Lost Adventure（Miniplay / GameDistribution）资源与技术解析报告

0. 结论速览

项目	结论
引擎	Unity 6（6000.0.49f1）WebGL，IL2CPP 编译为 WebAssembly
开发商	Ultra Games Entertainment（companyName `ultragames`，productName `LostAdventure`，v1.3）
分发链路	miniplay.com 页面 → `html5.gamedistribution.com/90dab544…/`（GD SDK 外壳）→ `…/rvvASMiM/90dab544…/index.html`（Unity 播放器）
主包体	`.data` 36.0 MB（未压缩）、`.wasm` 55.4 MB、`framework.js` 0.5 MB
额外内容	Addressables 2.4.6，8 个远程 bundle（≈56 MB），含飞机开场场景、坐骑、谜题预制体等
场景结构	单一超大主场景 `level1`（13.4 万 GameObject），5 个岛屿区域 + 飞机区，靠 激活/解锁 切换，而非多场景加载
角色	通用 Humanoid 命名骨架（Mixamo 风格，43 节点），模块化换装（身体/衣服/鞋/帽分 SkinnedMesh）
动画	Mecanim Generic 动画（无 Avatar 资产），单层状态机，`State`(int) + `Mul`(float) 参数驱动
寻路	Unity NavMesh（NavMeshSurface 运行时烘焙）+ A* Pathfinding Project 库
核心玩法	采集 → 背包 → 加工机（转化器）→ 解锁地块 / 建筑升级 → NPC 自动工作 → 迷你谜题开路 → 救援乘客、收集飞机零件

1. 加载链与文件清单

1. `miniplay.com/embed/lost-adventure` 内嵌 iframe 指向 GameDistribution，`mp_game_id=255006`。
2. GD 外壳加载 `html5.api.gamedistribution.com/main.min.js`（SDK 1.15.2），再把真实游戏 iframe 指到 `//html5.gamedistribution.com/rvvASMiM/90dab544fe47444eacfdfb97b704d5c1/index.html`。
3. Unity 模板加载：

文件	大小	说明
`Build/Lost Adventure Game Distribution.loader.js`	26 KB	Unity loader
`….framework.js`	516 KB	Emscripten 胶水 + JS 插件（GD SDK 桥）
`….wasm`	55.4 MB	IL2CPP 原生代码
`….data`	36.0 MB	UnityWebData1.0 容器
`StreamingAssets/aa/`	—	

文件	大小	说明
		Addressables catalog（catalog.bin）+ `WebGL/*.bundle`

`.data` 解包后（UnityWebData1.0 头 + 文件表）：

文件	大小
`data.unity3d`（内含 globalgamemanagers、level0、level1、sharedassets0/1、resources.assets）	22.2 MB
`Il2CppData/Metadata/global-metadata.dat`（v31）	11.4 MB
`sharedassets1.resource`（音频/流数据）	0.8 MB
`Resources/unity default resources`	1.6 MB
`ScriptingAssemblies.json` / `RuntimeInitializeOnLoads.json` / `boot.config`	小

Addressables bundle：

Bundle	大小	内容
defaultlocalgroup_scenes_all	24.1 MB	场景 `Main Plane.unity`（飞机坠毁开场）、`Global Links.unity`（连线小游戏）
prefabgroup_assets_all	15.5 MB	32 个预制体：坐骑（鸵鸟/猪/狼/海王）、鲨鱼、豆茎、Match3、Block Colors、Block Escape、Sort Away、铜桥、矿机等
defaultlocalgroup_assets_all	5.3 MB	BGM `ForestWalk`
fontgroup / spritegroup / texturegroup	4.5 / 4.0 / 2.0 MB	TMP 字体、UI 精灵、贴图
monoscripts / unitybuiltinassets	小	脚本引用与内置 shader

2. 资源统计（主包）

类型	数量	类型	数量
GameObject	134,472	Mesh	231（+bundle 695）
MonoBehaviour	80,394	SkinnedMeshRenderer	8,416
MeshRenderer	23,403	Animator	1,341
BoxCollider	10,256	AnimationClip	85（+bundle 78）
Rigidbody	3,628	AnimatorController	17（+bundle 26）
ParticleSystem	3,459	Texture2D	396
MonoScript	3,225（游戏自写 379）	AudioClip	19（+bundle 5）
NavMeshAgent	10	NavMeshData	2

GameObject 数量巨大，原因是所有关卡区域、资源点、UI 都预摆在同一场景里；SkinnedMeshRenderer 8,416 个说明大量“可破坏物/树/植物”也做了骨骼。

3. 人物与骨骼

3.1 主角 Player

场景路径：`Main Scene/Player/T-Pose_01 (5_Shoes)`，183 个节点。

- 模块化部件（独立 SkinnedMesh）：`Body`（1,343 顶点 / 73 骨骼影响）、`Shirt`(26)、`Pent`裤子(18)、`Cap`(8)、`Belt`(10)、`Right_Shoes / Left_Shoes`(8)、`Tie`、`Face_01/02`；`Pilot_Cloth` 为飞行员服装组。
- 加速装备（挂在同一骨架下，按需激活）：`Ice_Skate`、`Rollar_Skate`、`skateboard`、`model_0:Hover_Board`、`Snow_Shoes`——对应枚举 `SpeedBoosts {IceSkate, RollerSkate, SkateBoard, HoverBoard, SnowShoe}`。
- 骨架（43 节点，Mixamo 命名）：

```
Hips
├ LeftUpLeg → LeftLeg → LeftFoot → LeftToeBase → _end
├ RightUpLeg → RightLeg → RightFoot → RightToeBase → _end
└ Spine → Spine1 → Spine2
   ├ LeftShoulder → LeftArm → LeftForeArm → LeftHand
   |    ├ LeftHandIndex1-3(_end)    └ LeftHandThumb1-3(_end)
   ├ Neck → Head → Head_end
   └ RightShoulder → RightArm → RightForeArm → RightHand
       ├ RightHandIndex1-3(_end)   └ RightHandThumb1-3(_end)
```

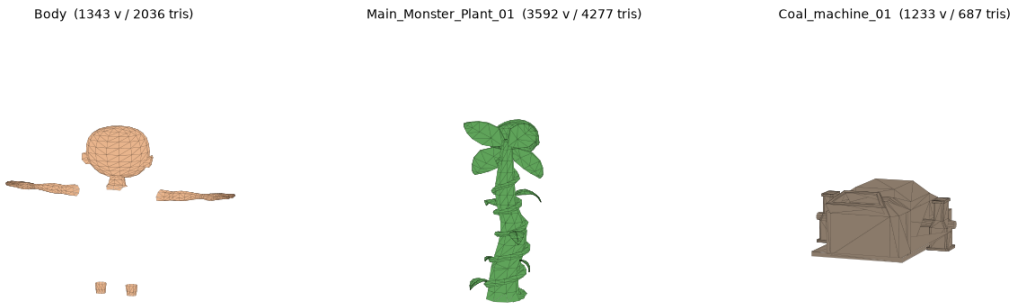
手只有食指+拇指（低模卡通风格）。工具（斧头 `Axe_01…05`、`OldAgeAxe`、`FlameThrower`）挂在手骨上。

3.2 NPC / 乘客

- 9 个 NPC（`All NPC`，每个 44 节点），共用 `T-Pose_01 (Boy & Girl)` 骨架，靠 `boyPartsIndex / girlPartsIndex` 随机组合发型、外套、帽子、胡子、耳机、斗篷等部件。
- 开场场景中乘客有多套姿态网格：`Fear_1 / Fear_02 / Sitting_Fear_01 / Pray_01 / Sleep_01 / Passenger_01-05`。
- `T-Pose_001 (Aquaman & Mermaid)`：海王/美人鱼形态（`AquaSuit`、`Mermaid_Tail`、`Trident`）。

3.3 敌人与生物

对象	网格/骨骼	控制器
食人植物 Monster Plant	`Main_Monster_Plant_01`（62 骨）	`Regrow, Attack Short, Damage, Idle, Attack Long`，参数 `Plant`
鲨鱼 Shark	`Shark`（1,946 v / 33 骨）	`Idle, Swim, Attack, Right Turn, Left Turn`
坐骑：鸵鸟 / 猪 / 狼	`Ostrich`(29 骨)、`Pig_Mesh`(38 骨)、狼	独立 Idle/Run，并有骑乘版玩家控制器
陆地敌人（代码）	—	`EnemyType {Wolf, Bear, Yeti}`，`AttackWolf / AttackBear / AttackYeti`



主角身体、食人植物、煤炭机网格（从游戏资源导出后渲染）

4. 动作与动画系统

4.1 玩家状态机（Player / HoverBoard / IceSlate / RollerSkate / SkateBoard / Player Wolf / Pig / Ostrich / Aquaman 共用结构）

单层、15 个状态，参数 `State`（int，状态编号）+ `Mul`（float，播放速度倍率）：

#	状态	时长	动画事件
0	Idle	7.0 s 循环	—
1	Walk / Run	0.67 s 循环	`WalkAudio` ×2（脚步声）
2	JumpInWater	1.25 s	—
3	Swim	0.92 s 循环	—
4	SwimIdel	2.17 s 循环	—
5	ClimbFromWater	3.58 s	—
6	WalkCut（边走边砍）	2.33 s 循环	`BoatHit` ×2（命中判定）
7–8	Cuting1 / Cuting2	0.42 / 2.5 s	`BoatHit`
9	Die	3.33 s	—
10	Construct	0.62 s 循环	—
11–12	Stack Idle / Stack Run（扛物）	1.67 / 0.83 s	`WalkAudio`
13	Respwan	0.83 s	—

要点：

- 伤害结算由动画事件驱动：砍伐/挖矿命中通过 `BoatHit` 事件回调，而非按时间轮询；`Mul` 用来随装备等级加快砍伐速度。
- NPC 控制器：`Idle, Walk, WalkCut, Cuting1, Cuting2, Happy, Freezing, Hanging`（`Freezing` 对应寒冷区域）。
- 机器动画带事件：`Coal Animation` / `Wood Cutting Machine` → `FillBar`、`TakeOutput`；`Fish Cook` → `OutPut`。即产出节奏 = 动画循环时长，参数 `Speed` 控制产速。
- 食人植物攻击动画事件 `DamageToPlayer`、`DamageDone`、`CheckIdle`。
- UI/提示类用 Legacy Animation：`TapHand`、`ZoomInOutTuto`、`Match3 Hand`、`Rotation Puzzle Hand`、`ScaleUpPopUp`、`Shake` 等。
- 补间大量使用 DOTween / DOTweenPro 与 Feel（MoreMountains Feedbacks） 做跳跃收集、震屏、数字飘字。

5. 场景与地图引擎

5.1 结构

```
Main Scene
 ├ Player / Cam & light / UIManager(1,559 节点) / Canvas
 ├ WaterEnvironment (Mobile Optimized Water) / Fish (鱼群 Flock)
 ├ All NPC (9) / All Enemy SHARK (8 条鲨鱼)
 ├ Area All (131,512 节点)
 |   ├ Area (1)~(5), 每个区域四个子组：
 |   |   Unlocked AREA / Locked AREA / Puzzle AREA / NPC AREA
 |   ├ Airplane Area (坠机残骸 PlaneCrash_Parts1~37、螺旋桨)
 |   └ Plane Assemble / Plane Scene Addressable
 └ Tutorial Things
```

- Locked AREA 内按 `Step N` 分块（Area 4 有 Step 11~18）。每个 Step 都是一个标准模板：`Base` 地块、`BG`、`Trees`、`Stone`、`Coal`、`Iron`、`Bronze`、`Purple Crystal`、`Cabbage`、`Strawberry`、`Grass`、`Storage`、`Converter`、`Enemy Monster plant`、`Mini Map Area`、`Unlock`（解锁牌）、`CenterPointCanvas`。
- 这是手摆的岛块拼接地图（地块 `Base 01 GO`、`Step _19`，道路 `Road Straight / Road Turn / Road_T / Road_End`，桥 `Wood Bridge / Bronze Bridge`），不是 Tilemap/地形系统。
- 区域谜题分布：

区域	谜题
Area 1	RotationPuzzle
Area 2	Maze Puzzle、Block Color、Sort Away
Area 3	BridgePuzzle、Maze Puzzle 2、Block Escape
Area 4	BridgePuzzle 2、Mirror Puzzle（激光反射）、Block Color 2、Match3
Area 5	Bronze Bridge、RotationPuzzle 2、Block Escape 2、Match3

5.2 运行机制

- 解锁 = 激活：`UnlockManager` 持有 `unlockObjects / Collectables / ExtraNavObjs / BridgeObj / lastUnlockedGround`；解锁后启用对应 Step，并调用 `NavigationBaker` → `NavMeshSurface.BuildNavMesh()` 运行时重烘焙导航网格（`GameManager.NavGenerator` 也有同类调用）。
- Addressables 按需加载：`AddressableMain` 缓存/释放资源，`UnlockThing.sceneAddressable / coinChestAddressable` 在解锁时才拉取坐骑、谜题、宝箱预制体，降低首包体积。
- 相机：Cinemachine + 自写 `MtCameraSys`（中心点平滑跟随）、`ZoomInOutManager`（双指缩放），`NPCManager` 有 7 个 NPC 特写机位。
- 小地图：`MiniMapManager` 独立正交相机 + RenderTexture，支持缩放/拖拽，`xMinMax / zMinMax` 限定边界。
- 渲染：Toony Colors Pro 卡通着色、Mobile Optimized Water、Kino Bloom、AllIn1Vfx / Epic Toon FX / CFX 粒子；草地 `GrassGPUInstancing` + `GrassCut`（按半径遮罩割草）。
- 绳索物理：Obi Rope（`RopeManager`、`RopeSweepCut`、`WrapRopeGameController`）。

6. 核心操作

- 移动：拖动摇杆式控制（`DragController` / `TouchDrag` → `PlayerController.Movement`），角色运动用 Easy Character Movement（`CharacterMovement`，含 `acceleration / deceleration / groundFriction / airFriction / jumpImpulse / gravity`），水面检测 `WaterlayerMask` 切换游泳。

- 自动交互：靠近目标自动执行——`UpdateTarget` 在 `Range` 内找最近可采集物（`nearestObj / shortestDistance`），进入砍/挖状态；靠近机器/仓库/解锁牌的 Trigger 自动投放资源（`OnTriggerStay → TakeFromPlayer`）。
- 点击：`TouchRaycast`、`BoxColliderClicked`、`NPCmain.OnMouseDown`（给 NPC 派任务）、谜题内拖拽 `ObjectDragger`。
- 按键：Miniplay 页面标注的控制为 “MOVE / SELECT”，即方向移动 + 点击选择。

7. 核心玩法与算法

7.1 资源与背包

- 资源枚举 `CollectableTypes`：Wood, Coal, Bronze, BlueCrystal, Cabbage, Stone, Strawberry, Iron, Fish, CookedFish, WoodPlank, CoalTar, IronOre, Plants（14 种）。
- `InventoryManager`：`storageCapacity / curStorage` 容量上限（满包有 `BagMaxAnim` 抖动提示），`PlanePartsCount / TotalPlanePartsCount` 统计飞机零件进度。
- 采集物 `TakeSingle`：被拾取后 `Deactive`，按 `GameManager.spawnPer / respawnTime` 定时 `Respawn`；部分为飞机零件 `isPlanePart`。

7.2 加工链（ConvertorMain）

输入1/输入2 → 产出（如 原木→木板、煤→煤焦油、铁矿→铁、鱼→熟鱼）：

```
OnTriggerStay(Player) → TakeFromPlayer(扣背包, inCount += giveAmount)
Update: inCount ≥ takeAmount 且 outCount < maxAmount
     → 播放机器动画(AnimSpeed), 动画事件 FillBar 更新进度条
     → 事件 TakeOutput: inCount -= takeAmount; outCount += 1
Player 再进入 → StartCollection → 产出飞入背包
CheckForUpgrade → UpgradeOutCountSpeed（提高 AnimSpeed / 容量）
```

7.3 解锁与经济

- 解锁牌 `UnlockThing`：玩家站在牌上时按 `unlockTime` 节奏逐个扣金币（`valueCoinEach`，`_timeResc` 计时），扣满 `UnlockCoins` 后 `UnlockThings()` 激活下一 Step，可选 `ChangeScene`（加载 Addressable 场景）或 `CallJones`（剧情 NPC）。
- 建筑 `BuildingMain`（帐篷 1~3 级，决定 `maxNpc`）、仓库 `WareHouseMain`（NPC 存入、玩家一键取出、升级扩容）、农田 `FarmManager`、商店 `SellThings`（卖资源换金币）。
- 价格曲线：`ClickerUpgradeSystem` / `EquipmentSystem` 都有 `SetPriceExpo`、`FindPercent`，配合枚举 `IncrementType {Simple, Double, Percent, Exponential}`——即每级价格可选线性、翻倍、百分比递增或指数递增；数值存在 `UpgradeParameter.AllLevelUpgradeSystem`。

7.4 装备

`Equipments {FlamThrower, Axe, Bag, Coat, Boot, Pet}`：斧（砍伐伤害/速度）、包（容量）、外套（抗寒，对应 `coldResist`、NPC `Freezing`）、靴（速度）、喷火器（融冰/烧障碍，配合 `iceRegrowTime`）、宠物。`EquipmentSystem.UpgradeEquipment` 消耗资源升级，UI 显示两项属性的当前/升级值。

7.5 NPC 自动化（NPCmain）

1. 被救出/解锁后进入工作循环：`FindCurrentTraget` 在可采集类型 `collectableTypes` 中挑最近目标 → NavMeshAgent 走过去 → 砍伐 → 背包满（`maxCollected`）→ 送往 `StoresTraget`（仓库）。
2. 饥饿机制：`HungerTime` 计时后 `isHungry`，弹出饥饿图标，玩家需给熟鱼等食物（`NPCHungryTime`、`NPCDinnerPlcae`），否则停工。

3. 无目标时 `notargetTime` 后待命/睡觉（`isSleeping`），可由玩家点击 `OnMouseDown` 指派任务（`PassengersManager` 选择面板）。

7.6 战斗

- 玩家：`curHealth / maxHealth`，`TakeDamage`、`PlayerDie` → 掉落 `Die Crate`（背包资源装箱，可回收）→ `Respwan`。
- 敌人 `EnemyMain`：待机于出生点，玩家进入 `range` 后 `GotoPlayer / RushToPlayer / JumpToPlayer`，进入 `attackRange` 按 `attackTime` 攻击；离开范围 `GoToMainPos` 回巢；死亡掉落 `rewards`，`respawnEnmyTime` 后复活。
- 食人植物 `MonsterPlant`：`ShortAttackRange / LongAttackRange` 两段攻击，被砍 `TotalHits` 次后由大→中→小形态缩减，最终解锁 `UnlockingPoint`，之后 `Regrow`。

7.7 谜题算法概要

谜题	机制
RotationPuzzle	点击旋转部件，`CheckRotations` 判定所有角度对齐
Match3	`Match3Controller` + `SwipeConttoler` + `WoodSpwaner`，三消获得木头，完成后自动走过桥
Block Color / Block Escape	华容道式滑块（`TileType` 定义 1~4 格形状，`BlockMovementHandlerBE`），把色块移入目标区
Sort Away	分类排序
Mirror Puzzle	`LaserReflector` 激光反射：旋转镜子把激光导向目标点燃火堆
Wire / Global Links	连线通电（`CheckConnection`、`ChargingCable`、`Battery`）
Rope / Maze	Obi 绳索剪切、迷宫推球

7.8 其它系统

- 剧情：`GameManager.dialogueList` 对话面板、`NPCChat` 气泡；开场 `Main Plane.unity` 飞机坠毁（`Plane Camera Animation`）。
- 最终目标：`PlaneAssemble` 检查所有 NPC 获救 + 零件收集完毕 → 修好飞机。
- 激励视频：`GameDistributionSDKManager.ShowRewardedAd` → 速度奖励（`SpeedRewards`，坐骑 `SpeedType {Pig, Osctrich, Wolf, Aquaman}`）、资源翻倍、一键卖出（`SellAllAdsInventory`）。
- 存档：各建筑/仓库/装备类各自实现 `LoadData / StoreData`（推断为本地 PlayerPrefs 存储）。

8. 音频

19 + 5 个音效/音乐：`LA Main Scene intro`、`LA intro Final`、`ForestWalk`（BGM）、砍伐 `Cut SFX`、`Cut Stone, coal`、`Swing 4`、`Walking sound effect_2`、`Swimming`、`Water Splash`、`Unlock SFX`、`New Base Unlock SFX`、`Yay Saved`、`Cheer_02`、`LevelFail`、`StarCollect`、`Drink potion 2` 等（44.1 kHz）。

9. 第三方库清单

Cinemachine · Easy Character Movement · A* Pathfinding Project · AI Navigation(NavMeshSurface) · Obi Rope · DOTween/Pro · Feel (MoreMountains Feedbacks/Tools) · Nice Vibrations · Toony Colors Pro 2 · AllIn1Vfx · Epic Toon FX · Cartoon FX · TrailsFX · Mobile Optimized Water · Distant Lands Fish Flock · Coffee UIEffect · TextMeshPro · Unity Visual Scripting · Unity IAP · Addressables · In-game Debug Console · Newtonsoft.Json。

10. 方法与局限

- 方法：直接下载 GameDistribution CDN 上的 Unity WebGL 构建与 Addressables bundle，自写 UnityWebData 解包脚本，用 UnityPy 读取序列化资产，自写 global-metadata v31 解析器导出 602 个 C# 类的字段与方法名。
- MonoBehaviour 具体数值（如每级价格、伤害）存放在剥离了 TypeTree 的序列化数据中，需结合 wasm 中的类型信息还原；本报告只确认了少量（如 `maxHealth = 100`、`coldResist = 5`）。如需全部数值，下一步可用 Il2CppDumper 对 wasm+metadata 生成 DummyDll，再用 AssetStudio/UnityPy 带类型导出。
- 函数体逻辑（算法细节）位于 wasm 机器码，本报告的流程描述基于类名、字段名、方法名、动画事件和场景结构推断，属高可信推断而非反编译源码。