

CH (ChangHe) Plan

Owner

Development: Peng Tao(XiaoP)

Revision History

Version	Date	Author	Comments
1.0	9/1/04	XiaoP	Initial release

Table 1: Revision History

Table of Contents

一、观念（Vision） 3

 1. 主题描述 3

 2. 世界的哲学抽象 3

 3. 物质（Sub） 4

 4. 物体（Obj） 4

 5. 世界实例（World Instance） 5

二、系统层次与体系（Hiberarchy & Architecture） 5

 1. 内容（Content） 5

 2. 实现系统（Implement System） 6

三、客户形式和界面（Client & Interface） 7

 1. 客户形式（Client form） 7

 2. 人类界面（Client Human Interface） 7

 2. 1 第一人称 3D视角 7

 2. 2 第三人称受限 3D视角 8

 2. 3 第三人称策略管理 9

 2. 4 辅助场景 9

 2. 5 特殊效果 10

四、内容产生工具（Content Tools） 10

一、观念 (Vision)

1. 主题描述

在一个拟真世界中，存在着各种不同类型的物体，其中也包括所有的外部接入者，比如人类玩家、NPC 等等，他（她、它）们尽可能以现实世界的方式进行交互或相互作用（Interaction），整个虚拟世界将尽可能模拟现实世界进行发展和演化（Evolvement）。

这个虚拟世界的总体被称为“长河世界（ChangHe World）”，简称为 CH。

2. 世界的哲学抽象

CH 世界仅由“物体（Objects）”构成，各种各样的物体自然构成了一个世界。这里没有别的实体，就只有物体，世界充满了物体，这是哲学抽象后的结果。而“物质（Substance）”是一个抽象的概念，一种物质定义可以对应多个物体实体。世界中每一个物体都是属于一种物质的，物体之间的相互联系相互影响促成了整个世界的演化发展。

“空间（Space）”并不算作世界单独的一部分。因为本质上，物体和空间是不可分割的，任何物体都处于（属于）某一个空间之中，而任何空间都应该属于某一个物体（称为“包容器物体”），有了物体，自然就有了空间。在设计上有一个总空间是默认的，由于不能给它设置包容器，所以它就被预先定义。所有的物体就在这个总空间内诞生，它们又有各自的子空间，子空间里面又有各种各样的物体，这样的世界便是无穷无尽的。

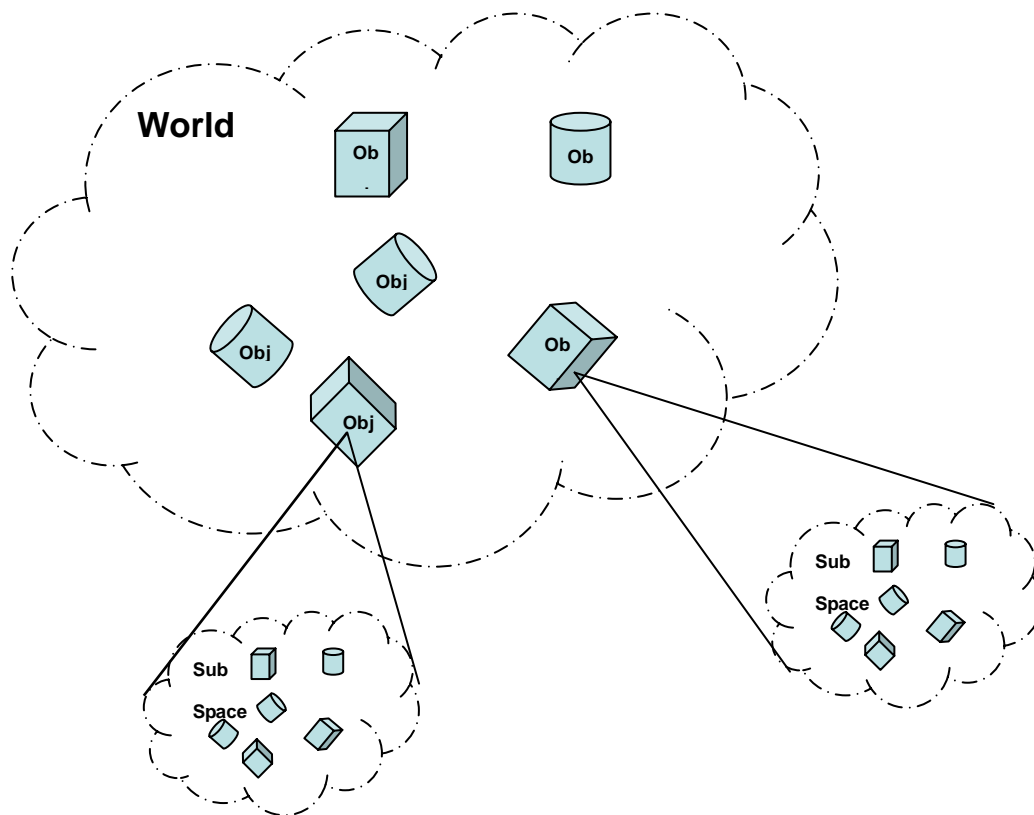


Fig.0 CH 分层世界

在设计上，实体（对象）和定义（概念）被分别对待。

3. 物质 (Sub)

物体本质特征的定义体。

对于一种物质，它有哪些特征，也就是它具有什么属性 (Property) 及其意义；该物质的一个实例 (物体对象) 诞生时它所有属性的初始值是什么；由于要让人类接入来观察这个世界并交互，则它还规定了在某个或某些属性组合下对应实体对外的图形表现是什么样的，这些属性组合被称为状态 (State)。属性组合和特定状态是相对应的，但为了数据查询的方便，它们会被单独定义并在物体对象数据上得到统一。

物质还要定义实体的行为 (Action) 有哪些，行为的图形表示是什么，每个行为怎样实施；它执行一个行为会向行为目标对象给予什么样的影响，也就是发出什么样的相互作用 (Interaction)，相互作用的图形表示是什么；它自己接受什么样的相互作用，状态受到什么样的影响，怎样去响应，其图形表示又是什么。

总之，一个物质定义体系将产生整个世界所有的基本物理定律，其图形表示只是方便人类接入者观察。

特别说明，一种物质还要定义多个内部刺激源 (Inner Stimulator)。对于一个实体对象，这些刺激源将被定期检查，特定的刺激源在一定条件满足后会被触发，接着对物体自身施加一个影响 (相互作用)，从而引发物体的响应机制。这会使得物体产生一些经常性活动，对外表现为物体具备某些内在动机。这可以模拟有生命物质对应的实体为了维持生命而进行的经常性行为。比如简单的生长现象，动物获取水、空气、食物的过程等等，都可以看成这种经常性行为加以定义并不断被用来响应内部刺激。这种动机被称作“本能 (Instinct)”，它们被周期性的检查驱动。

物质可以抽象定义并多父继承。

4. 物体 (Obj)

物体是对应于某种物质的一个实体对象，一个物体只能属于一种确定的物质，它的行为、响应按照这种物质所定义的来进行。一种物质可以有多个物体实体。世界就由这些物体组成，世界的所有数据就在于这些物体对象数据的总和，这些数据存在于一个统一的数据库系统中。所有物体的状态勾勒出了整个世界的面貌，它们改变世界也随之而变。这个数据库系统总是在各种情况下被不断改变着的。

系统将物体的一切行为 (Action) 实现为一种刺激响应。无论是自发的还是外在的，物体都在一定的刺激 (相互作用) 下进行响应行为，只不过自发的响应对外便会表现为一种主动行为，对外来相互作用的响应则表现为一种被动行为。如果被动行为再次对目标产生相互作用的话，作用和反作用的演化机制将在系统中自然形成。

复杂的响应行为序列将构成“高级人工智能 (AAI)”。

一种更复杂的宏观行为模式——**剧本 (Scenario)**，用剧本来实现各种人工智能、英雄 AI (历史人物 AI) 按照历史进行的既定活动。定义一种特殊的物质——剧本执行者 (God)，可以定期刺激它执行一系列精心设计的剧本行为，这些行为对指定的角色目标产生相互作用。任何其他物质实体都必须接受并响应 God 发来的相互作用，这些相互作用往往会无条件地改变指定物体的属性和无条件地触发物体的内部刺激源。

虽然整个世界通过物质实体的各种运动来前进发展，但世界的面貌大体上将按照历史而定。作为娱乐性的系统来说，AI 可以改变世界面貌，人类接入者可以在许可范围内改变世界，但是不能更改主要历史。人类接入者能够参与历史事件，甚至还可以影响一些野史和文学化的事件发展 (会不会总有一些玩家要在风波庭拼命实施营救岳飞的行动，但应该是不能够成功的)。

需要注意的是，系统具备一些专用子系统用于接入者之间的消息交互，是独立于世界模拟体系之外的，比如聊天系统、外部交易系统、外部信息系统 (如管理系统、BBS) 等等^{TBD¹}。但世界中也会模拟一些实体之间的信息传递行为，比如交谈、信件往来这类。

¹ **TBD** means to be decided

5. 世界实例 (World Instance)

宏观上, CH 持续进行**演化 (Evolution)**, 这是系统运行的时间效应。举例来说, 系统会在一定真实时间内 (比如 1 年) 完整地演绎一段历史 (比如 300 年的模拟历史时间段)。

为了能够演绎完整的世界历史又不失娱乐性, 底层系统中会同时存在多个时间段 (历史段) 的世界实例 (World Instance) 以包含整个世界历史, 这些实例构成世界实例组 (World Instance Group)。每个实例均处于特定的历史时期, 比如间隔 300 年一个实例, 并按照真实的历史为主线发展。另外, 系统还维护着一个“现代同步实例 (Super World Instance)” 和一个“假想未来的实例 (Future World Instance)”。所有这些实例都是并行推进的, 实例之间只能允许有限的穿梭跨越。

对于一个接入者 (玩家), 可能先在 21 世纪生活, 接着又想进入远古的世界中去。为了不破坏系统按照历史的正常演绎又保证一定的娱乐性, 系统将会允许“实例跨越 (Instance Spanning)”。当然, 一个接入者从某个实例 (可能在 20—21 世纪) 进入到另外一个实例 (3—5 世纪) 是要付出代价的。

每个实例在演化推进过程中都将自然地走完历史的全过程, 最终融合进 Super Instance, 这个特殊的实例与当前的现实世界保持同步并一直存在。同时, 不断有新的实例产生于历史起点, 加入到 CH 实例组中来。为了满足娱乐性, 系统也会一直维护着一个 Future Instance, 比如定位在 25 世纪。

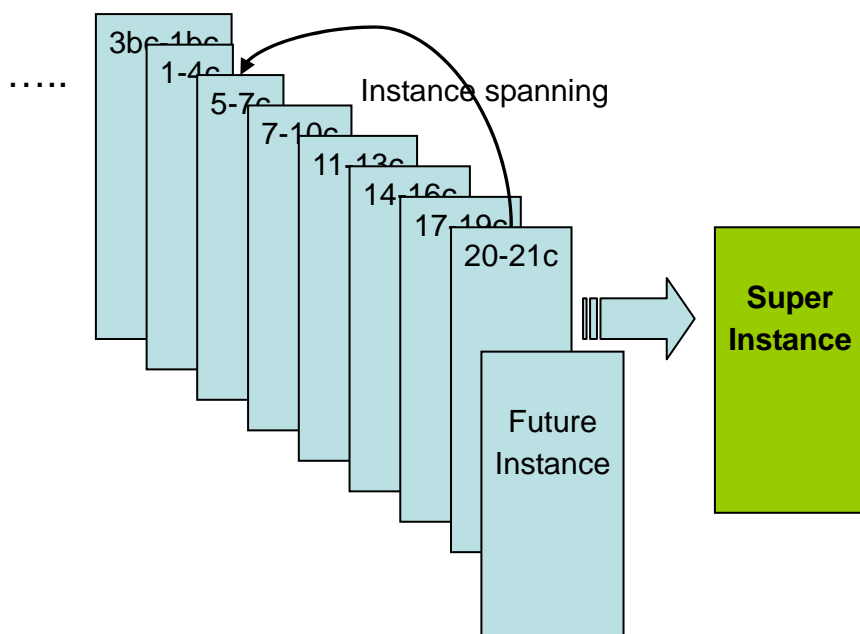


Fig.1 CH World Instances, 固定历史段宽度=300 年

CH 对于人类接入者来说, 它要超越娱乐方式成为一种**生活方式**, 并且以虚拟干涉现实、现实指导虚拟的形式搭建出和现实社会平行的虚幻社会。虚拟干涉现实, 是指在虚拟世界中, 接入者可以进行能够影响现实的活动, 比如电子商务、媒体广告、教育等。现实指导虚拟, 是指虚拟世界中物体的相互作用和世界的演化要尽量遵循现实的规律, 包括自然的和社会的。

二、系统层次与体系 (Hierarchy & Architecture)

1. 内容 (Content)

二次开发人员利用高级语言 (如 C++、Lua 等) 和各种客户端工具创建世界的各种基本物质及其相互作用

用规律。三次开发人员和众多有权限的接入者可以用高级语言（如解释语言）创建高级 AI 以及世界的演化规则和既定历史，世界在进行模拟演化的同时增加新的开发数据，在不断扩充的情况下提高拟真近似程度。

许多基本的底层规律会编译为低级代码直接链入系统以提高性能和稳定性^{TBD}，历史剧本以及高级AI都在高级语言下进行解释执行。

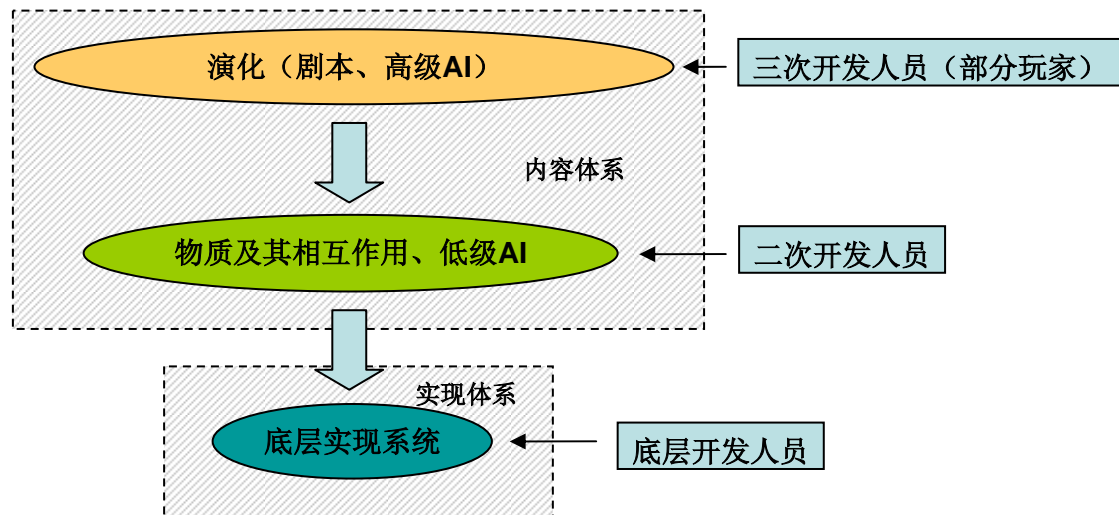


Fig. 2 内容和实现系统的层次

2. 实现系统 (Implement System)

CH 的底层实现系统是一个大型的网络体系，从工作模式上看，分为**服务系统 (Server System)**和**客户接入群 (Client System)**两大部分。

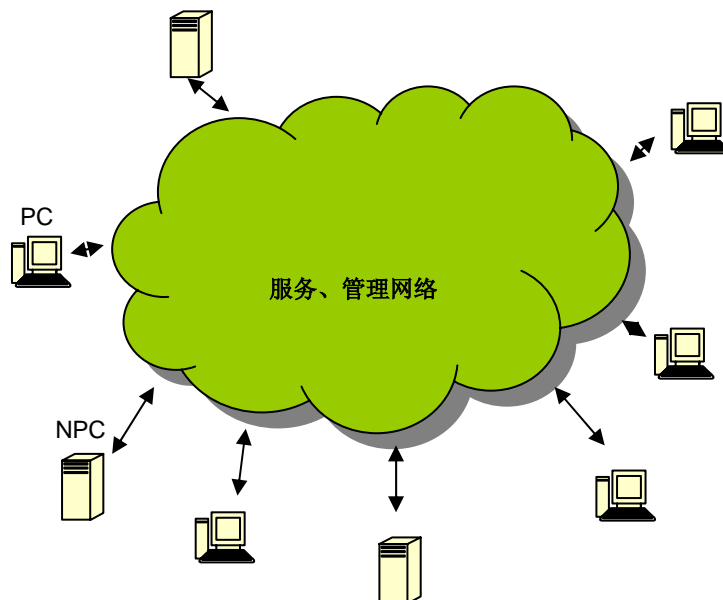


Fig.3 网络体系模式

服务系统内部不间断地维护、更新世界实例组，并且具有多个分布在各处的客户端接入点（分流、路由）。服务系统是个统一的实体，统一在同一个核心数据库系统上。也就是说，在一个历史时间段的世界实例中，世界是唯一的并且是统一联系的，这从本质上不同于独立服务器组的分区制。另外，考虑到在核

心数据库上物体可以进行历史段世界实例的跨越，那么，从游戏意义上来说，CH 宏观世界也只有一个，即无论两个接入者从何处接入、在何时接入、进入到哪个实例中，他们都是能够互相影响的。

服务系统除了维护世界的模拟以及处理客户端的接入交互，还维护一些独立于世界模拟以外的信息系统，比如聊天系统、外部交易系统、外部信息系统（如管理系统、BBS）等等^{TBD}。这些外部信息系统的部分数据来自于中心数据库，在一定权限下，它们也可以更新中心数据库的部分信息。

客户接入群是大量的网络接入客户端，它们可以得到角色物体的相关信息，也不断向服务器发出操作指令指示代表物体的行动。

三、客户形式和界面 (Client & Interface)

1. 客户形式 (Client form)

CH 客户端具有多种形式，可以是 NPC 客户或者人类接入者，甚至可以是它们的各种组合。客户端和服务端之间的交互按照统一的应用层协议通信。

对服务系统而言，无论是人类还是人工智能，所有的客户接入端都被同等对待。连接界面 (Link Interface) 具备统一规范的协议，各种数据连接都被封装在连接界面内，包含了消息通道、主体通道、更新通道等等^{TBD}，各通道详细的协议解释参见协议文档。

2. 人类界面 (Client Human Interface)

在客户端，NPC 接入者可以不需要任何显示界面，NPC 接入者可能仅仅是远程的响应机器，它响应任何相互作用事件，这些事件可能是来自外部也可能来自它内部^{TBD}。NPC 在一定权限下，可以具备自己单独的高级 AI。

但对于人类接入者的客户端，除了一般的数据界面，还需要一个复杂的图形的显示界面。由于 CH 世界的多样性和全开放性，人类的观察界面必须具备多个自由度。

基本的角色视觉场景使用“第一人称 3D 视角”和“第三人称受限 3D 视角”界面。它们表达的场景信息以及观测范围与角色的位置等状态相关。

2.1 第一人称 3D 视角

第一人称 3D 视角类似于 FPS 类型游戏，操作者会在即时战斗中或者观察周围情况时经常使用。



Fig.4 HalfLife2 中的第一人称 3D 视角

2.2 第三人称受限 3D 视角

第三人称受限 3D 视角是在第三人称视角的情况下，为了避免摄像机被角色周围物体遮挡，而使摄像机根据周围情况不断调整。



Fig.5 Farcry 中的第三人称视角



Fig.6 天堂 2 中的第三人称视角

2.3 第三人称策略管理

在进行策略性交互时，使用大场景 3D 视角。在这种场景下，物体一般是大量的、分离的，摄像机的位置（操作者观察位置）和观察角度是任意的，但一般都位于各种物体的外部，起场景观测的作用。操作者在这样场景下的操作一般受到限制，可以执行的操作一般都是策略性的。比如：指挥战争、管理建设等等。CH 中这样的模式称为**第三人称管理**。



Fig.7 Homeworld 2 中的大场景策略视图和观测视图

2.4 辅助场景

部分虚拟物质能够产生辅助场景，比如“瞄准镜”、“显微镜”、“图示仪器”等等。它们将会使用特殊的视角界面。可以望远，也可以表示放大的显微镜头。辅助场景可以是全屏的也可以是部分对话框视图。

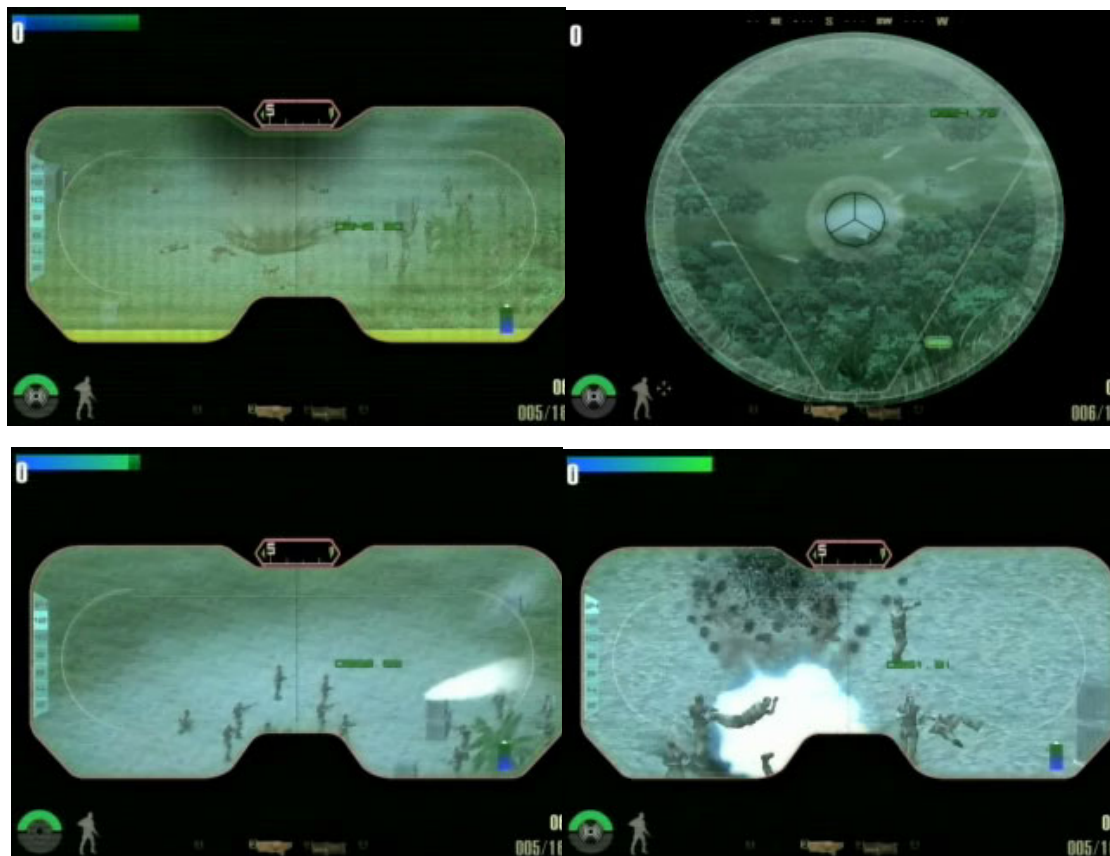


Fig.8 Farcry 中的瞄准镜、望远镜产生的辅助场景（long view distance）

2.5 特殊效果

有时候，特定天气情况下或者在使用了特定物质后，场景需要表现为特殊效果。

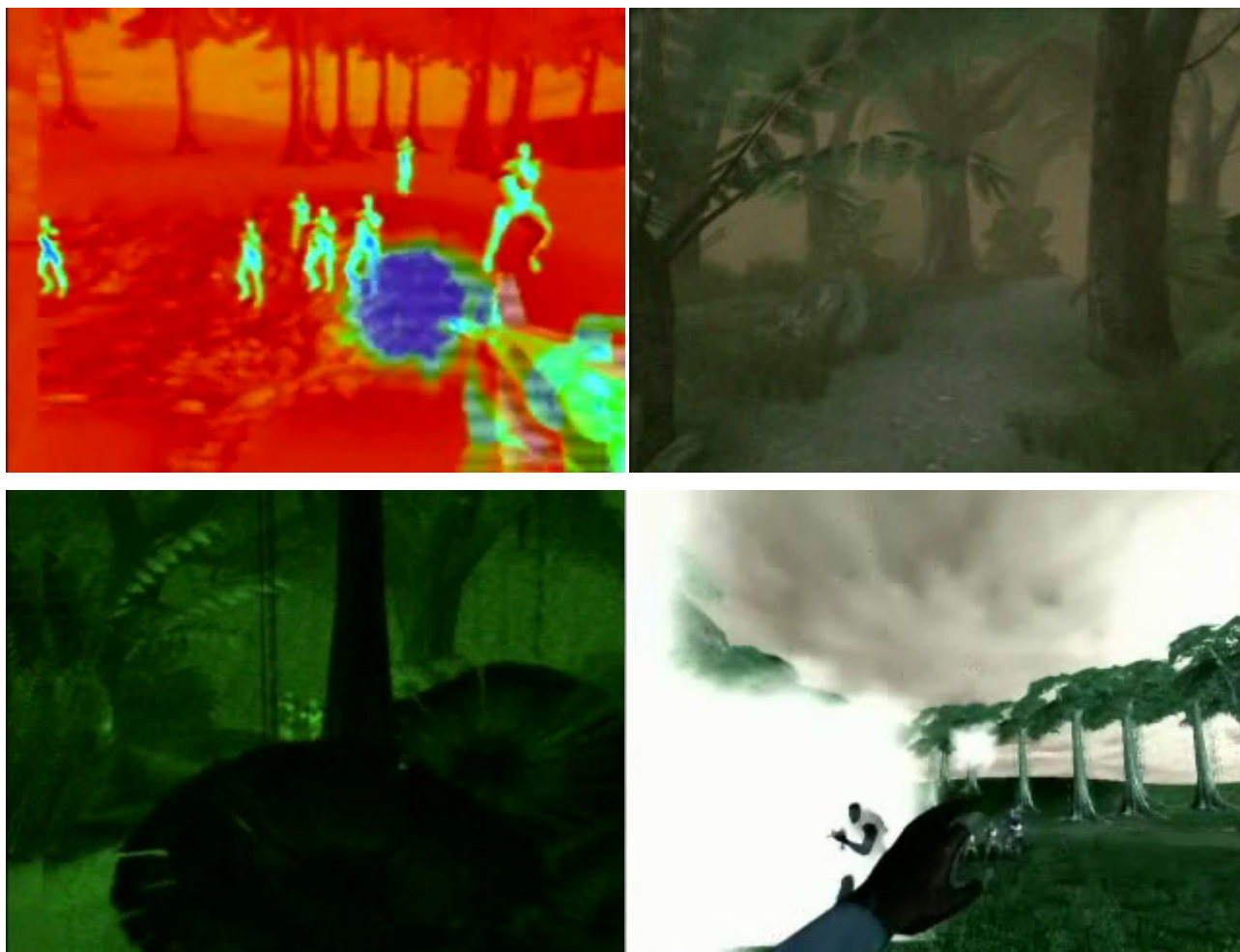


Fig.9 Farcry 中的红外仪场景、浓雾天气、夜视仪场景、角色视觉暂时受损后场景

四、内容产生工具 (Content Tools)

二次开发和三次开发人员使用丰富的内容产生工具，这些工具使他们能够非常容易地产生世界底层的物质信息、规律以及高层剧本和高级的 AI。在可以使用高级脚本语言（比如 Lua）写入数据文件的同时，内容产生工具要尽量图形化和自动化地产生脚本（比如 Lua）和数据文件（比如 XML），一些工具还可以对脚本文件进行编译并转换为封装的数据文件格式。

建模工具使用已有的建模工具并在其上增加插件输出，比如 Maya，3DMax 等。

部分内容产生工具能够通过特定通道以及协议与服务器交互，作为内容产生器客户端。测试通过的内容数据最后被加入到 CH 世界数据中心。

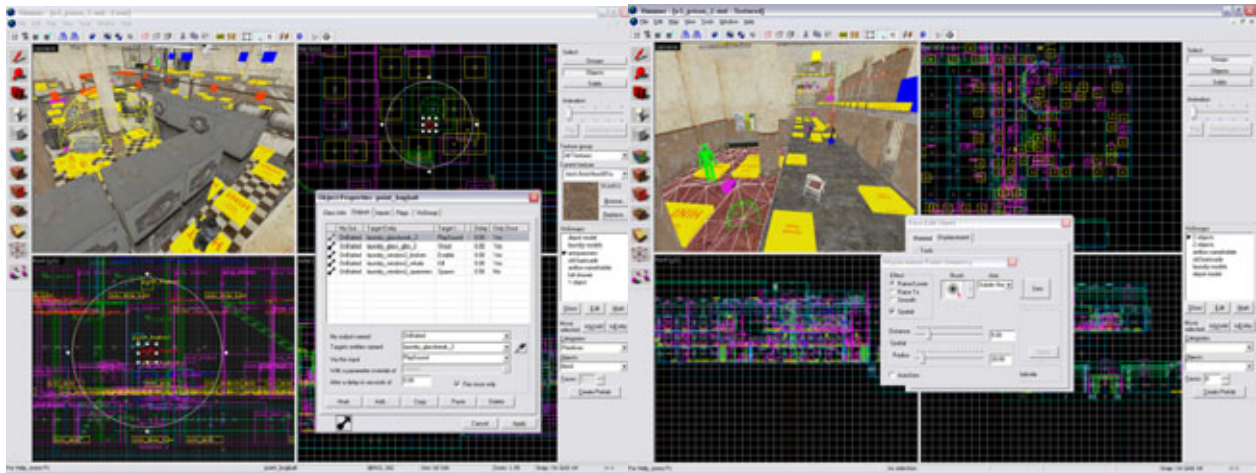


Fig.10 HalfLife 2 的内容产生工具

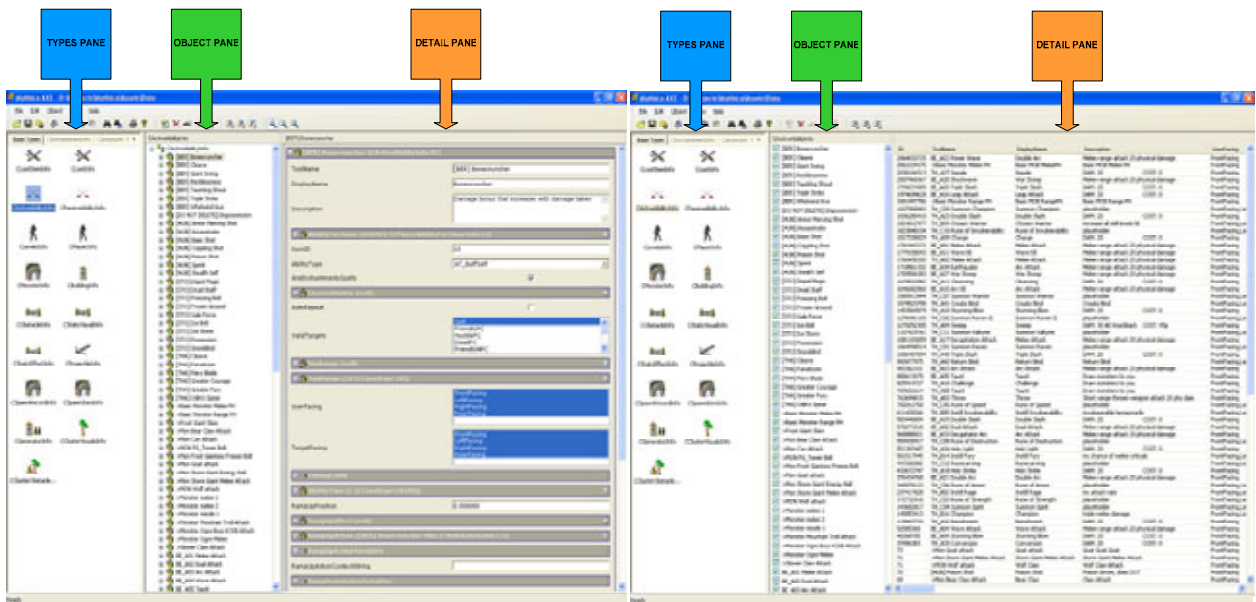


Fig. 11 Mythic 的 AXE 工具