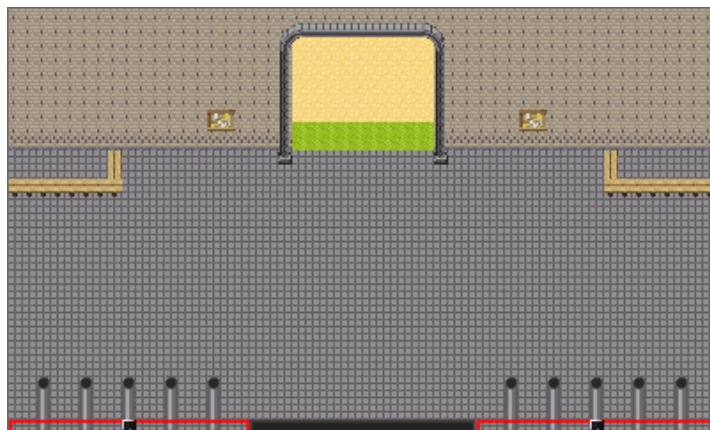
【ロトリア】

山間の**細長い町**。

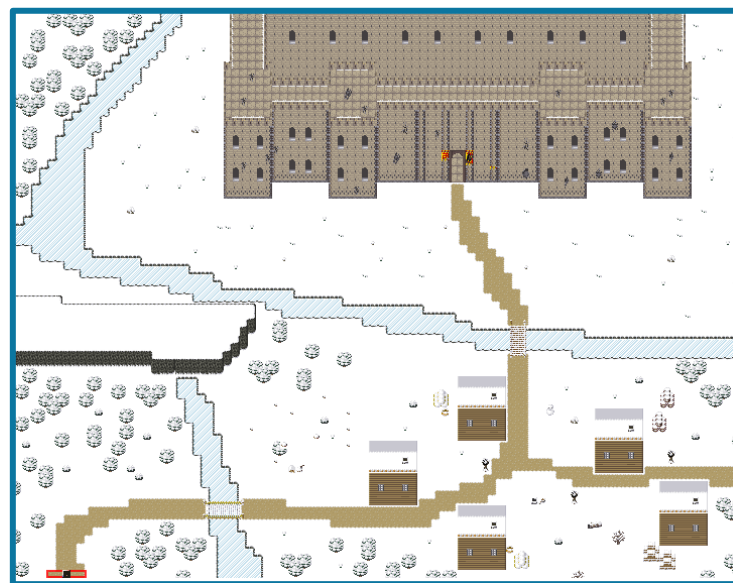
近くにドラゴンの住む山がある。

(実は**縦スクロール(p.18)**の為にこの構造)

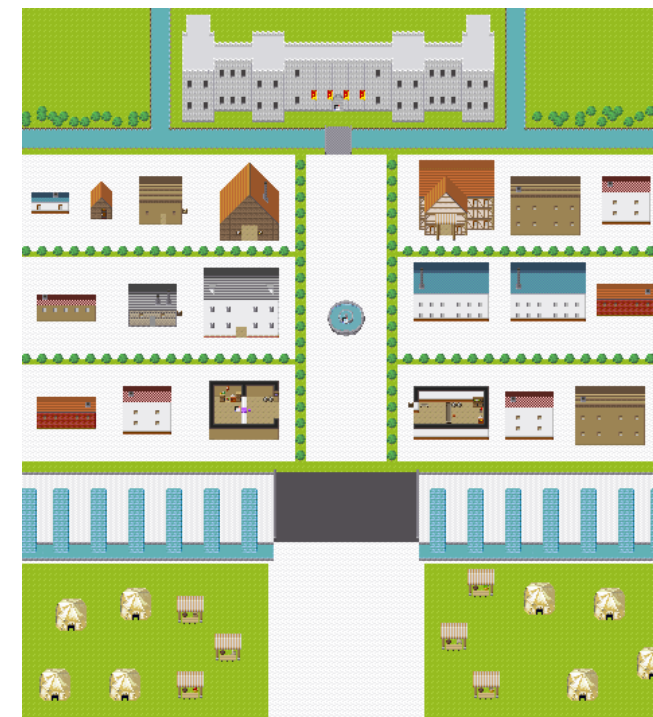
マップ紹介④



【闘技場 受付】



【放棄された城塞】



【王都ノリア】

世界観・ストーリー

p.2

ゲームシステム

p.8

イチオシ！ 様々なゲーム性

p.17

快適&奥深い！ ゲーム体験

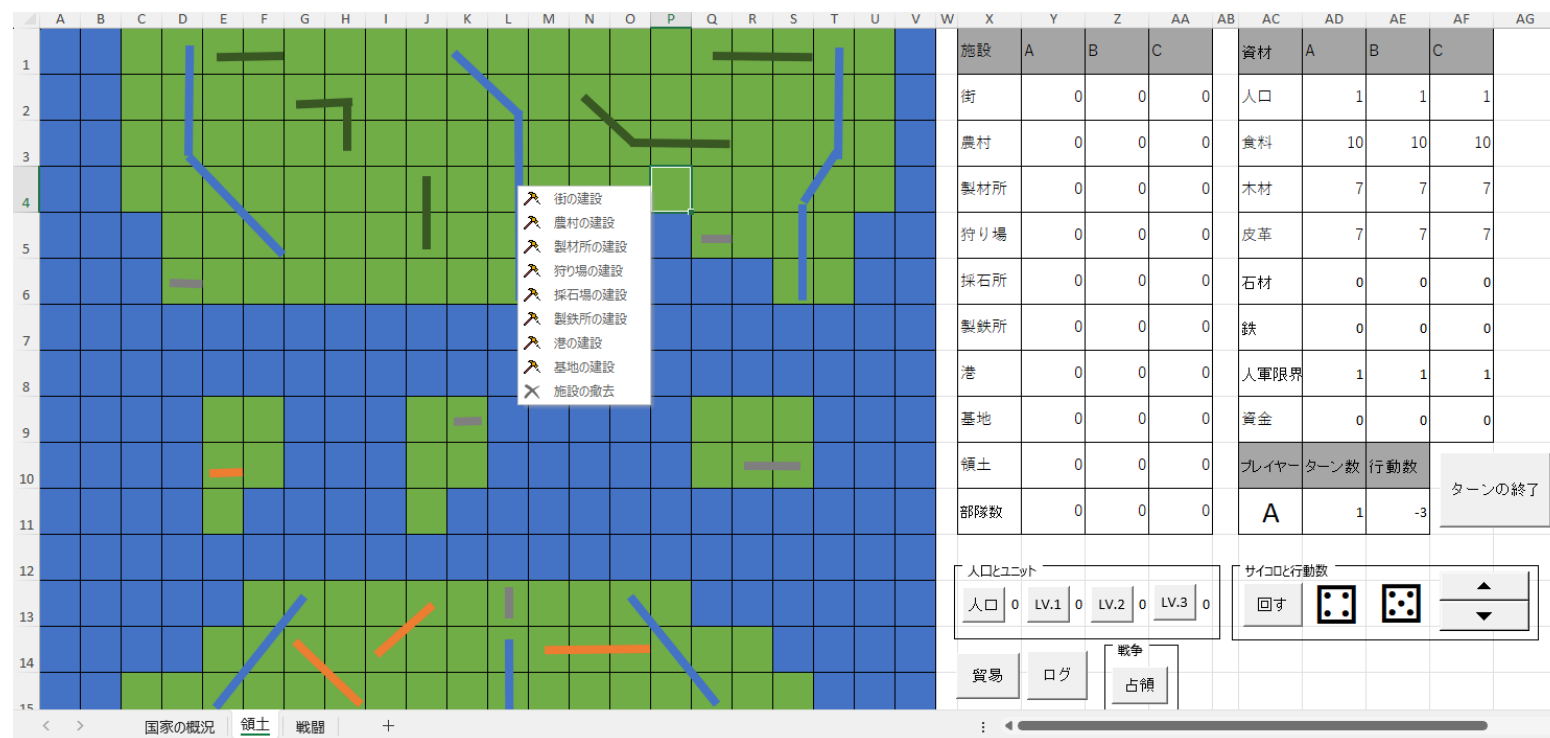
p.28

制作の経緯やツール、評価

p.41

制作の経緯

- 1 2022年夏に友人とExcelマクロを用いた
ボードゲームを制作
- 2 **ストーリーやイベントのある**本格的なゲームも制作したいと思ったことがきっかけ



過去に制作したボードゲーム

メインツール



<https://www.silversecond.com/WolfRPGEEditor/>
(公式サイト)

使用理由・メリット

- 1 C++やPythonで一から本格的なゲーム制作は労力がかかる
- 2 UnityなどもRPG向けではない
- 3 かなり高度な開発が可能で
歴史も長く(2008から)
解説資料・サイトも豊富
- 4 無料ツールで、
制作物の配布などが自由

サブツール



Word

シナリオ制作



Excel

作業進捗管理
フラグ管理
データ管理



PowerPoint

画像編集



**CLIP STUDIO
PAINT**

イラスト制作
イラスト編集

コード紹介①

- ▷C++などで一から作るよりは遥かに手間がかからない。
それでも**プログラミングは大変**
- ▷左はオープニングシーン
基本的にはキャラを動かし、台詞を言わせるだけのシーンだが、**裏では長大なコード**を要する

```

■イベントの挿入: コモン5: [ ◦メンバーの増減 ] / 0:仲間に加える / 0:主人公
■動作指定: 主人公 / 上向 / 向き固定ON
■場所移動: 主人公 ▲マップID1[「スタート」後マップ] X:34 Y:9 [精密] [トランジション + 暗転有り]
■回数付きループ [ 1 ]回
|■マップエフェクト: [ズーム] 中心X 120 / 中心Y 90 / 200% (6)フレーム
|■
◇ループここまで◇◇
■キー入力禁止/許可: 決定 キャンセル [ 移動時× キー入力× ]
■文章: %s[1] %n すやすや ..... % % % % % % % % ^
■文章: %s[1] %n すやすや ..... % % % % % % % % ^
■文章: %s[1] %n すやすや ..... % % % % % % % % ^
■文章: %s[1] %n す ..... % % % % % % % ^
■文章: %s[1] %n ..... % % % % % % ^
■文章: %s[1] %n ..... % % % % % % ^
■文章: %s[1] %n ..... % % % % % % ^
■文章: %s[1] %n [32]「はっ!」 % % % % % % % % ^
■動作指定: [ウェイト] 主人公 / 向き固定OFF / 下 / 左 / 左 / 下向
■キー入力禁止/許可: 決定 キャンセル [ 移動時○ キー入力○ ]
■文章: %s[1] %n「今日はギルドに名簿を登録する日だった!」 %n<C><決定キーでメッセージ進行>
■文章: %s[1] %n「%c[2]あいつ %c[0]を待たせてないといひけど.....」
■文章: %s[1] %n「とにかくお母さんに一言言って %n早く待ち合わせ場所に向かわないと!」
■マップエフェクト: [ズーム] 中心X 120 / 中心Y 90 / 100% (6)フレーム
■場所移動: 主人公 ▲マップID1[「スタート」後マップ] X:12 Y:5 [トランジション + 暗転有り]
■動作指定: [ウェイト] 主人公 / 上向
■ウェイト: 100 フレーム
■動作指定: [ウェイト] 主人公 / 下向 / 下
■条件分岐(変数): 【1】 V71[主人公_性別] が 0と同じ 【2】 V71[主人公_性別] が 1と同じ
-◇分岐: 【1】 [ V71[主人公_性別] が 0と同じ ]の場合
|■イベントの挿入: コモン6: [ ◦主人公情報の変更 ] / 0:主人公 / 2:歩行グラフィック画像 / %sdb[8:1:0]
|■
-◇分岐: 【2】 [ V71[主人公_性別] が 1と同じ ]の場合
|■イベントの挿入: コモン6: [ ◦主人公情報の変更 ] / 0:主人公 / 2:歩行グラフィック画像 / %sdb[8:1:1]
|■
◇分岐終了◇
■動作指定: このイベント / 下向 / 右に1つ回転 / ウェイト 2フレーム / 右に1つ回転
■イベントの挿入: コモン215: [ [北]メッセージ表示コモン ] / 0 / 0:表示して決定待 / 160 / 120 / 1階に向かいましょう
■イベントの挿入: コモン215: [ [北]メッセージ表示コモン ] / 0 / 3:全消去 / 160 / 120 / 1階に向かいましょう
■文章: 十字キーまたはWASD: 移動 %n ZキーまたはEnter: スペースキー: 決定 %n XまたはEsc: キャンセル
■変数操作: V64[開始イベント] = 1 + 0
■変数操作: V70[歩数] = 1 + 0
    
```

コード紹介②

- ▷ 左は**クエストボード**を表示させるコード
- ▷ イベント・演出のコードよりもこのような**基礎的なシステム**のプログラミングの方が**より困難**
- ▷ **座標系**の理解や**オブジェクト指向**の設計が必要になる為、プログラミングの練習になった

```

■可変DB書込:[タイ° 万能ウィンドウ一時DB(19) を初期化]
▼ 表示位置など
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ表示X: 0 ] (18: 9: 数値) = 0
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ表示Y: 0 ] (18: 10: 数値) = 0
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ横サイズ: 0 ] (18: 11: 数値) = 120
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ縦サイズ: 0 ] (18: 12: 数値) = 240
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ横項目数: 0 ] (18: 13: 数値) = 1
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ縦項目数: 0 ] (18: 14: 数値) = 15
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ縦スクロール位置: 0 ] (18: 15: 数値) = 0
■可変DB書込:DB[ 基本システム用変数: [一時]万能ウィンドウ最終カーソル位置: 数値 ] (18: 61: 0) = 0
▼ 右側の補足情報の表示
■DB読込(ユーザ): CSelf5[ウィンドウ画像] = ユーザDB[ 画像/音声設定: 0: ウィンドウ画像ファイル ] (16: 画像/音声基本設定)
■ウィッチャ表示: 16000 [左上]ウィッチャ(CSelf5[ウィンドウ画像]) サイズ[200,240] X:120 Y:0 / 0(0)フレーム / パターン 1 / 透 255 /
▼ 万能ウィンドウの設定
■変数操作: CSelf12[項目ID] = 0 + 0
■変数操作: CSelf30[ウィンドウのID] = 0 + 0
■DB読込(可変): CSelf10[総項目数] = 可変DB[ タイ°クエストボードのクエスト(21)のデータ数 ]
■可変DB書込:DB[ 基本システム用変数: 万能ウィンドウ項目個数: 数値 ] (18: 17: 0) = CSelf10[総項目数]
■回数付きループ [ CSelf10[総項目数] ]回
| ■DB読込(可変): CSelf14[追加フラグ] = 可変DB[ クエストボードのクエスト: CSelf12[項目ID]: 追加フラグ ] (21: -: 2)
| ■DB読込(可変): CSelf15[消去フラグ] = 可変DB[ クエストボードのクエスト: CSelf12[項目ID]: 消去フラグ ] (21: -: 3)
| ■条件分岐(変数): 【1】 CSelf14[追加フラグ] が 1と同じ
| | -◇分岐: 【1】 [ CSelf14[追加フラグ] が 1と同じ ]の場合↓
| | ■条件分岐(変数): 【1】 CSelf15[消去フラグ] が 0と同じ
| | | -◇分岐: 【1】 [ CSelf15[消去フラグ] が 0と同じ ]の場合↓
| | | ■DB読込(可変): CSelf6[クエスト名] = 可変DB[ タイ°クエストボードのクエスト(21) データCSelf12[項目ID](-)のデータ名 ]
| | | ■DB読込(可変): CSelf13[返り値] = 可変DB[ クエストボードのクエスト: CSelf12[項目ID]: 1 ] (21: -: 返り値)
| | | ■可変DB書込:DB[ 万能ウィンドウ一時DB: CSelf30[ウィンドウのID]: 0 ] (19: -: 項目文) = CSelf6[クエスト名]
| | | ■可変DB書込:DB[ 万能ウィンドウ一時DB: CSelf30[ウィンドウのID]: 1 ] (19: -: 項目コード) = CSelf13[返り値]
| | | ■可変DB書込:DB[ 万能ウィンドウ一時DB: CSelf30[ウィンドウのID]: 2 ] (19: -: 選択可能状態[1=可]) = 1
| | | ■変数操作: CSelf30[ウィンドウのID] += 1 + 0
| | |
| | | ◇分岐終了◇
| |
| | ◇分岐終了◇
| ■変数操作: CSelf12[項目ID] += 1 + 0
|
| ◇ループここまで◇◇
▼ 万能ウィンドウの表示
■イベントの挿入: コモン64: [ X[共]万能ウィンドウ描画処理 ] / 0:表示 / -1 / 0:表示しない / 0:なし
▼ クエスト画像を表示
■ループ開始
    
```