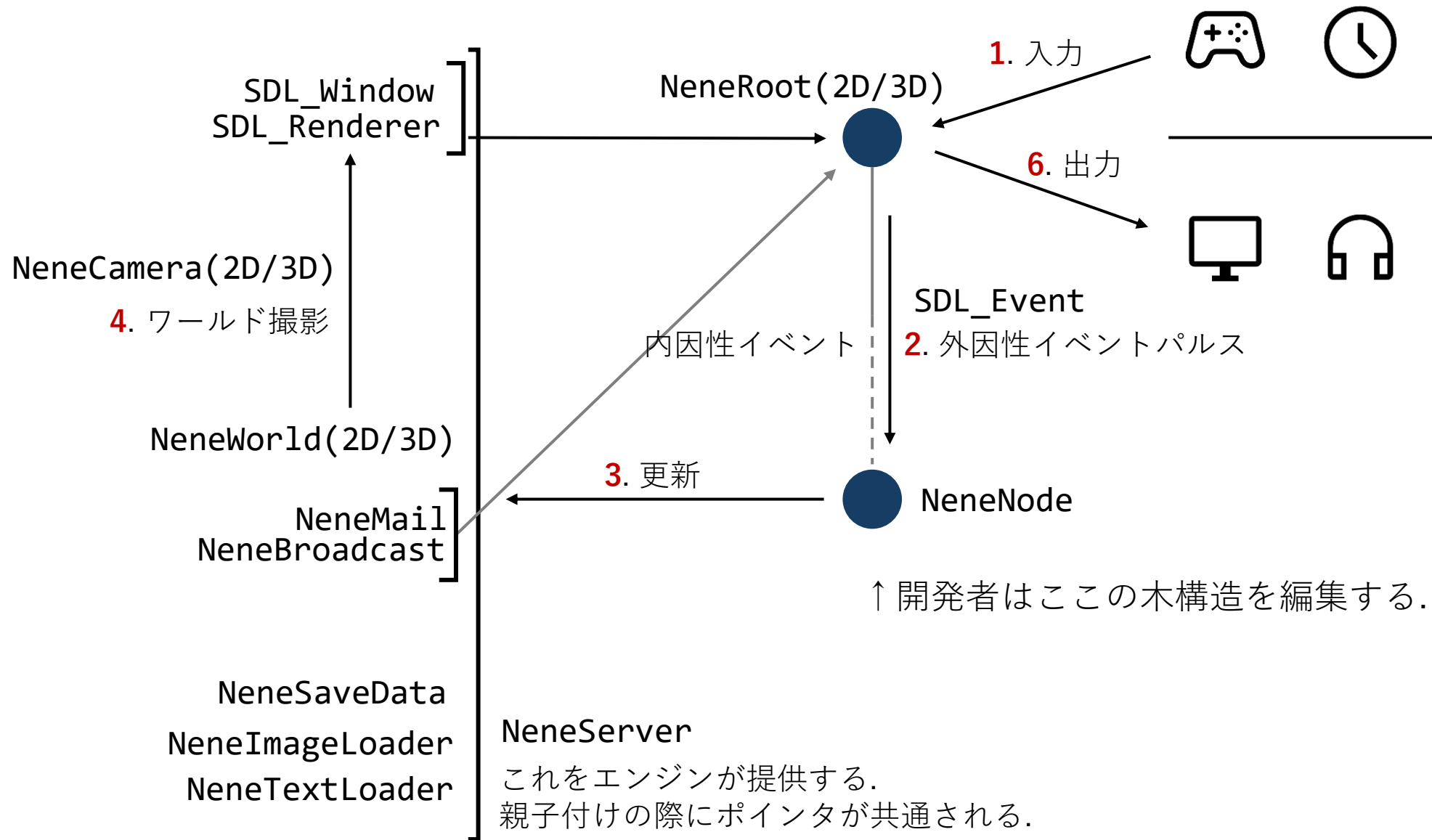




Tree-and-Pulse; GameDevEngine

NeneEngine

ねねエンジンの仕組み



NeneNode

親	生ポインタ
子	unique_ptrの配列
名前	char*
ダーティフラグ	bool
save()	自身の状態を復元するのに十分なstringを生成
load(std::string)	セーブデータを基に子ノードを生成
relay_mail()	

内部イベントシステム: NeneMail

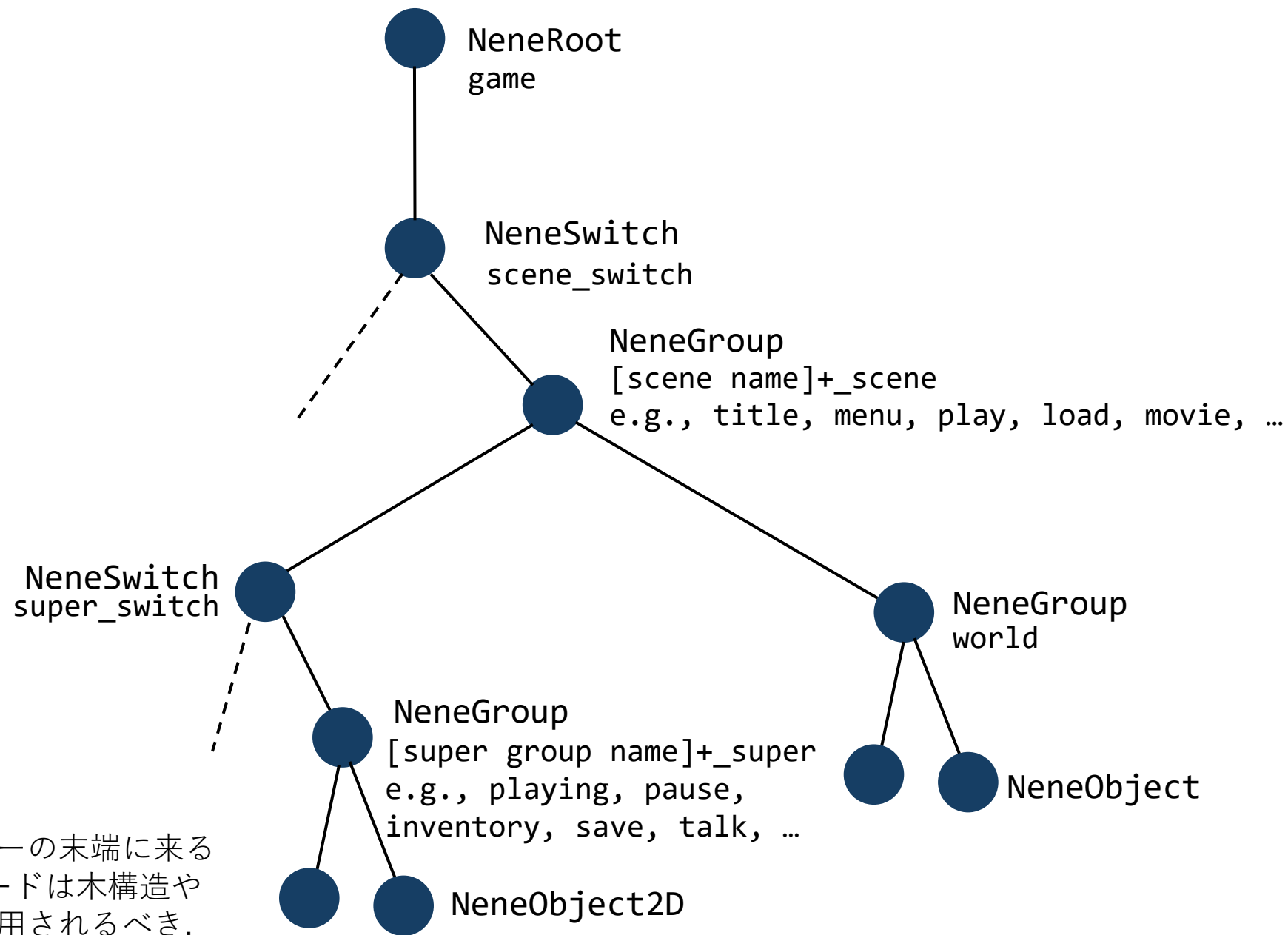
NeneMail

宛先	ノードの名前(可Null)
送り主	ノードの名前
件名	std::string (e.g., “switch scene”)
本文	std::string (e.g., “1”)

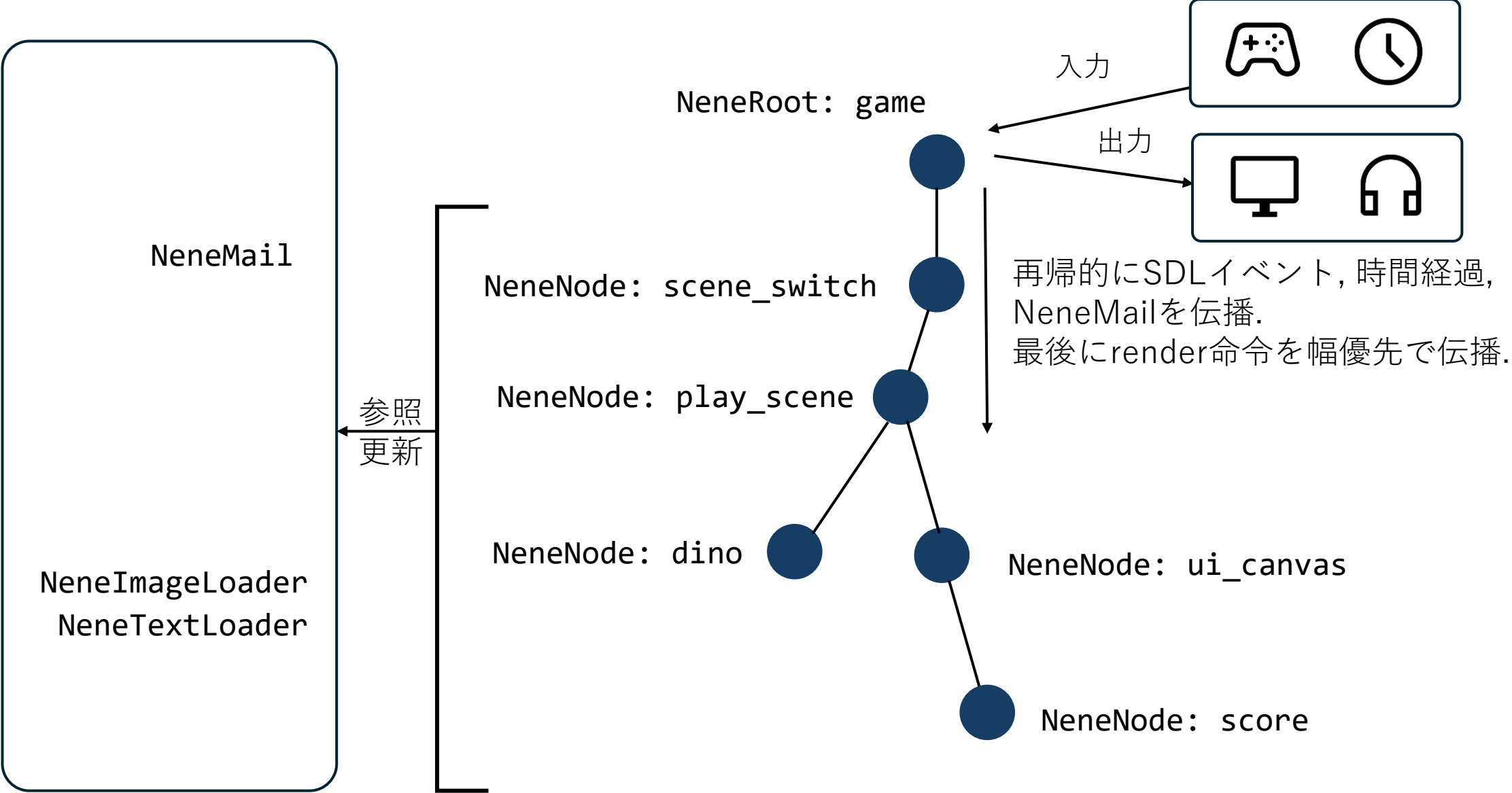
NeneMailServer

mail_queue	NeneMailの配列
push()	配列にメールを追加
pull()	配列からメールを取得

ねねツリーの例



NeneEngine2D(ver. 1.0.0)



ファイル構成, クラス構成

include/NeneEngine/

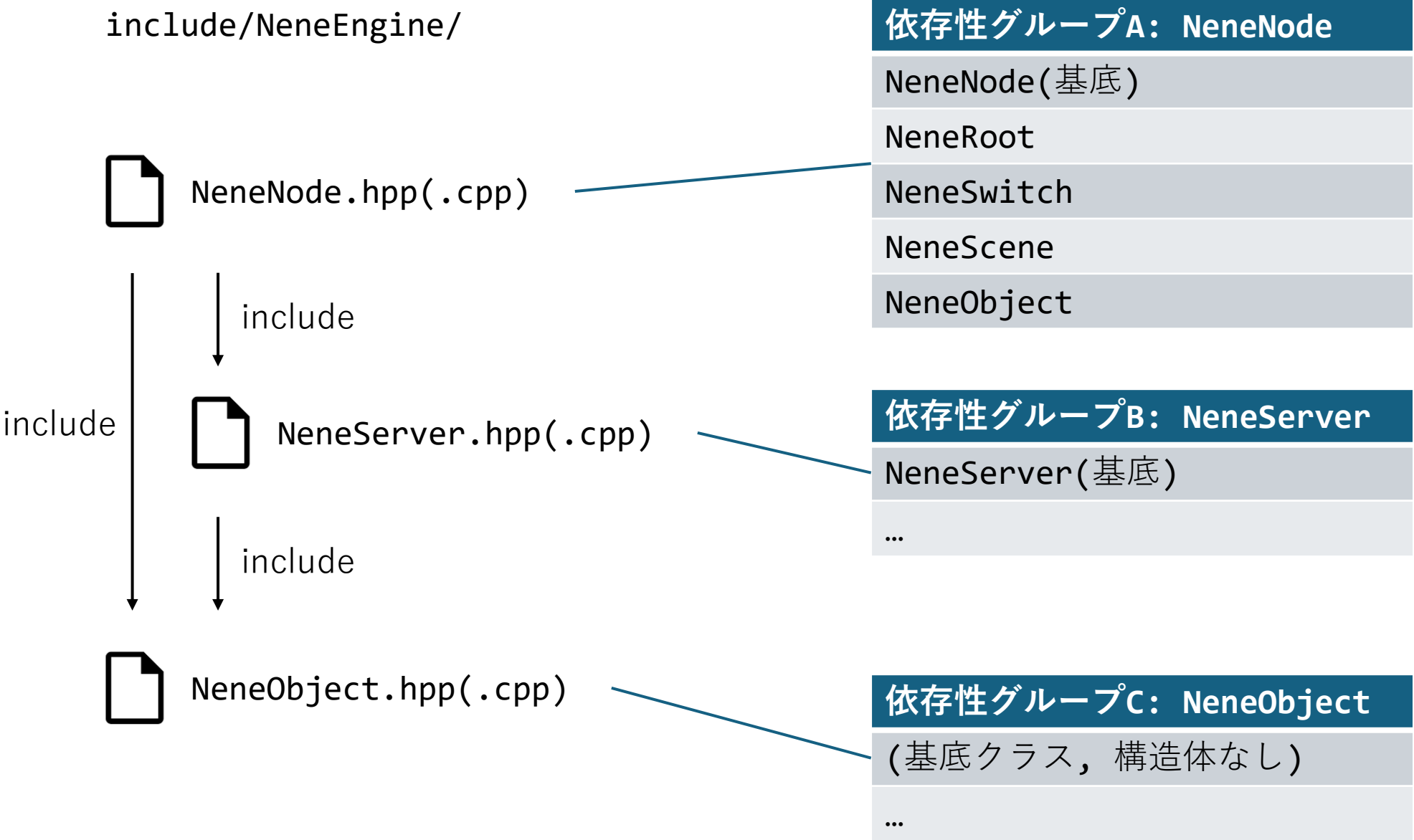
NeneNode.cpp(.hpp)	
NeneNode	基底クラス
NeneRoot	メインループ. tick制御, イベント伝播



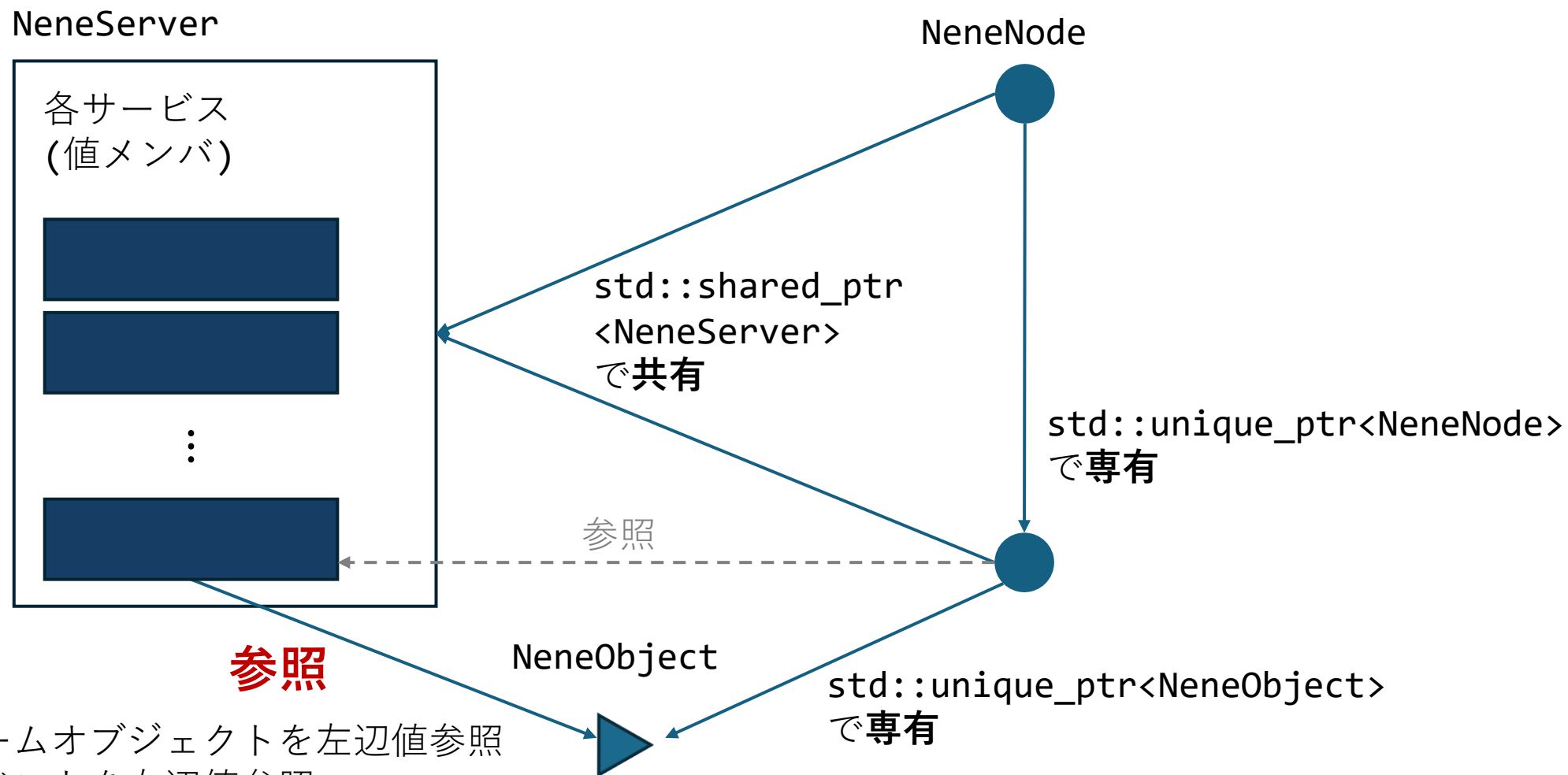
NeneServer.cpp(.hpp)	
NeneServer	基底クラス
NeneMail	特定のノードにイベントを命令(シーン切り替えとか)
NeneBroadcast	全てのノードにイベントを通知(UIボタンが押されたとか)
NeneImageLoader	画像ローダ
NeneTextLoader	フォントローダ

SDL3
SDL_Window
SDL_Renderer
SDL_PollEvent
⋮

依存性グループ



所有権とアクセス権



ゲームワールドはゲームオブジェクトを左辺値参照
イベントキューはイベントを右辺値参照
(なんか偶然やりたいことが参照とまったく同じ概念だった)

(イベントは使い捨てなのでそもそも所有しない)

NeneCamera	camera	
NeneCompositor	compositor	
NeneViewWorld		
NeneSuperimpose		
NeneEventQueue		

shoot(), compose()はrendererを
持ってるルートノードのみが実行可能.

NeneRoot

● window
renderer

!!!	強制終了を伴う致命的なエラー
!!	
!	想定と異なる挙動を伴う警告
#	一般的なメッセージ
+	ツリーへのノードの追加
-	

ねねエンジンのWebサイト、ねねサイト

後で作る

ねねサイト	内容
ねねポータル	概要
	ダウンロード, 更新ログ
	各種サービスへのリンク
ねねチュートリアル	環境構築の手順
	ねねエンジンの仕組みについて
	ゲーム制作の手順とコツ
ねねリファレンス	仕様の詳細な解説
ねねテンプレート	基本的なゲームツリーとエントリーポイントのテンプレート
	ねねノードのテンプレート(非同期ロードを行うスイッチとか)
	アニメーション, 力学シミュレーション, その他のテンプレート
ねねタイルマップエディタ	レベルデザイン支援ツール
ねねメジャー	ウィンドウにおける座標・サイズ取得ツール
ねねツリーエディタ	ねねツリー編集ツール