

# 本科毕业论文（设计）

论文题目： 地质灾害信息的多事件源融合软件开发

指导老师： \*\*\*

学生姓名： \*\*\*

学 号： \*\*\*\*\*

院 系： \*\*\*\*\*

专 业： \*\*\*\*

毕业时间： \*\*\*\*\*



---

## 关于本文档

本文作者对本文保留所有权利。未经本人允许，不得对本文进行复制、修改、出售或进行任何商业性利用。

本文档使用  $\text{\LaTeX}$  编写，使用 Tectonic 引擎编译。

以下是本文档使用的字体及其许可证：

- 正文（简体中文）：Noto Serif CJK SC  
许可证：SIL Open Font License, Version 1.1
- 正文（繁体中文）：Noto Serif CJK TC  
许可证：SIL Open Font License, Version 1.1
- 正文（日文）：Noto Serif CJK JP  
许可证：SIL Open Font License, Version 1.1
- 正文（拉丁文）：Noto Serif  
许可证：SIL Open Font License, Version 1.1
- 标题（简体中文）：Noto Sans SC  
许可证：SIL Open Font License, Version 1.1
- 标题（拉丁文）：Noto Sans  
许可证：SIL Open Font License, Version 1.1
- 代码：Roboto Mono  
许可证：Apache License, Version 2.0
- 代码（简体中文）：Noto Sans SC  
许可证：SIL Open Font License, Version 1.1

本文中出现的各种组织、部门等名称皆按其原有的全名或缩写进行标注，各语言、文字、地区的名称及其格式皆遵循国际通用标准。以上所述内容没有任何政治暗示，不代表作者及任何参与文中所述应用的开发人员的政治倾向。



## 摘 要

地质灾害是一种具有突发性、难以预测的自然灾害。历史上大规模的地质灾害曾造成了重大的经济损失和人员伤亡。针对地质灾害，全球各地有相当多专门测定并发布相关信息的机构，但是其信息格式和语言往往各不相同，手动整合相当不便。

针对以上问题，本文分析、设计并实现了一套融合多事件源地质灾害信息的统一接口，并基于此接口设计并实现了两套分别适用于安卓系统和 Windows 系统的图形用户界面。界面以文字显示、地图绘制和语音播报实时呈现地质灾害事件。整个系统分为事件源接口层、事件源实现层、应用核心层和应用实现层。

本研究的主要特点是：(1) 科学性，系统开发采用多层开发架构，符合规范的软件开发流程，系统结构清晰合理，经测试系统迅速、准确、有效；(2) 实用性，至本文截稿，应用已更新数十个版本，并在地质灾害研究者社区得到好评。

**关键词：**地质灾害，地理信息系统，多事件源，信息应用，语音播报



## Abstract

Geohazard is a type of natural hazard that occurs suddenly and is hard to predict. Historically, large-scale geohazards generated severe economic losses and fatalities. There are many dedicated authorities around the world that observe and publish relevant information. However the format and language of the information from different authorities often differ, and integrating them manually is a big trouble.

Regarding the problems above, this article describes the process of analysis, design, and implementation of a unified multi-source geohazard information interface and two graphical user interfaces for Android and Windows respectively. These interfaces represent geohazard events in real-time with text description, map drawing, and text-to-speech output. The whole system consists of four layers: event source interface layer, event source implementation layer, application core layer, and application implementation layer.

The main characteristics of this research include: 1) scientificity—The development of the project employs a multi-layer architecture, which conforms to the software development process with a clear and reasonable structure, and the system is responsive, precise, and effective in tests; 2) practicality—by the completion of this article, the project have gone through dozens of version releases, getting positive feedbacks from the geohazard researching community.

**Keywords:** geohazard, geographic information system, multiple event sources, information application, text-to-speech



## 目 录

第 1 章 引言 .....	1
1.1 研究背景 .....	1
1.2 研究现状 .....	1
1.3 研究内容 .....	1
1.4 章节结构 .....	2
第 2 章 需求分析 .....	3
2.1 功能性需求 .....	3
2.2 非功能性需求 .....	4
2.3 本章小结 .....	5
第 3 章 概要设计 .....	7
3.1 系统架构 .....	7
3.2 接口层接口 .....	8
3.2.1 事件 .....	8
3.2.2 生成器 .....	10
3.2.3 事件源 .....	11
3.2.4 本地化资源 .....	11
3.2.5 语音接口 .....	12
3.2.6 地图接口 .....	13
3.3 核心层概要 .....	15
3.3.1 生成器管理器 .....	15
3.3.2 语音和音效 .....	15
3.3.3 核心工作器 .....	16
3.3.4 其它辅助类型 .....	18
3.4 本章小结 .....	18
第 4 章 详细设计 .....	19
4.1 组件结构 .....	19
4.2 视觉设计 .....	19
4.2.1 配色方案 .....	19
4.2.2 地图设计 .....	22
4.3 事件源层简述 .....	24

4.4 核心层逻辑 .....	25
4.5 音效设计 .....	29
4.6 本章小结 .....	29
第 5 章 实现与测试 .....	31
5.1 实现 .....	31
5.2 代码分析 .....	33
5.3 测试 .....	34
5.4 本章小结 .....	35
第 6 章 总结 .....	37
参考文献 .....	39
附录 A 默认配色参考 .....	41
附录 B 音效乐谱 .....	43
附录 C 版本历史 .....	45
致谢 .....	49
诚信声明 .....	51

## 图形列表

2.1 用例	4
3.1 系统架构	7
3.2 宏观数据流	8
3.3 事件对象类	9
3.4 分组键、单位键和修订键	10
3.5 生成器	11
3.6 事件源接口	12
3.7 本地化资源	12
3.8 语音条目对象类	13
3.9 地图元素	14
3.10 生成器管理器	15
3.11 语音工作器和声音播放器	16
3.12 核心工作器	17
4.1 底图样本	23
4.2 地图元素	23
4.3 生成语音条目	26
4.4 插入语音条目	26
4.5 语音工作器	27
4.6 事件分组	28
4.7 “添加到事件分组”	28
5.1 安卓端应用界面	32
5.2 Unity 端应用界面	33
5.3 代码分析器部分单元测试	34
5.4 核心层部分单元测试	34
A.1 默认配色	41
B.1 音效乐谱	43



# 第 1 章 引言

## 1.1 研究背景

地震是地球上主要的自然灾害之一，火山活动通常伴随着地震活动，而大规模的海域地震和海底火山喷发又往往会引发海啸。这些地质灾害一直以来都受到人们的关注，尤其是在火山地震带等地质活动活跃的地区。由于地质灾害具有突发性，难以预测，使得即时警报和灾后救灾显得尤为重要。<sup>[1]</sup> 在这其中，信息的传递是重要的一环。

长久以来，世界各地的研究人员建立了大量相关观测网，全球各地有着相当多测定并发布相关信息的机构，但是其获取方式各不相同，手动整合又相当麻烦。

## 1.2 研究现状

当前已有的公开地震信息软件中，大部分是面向特定地区的用户的，如面向日本的 JQuake、SREV、Quarog，面向台湾的地牛 Wake Up!，面向美国的 ShakeAlert。这些软件从特定地区的机构获取信息，其呈现一般比较详细，但地区局限性强。

也有一些软件通过整合各机构的信息呈现全球各地的地震信息，但这些软件一般只展示基本的震源位置、发震时间和地震震级，不会展示不同机构提供的不同详细信息。

此外，还有一些能够自行产出地震信息的软件，如 GlobalQuake 通过处理实时测站波形数据计算地震的震源位置、发震时间及震级。<sup>[2]</sup> 但其产出的信息并非官方权威信息，准确度有限。

## 1.3 研究内容

本系统是一个呈现地震相关信息的应用，其通过从不同事件源获取地震、海啸、火山等信息，将其整合于一个统一的接口，并加以文字显示、地图呈现和语音播报，且具有一定的预警功能。

## 1.4 章节结构

本文后续的章节安排如下。

第 2 章：需求分析，分析本应用的功能性需求和非功能性需求。

第 3 章：系统概要，设计本应用的整体结构、各组件的特性及其之间的关系。

第 4 章：详细设计，规定本应用的环境，进行视觉设计和音效设计，并整理核心逻辑。

第 5 章：实现与测试，说明本应用的一些实现细节和测试方法。

第 6 章：总结本文的内容。

## 第 2 章 需求分析

本系统的开发暂时分为两个阶段。第一个阶段为测试版开发阶段，假定用户具有基本的地震学知识<sup>1</sup>，该版本不适合一般大众使用。第二个阶段为正式版开发阶段，对象用户为一般大众。设计时，我们认为上述高级用户的需求为一般大众的需求的子集，因此能安全地仅针对高级用户进行需求分析。本文作为早期开发过程产出的文档，仅对主版本 0 做需求分析。

### 2.1 功能性需求

为方便描述，我们将本系统获取信息的来源称为“事件源”（英文：event source），从事件源中获取到的一个基本信息单位称为一个“事件”（英文：event）。

如第 1.3 节所述，本应用主要功能是呈现地震相关信息，通过从不同事件源获取地震、海啸、火山等信息，将其整合于一个统一的接口，并加以文字显示、地图呈现和语音播报，且能够有限程度上发出预警。本系统的用例十分简单，如图 2.1 所示，用户可以通过系统向事件源查询事件，事件源通过系统向用户推送事件。此外用户可以进行应用设置改变系统内部的一些行为。

值得说明的是，本系统用例简单不代表其复杂度就低。本系统的复杂度主要体现在如何有效地整合不同格式的信息，并将其以各种形式准确地呈现。

下面对每个用例进行功能性需求分析。

1. 呈现事件：系统需要能以简要和详细的形式呈现获取到的事件的信息，呈现方式包括文字、声音以及地图。呈现信息时需要使用正确的区域性，确保包括但不限于语言和字体符合用户的设置。
2. 查询事件：系统提供查询历史事件和查看已接收事件的功能。
3. 推送事件：系统能从用户指定的事件源获取其推送的事件并呈现，有必要时根据预警策略对用户发出有限的预警。
4. 应用设置：系统提供一系列的个性化设置，包括信息的呈现格式、地图的绘制风格、启用的事件源和相关参数、音效的设定和语音播报的语言等。

<sup>1</sup>旧称“地震信息的高级用户”，该名词直接引自日语的“地震情報的高度利用者”，后因在中文语境下容易引起歧义而弃用。

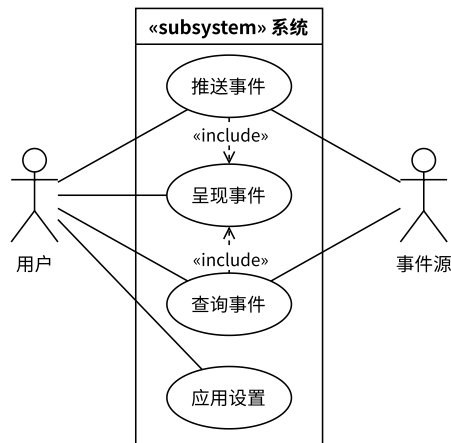


图 2.1 用例

Figure 2.1 Use Cases

目前本应用计划支持的事件源如下。

- 日本气象厅地震火山相关防灾信息 Atom 推送（日文：气象厅防灾情报 Atom フィード・地震火山）
- 台湾中央气象署气象资料开放平台（繁体中文：中央气象署气象资料开放平台）
- GlobalQuakeServer API
- 美国地质调查局实时 QuakeML 推送（英文：United States Geological Survey Real-Time QuakeML Feed）
- 欧洲与地中海地震中心实时通知（英文：European-Mediterranean Seismological Centre Real-Time Notifications）
- 美国海啸警报中心（英文：United States Tsunami Warning Centers）
- 印度尼西亚气象、气候及地球物理局开放地震数据（印度尼西亚语：Data Gempabumi Terbuka Badan Meteorologi, Klimatologi, dan Geofisika）

以上事件源皆为合法、有对外公开 API 的事件源。由于中国大陆地区没有类似的公开 API，故没有相关的事件源。特此说明。

## 2.2 非功能性需求

以下为本应用的性能需求。

- 系统需要迅速将接收到的信息以简要形式呈现在主界面。具体定量为：从接收到新的事件到界面响应的时间不超过 500 毫秒。
- 系统需要在可接受的时间内回应用户打开详细信息的请求。具体定量为：打开任意事件详细信息，界面响应时间均不超过 1 秒。
- 系统需要迅速将接收到的信息以声音提示。具体定量为：若接收到的事件需要播放音效，则从接收到事件到该音效进入音频引擎混音序列，延迟不超过 500 毫秒。
- 地图呈现应当足够流畅。具体定量为：即便在极端情况下，地图的渲染帧率也不小于 10 帧。

除去上述需求，对于本应用，实际影响用户体验的主要是各事件源的发报延迟以及网络延迟。由于这些延迟难以通过本系统的开发进行优化，故不再作讨论。

另外，本应用有较高的可扩展性需求，需要考虑到未来增加事件源的可能，使事件源的新增不损害现有系统的功能。

### 2.3 本章小结

本章分析了本应用的目标用户群体，并在此基础上分析了用例，确定了应用的功能性需求和非功能性需求。



## 第 3 章 概要设计

### 3.1 系统架构

本系统的架构大致分为两个部分：与事件源交互的部分及与用户交互的部分。与事件源交互的部分分为事件源接口层（英文：event source interface layer）和事件源实现层（英文：event source implementation layer）。与用户交互的部分分为应用核心层（英文：application core layer）和应用实现层（英文：application implementation layer）。

为方便描述，各层级简称如下：

- 事件源接口层：接口层（英文：interface layer）
- 事件源实现层：事件源层（英文：event source layer）
- 应用核心层：核心层（英文：core layer）
- 应用实现层：应用层（英文：application layer）

系统架构如图 3.1。

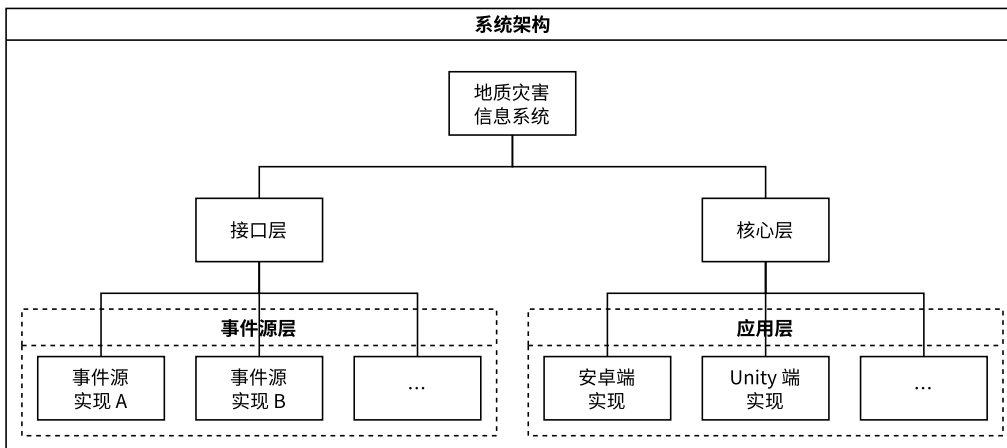


图 3.1 系统架构

Figure 3.1 System Architecture

接口层为所有事件、事件源交互对象的抽象接口，事件源层是这些接口的实现。核心层使用事件源层的实现接收实现接口层接口的事件，并进行一系列的处理生成方便应用层直接使用的视图模型，并将其下发至应用层，最终应用层将这些视图模型转化为视图呈现给用户。

用户也能在界面中查询历史事件，事件请求分别经过应用层和核心层，最后由事件源层向事件源发送请求。响应的事件同样由上一段描述的数据流进行处理。

宏观的数据流如图 3.2。

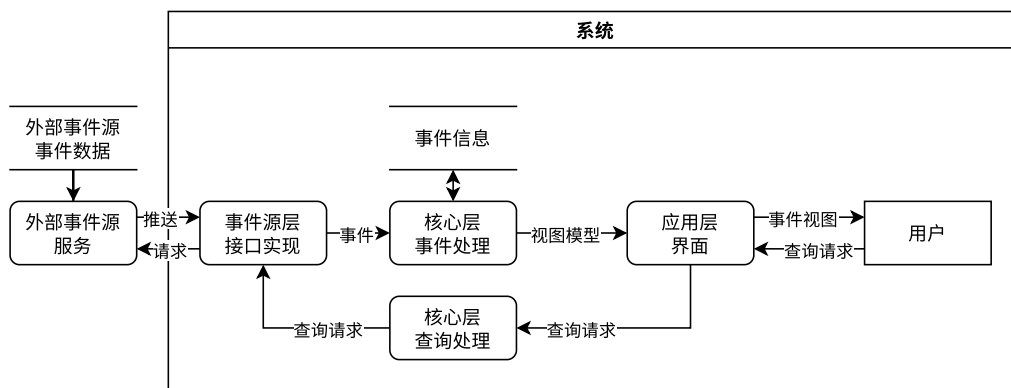


图 3.2 宏观数据流

Figure 3.2 Data Flow Outline

因篇幅限制，事件源层和应用层的设计本文仅简要概述。

### 3.2 接口层接口

本节描述接口层中定义的接口。

#### 3.2.1 事件

为了方便描述，我们将从事件源中直接获取到的表示事件的对象称为“原始事件对象”（英文：raw event object）。原始事件对象是事件源给事件源层推送的事件最原始的呈现，不同事件源所提供的原始事件对象一般拥有不同的模型。为了使事件最终在应用层的呈现效果统一，我们需要从原始事件对象中提取一些如事件位置、事件时间等的基本信息，将其封装到一个统一模型的对象中。我们称这类对象为“事件对象”（英文：event object）。事件对象包含标题、事件源名称、主体、时间等基本属性。一个事件对象包含多个事件属性。事件属性（英文：event property）对象用于封装一些事件对象本身没有涉及到的，但是比较关键的属性。事件对象类如图 3.3。

为了缓解向用户呈现事件时列表中出现大量互相关联的事件的问题，我们把相关的事件对象合并到一个事件分组（英文：event group）中以减小信息的

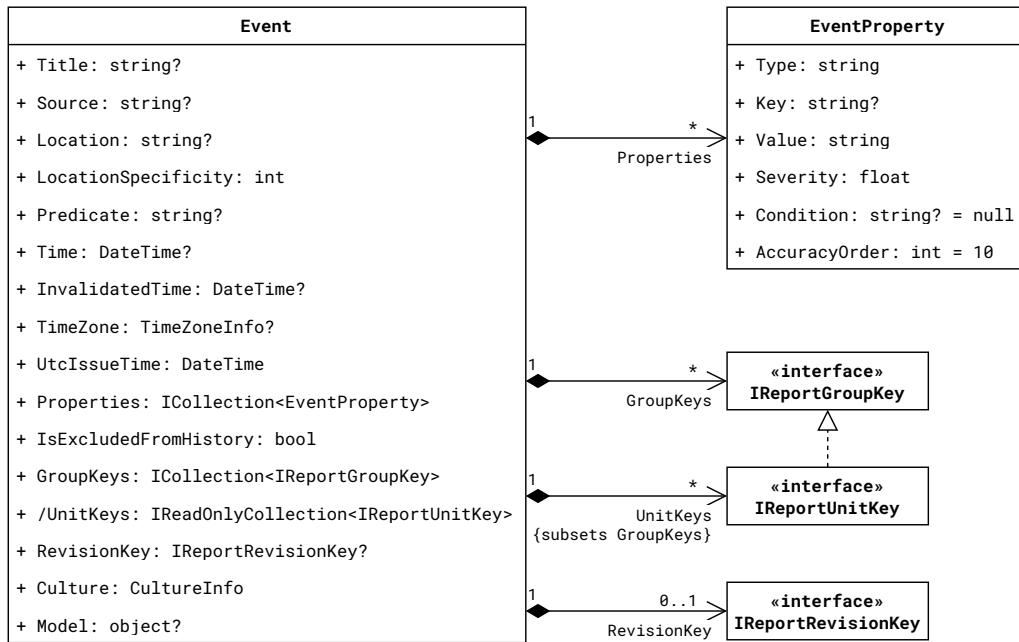


图 3.3 事件对象类

Figure 3.3 Event Object Class

重复性。此外，一些事件可能会随着时间被更新。对于任意两个事件，如果两者的原始事件对象模型相同，且其中一个事件起到更新另一个事件的作用，那么认为这两个事件属于同一个事件单位（英文：**event unit**）。

分组键、单位键和修订键是一组用于识别事件之间的上述关联性的对象。分组键（英文：**group key**）描述事件用于分组的一些特征；单位键（英文：**unit key**）用于识别事件的事件单位；修订键（英文：**revision key**）用于识别事件在事件单位内的更新顺序。这三类对象以接口定义，如图 3.4。

可排序分组键（英文：**sortable group key**）是一类特殊的分组键，用于模糊匹配分组。例如，对于一次地震，不同机构很可能会给出不同但是相近的地震要素，识别这些要素的关联性就会使用到可排序分组键。具体的匹配逻辑参见第 4.4 节。

由于一类可排序分组键之间有明确的顺序顺序，规定对于任意一个可排序分组键，与其预匹配的所有可排序分组键在上述定义的顺序下互相相邻，且与其匹配的所有可排序分组键必定也与其预匹配。在有一类可排序分组键数量非常多的时候，这个方法可以快速地缩小匹配范围，提高应用运行的性能。

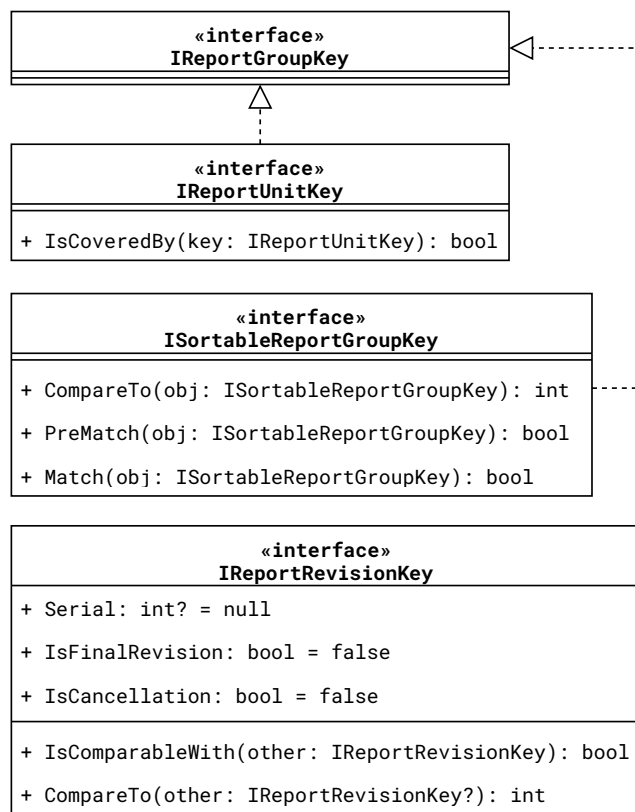


图 3.4 分组键、单位键和修订键

Figure 3.4 Group Key, Unit Key, and Revision Key

### 3.2.2 生成器

除了从原始事件对象生成事件对象，本系统中还存在大量需要从某类对象生成其它对象的逻辑。为了方便管理这些逻辑，我们把这些逻辑在系统中包装为一类接口，称之为“生成器”（英文：generator），如图 3.5。

有些生成器接受上下文输入，这些上下文是生成器生成时所需要的环境。

全面的本地化支持是本系统的特点之一，因此我们在所有生成方法都有一个以引用传入的区域性对象参数。传入时，该参数指定请求的输出对象的区域性。考虑到生成器不一定能支持请求的区域性，生成器返回结果时需要附带输出对象实际的区域性。

生成器的统一管理逻辑参见第 3.3.1 节。

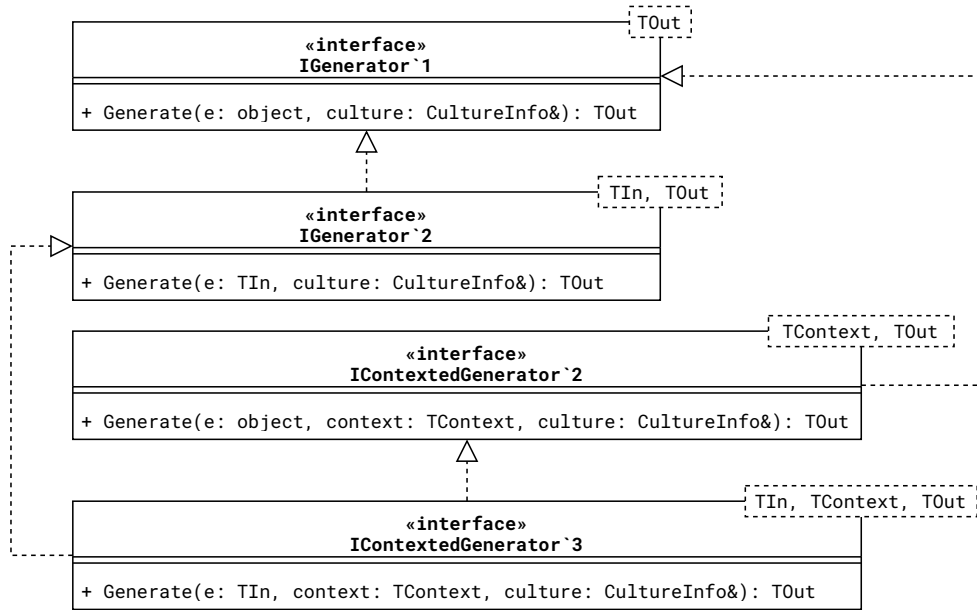


图 3.5 生成器

Figure 3.5 Generator

### 3.2.3 事件源

事件源接口是与事件源直接交互的接口，负责从事件源获取原始事件对象。此外事件源接口也需要报告自己与事件源连接的状态。相关的类型如图 3.6。

事件源的管理及其接收到的原始事件对象由核心层处理，参见第 3.3.3 节。

### 3.2.4 本地化资源

核心层在使用生成器生成各类对象时，会传入一个区域性参数，指定请求生成的区域性。为方便事件源层实现，接口层定义一些本地化资源类型，用于读取本地化资源，如图 3.7。

与生成器一样，本地化资源的一些方法里也有相同作用的区域性参数。

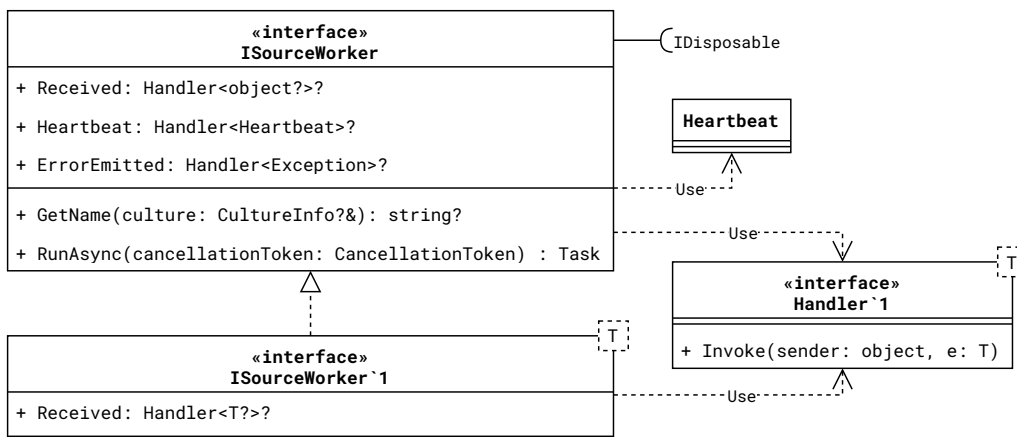


图 3.6 事件源接口

Figure 3.6 Event Source Interfaces

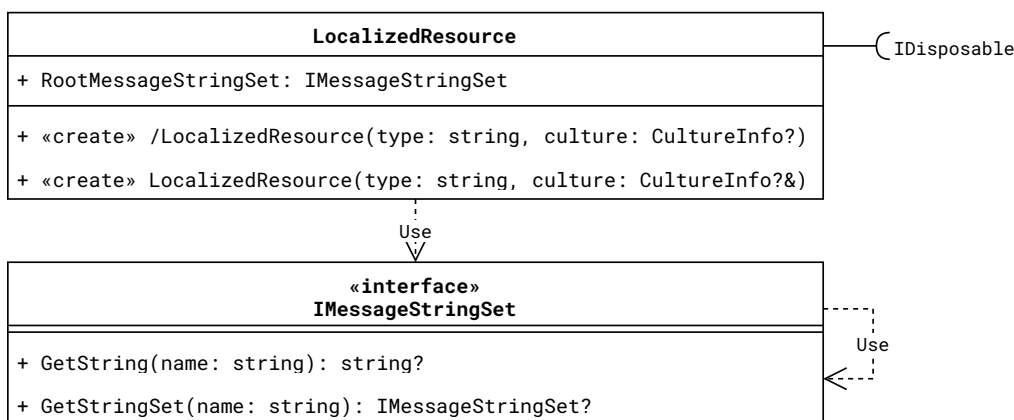


图 3.7 本地化资源

Figure 3.7 Localized Resource

### 3.2.5 语音接口

语音播报为本系统的重要功能之一。我们将每个事件需要播报的语音内容抽象为一个语音条目（英文：TTS entry），包含播报语言、标题、正文、优先级等信息，如图 3.8。

语音条目的处理参见第 4.4 节。

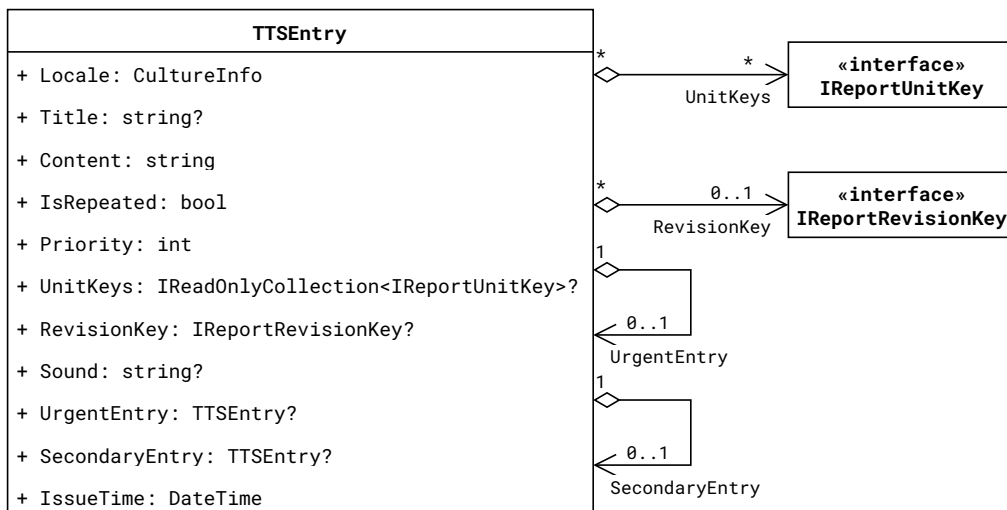


图 3.8 语音条目对象类

Figure 3.8 TTS Entry Object Class

### 3.2.6 地图接口

地图呈现也是本系统的重要功能之一。地图接口如图 3.9。

定义的地图元素如下。

- 组元素：组合一系列子元素的地图元素。
- 遮罩组元素：被指定的若干多边形遮罩的组元素。
- 生效组元素：用来表示生效中事件的组元素。
- 震源元素：表示震源的地图元素。
- 标签点元素：带文字标签的表示一点的地图元素。
- 通用点元素：不适合用其它类型表示的表示一点的地图元素。
- 若干线元素：表示若干线的地图元素。
- 若干多边形元素：表示若干多边形的地图元素。
- 海啸高度元素：表示海啸高度的地图元素。
- 地震波元素：表示地震波的地图元素。

这些地图元素对象生成后交由应用层呈现在交互界面上。

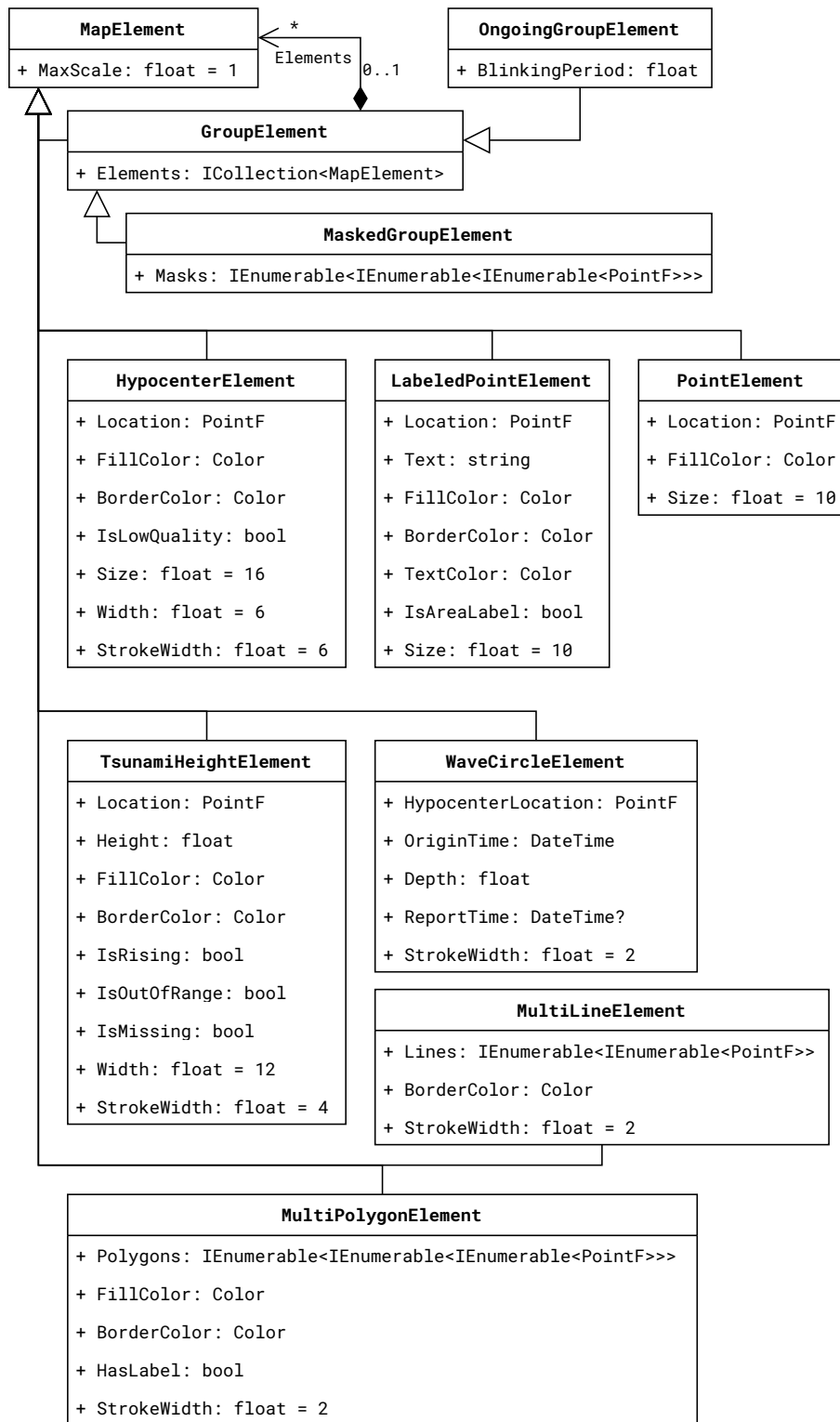


图 3.9 地图元素

Figure 3.9 Map Elements

### 3.3 核心层概要

本节描述核心层中定义的类型。

#### 3.3.1 生成器管理器

同一输出类型的生成器由一个生成器管理器（英文：generator manager）集中管理。上层逻辑先在生成器管理器中注册已知的生成器，需要调用生成器时直接传入输入对象，由生成器管理器负责判断输入对象的类型，并调用合适的生成器。相关类型如图 3.10。

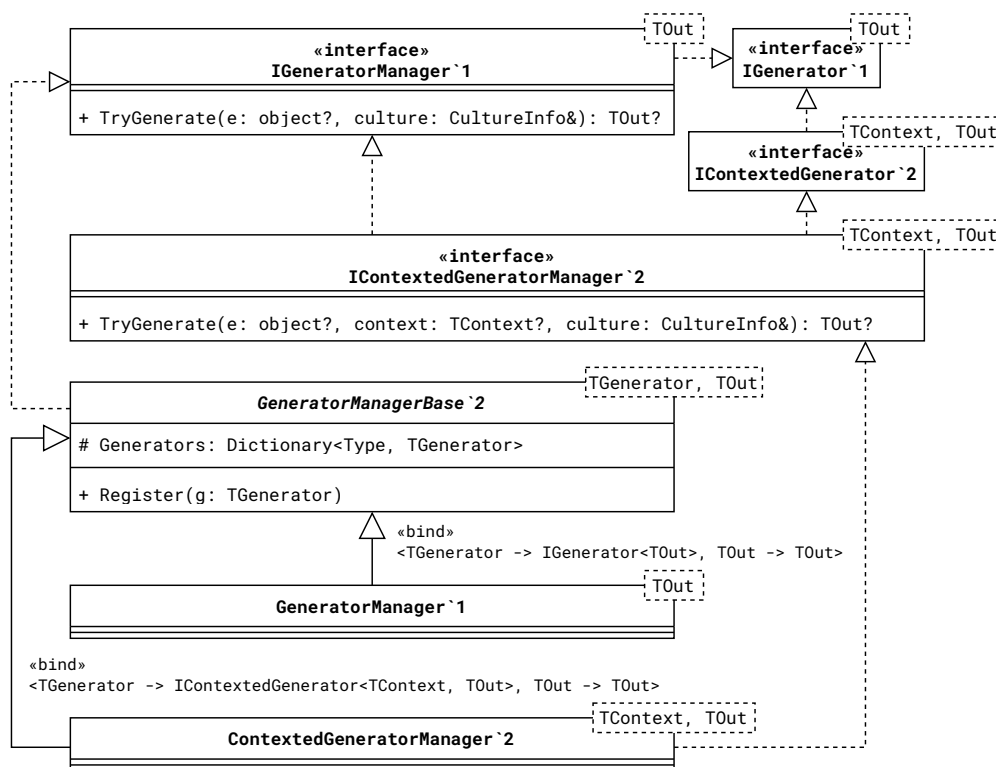


图 3.10 生成器管理器

Figure 3.10 Generator Manager

#### 3.3.2 语音和音效

语音条目由语音工作器（英文：TTS worker）处理并送入语音引擎。附带音效的语音条目，其音效的播放由声音播放器（英文：sound player）处理。相关类型如图 3.11。

语音条目队列的处理逻辑详细参见第 4.4 节。

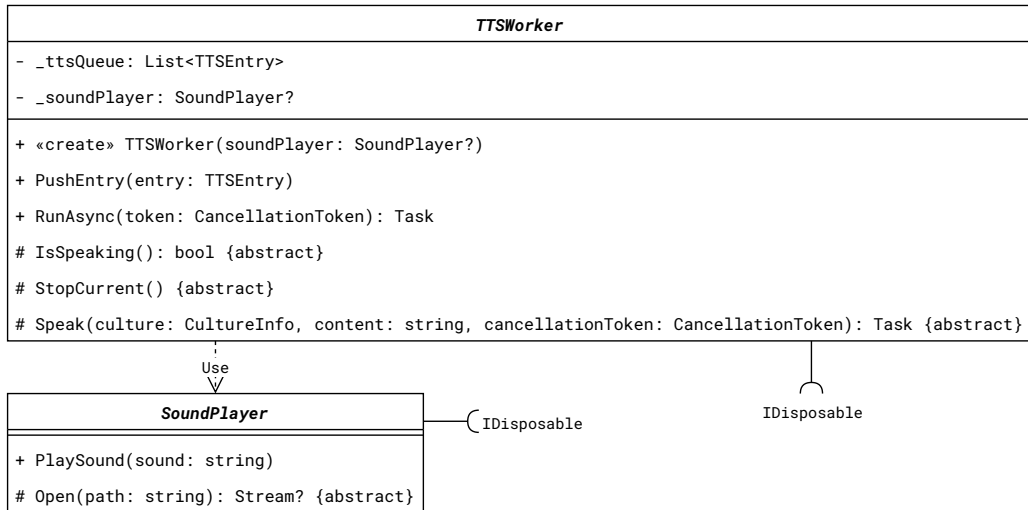


图 3.11 语音工作器和声音播放器

Figure 3.11 TTS Worker and Sound Player

### 3.3.3 核心工作器

核心工作器（英文：core worker）负责整个系统中最核心的逻辑，包括事件源的管理、事件单位和事件分组以及事件对象和语音条目的生成与下发。相关类型如图 3.12。

事件对象和语音条目的生成器管理器在核心工作器中，而地图元素的生成器管理器在各应用层中。

核心工作器的逻辑和分组逻辑详细参见第 4.4 节。

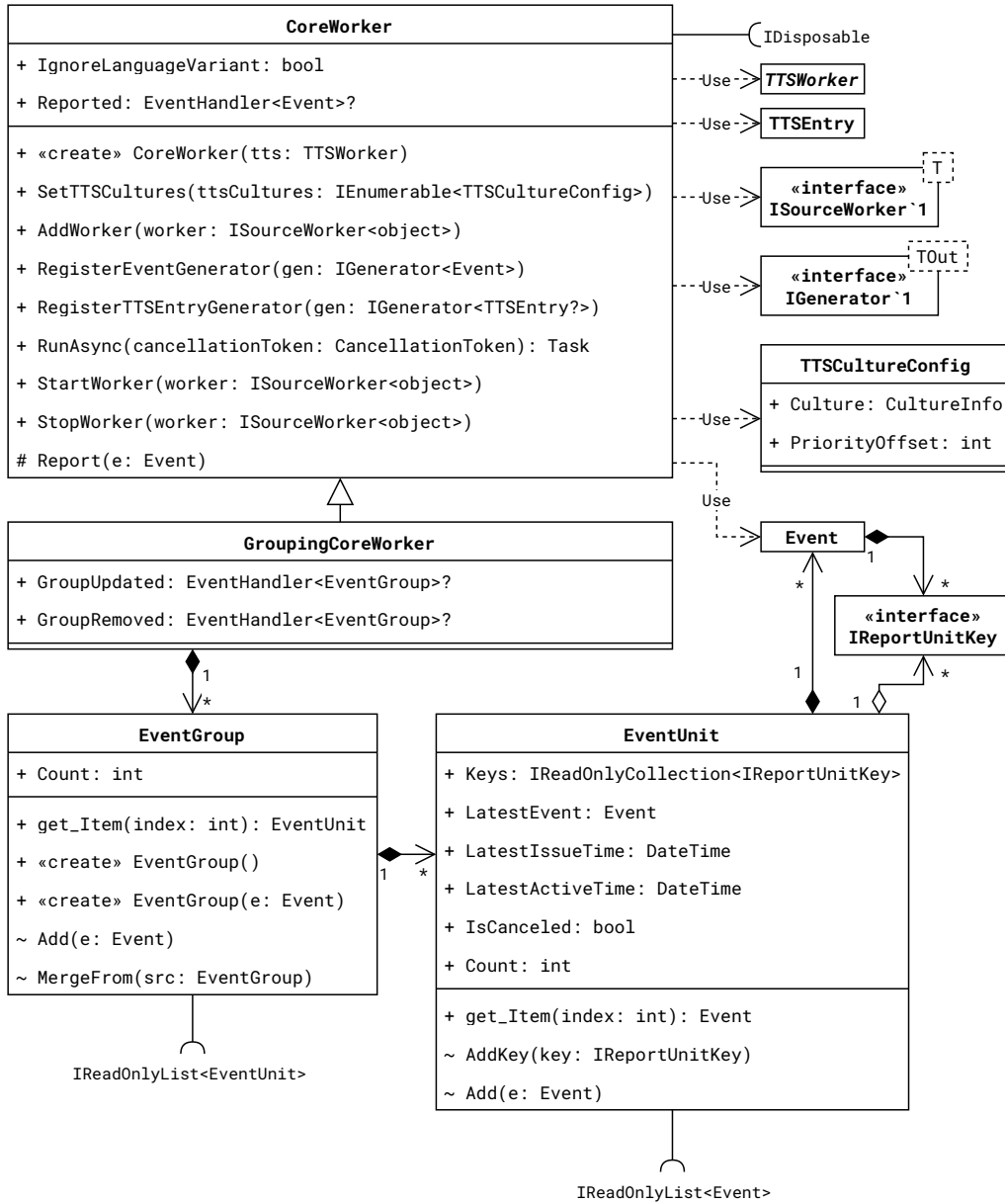


图 3.12 核心工作器

Figure 3.12 Core Worker

### 3.3.4 其它辅助类型

应用中经常出现需要计算地震波走时的场景,如绘制地震波二圆时需要计算地震波在某一时刻与震中的距离。TauP 工具集可以很方便地实现这个功能,通过输入地震参数和地震经过时间即可得到地震波震中距离。<sup>[3]</sup>不过由于这些计算性能要求较高,直接在运行时调用该工具集并不现实。为此,可以先取一些关键参数进行采样,预先计算走时并生成一张走时表,运行时取参数相近的采样点进行插值即可。

## 3.4 本章小结

本章描述了系统的整体架构,设计了接口层中的事件、生成器、事件源、语音、地图等关键接口,以及核心层中生成器管理器、语音工作器和核心工作器等核心组件的大致功能及相互之间的关系。

## 第 4 章 详细设计

### 4.1 组件结构

本系统使用的所有中间件如下。

- Cryville.Audio: 实时音频渲染引擎。
- Cryville.Common Buffers: 通用缓存库。
- Cryville.Common.Compat: 通用框架兼容层。
- Cryville.Interop.Java.ObjectStream: Java 对象流的 .NET 实现。
- NVorbis: 轻量 Vorbis 音频编码解码库。
- System.Text.Json: JSON 序列化库。

系统内所有组件的版本号格式遵循语义化版本 2.0.0<sup>[4]</sup>。

### 4.2 视觉设计

#### 4.2.1 配色方案

符合直觉且统一的配色有助于用户更迅速地感知到信息，以下是专门为本应用设计的默认配色方案。

为了符合人类的视觉感知，整套配色方案在 Oklab 色彩空间<sup>[5]</sup>下进行设计。Oklab 是一个在人类感知上统一的色彩空间，能较好地反应人类真正感知到的颜色的属性。由于所有色彩最终需要显示在屏幕上，而市面上大部分屏幕的色彩空间均为 sRGB，故额外要求该配色方案的颜色尽量在 sRGB 色彩空间范围内。

为了配色的统一性，我们将需要赋予颜色的属性先映射到一个一维的值，称该值为“严重度”（英文：severity）。随后将严重度映射到一个颜色。由于 Oklab 色彩空间感知上统一，我们自然也基于人类对不同属性的感知程度来规定严重度的基准含义。规定严重度的基准及对应默认配色的要求如表 4.1。

对于与感知没有直接关系的属性，我们基于其典型情况下的感知情况设计其严重度映射。

表 4.1 严重度基准含义和默认配色

Table 4.1 Semantics and Default Colors for Severity Values

严重度	基准含义	配色要求
0.00	仪器能够检测到、但是人类无法或一般无法感知。	蓝色色调, 亮度较低
0.50	部分人开始有感知。	绿色色调, 亮度中等
0.75	大部分人有明显的感知。	黄色色调, 亮度较高
1.00	所有人有强烈的感知, 感到恐慌。	红色色调, 亮度及色值较高
1.25	所有人有剧烈的感知, 能够造成显著的灾难。	紫色色调, 亮度较低, 色值较高

基于上述要求, 最终设计如下。

首先, 将属性映射到严重度。根据属性本身的性质选择合适的函数模型, 通过取关键点拟合或直接调整参数得到映射函数。严重度  $S$  对于各属性的函数如表 4.2。

表 4.2 默认严重度函数

Table 4.2 Default Severity Functions

属性	变量	严重度函数
地震震级	震级 $m$	$S(m) = 1.75 - 2.10 \times 0.84^m$
震源深度	深度 $d$ km	$S(d) = \exp(-(0.005d)^{1.2})$
海啸高度	高度 $h$ m	$S(h) = \min(1.50, h^{0.2})$
火山口上方喷烟高度	高度 $h$ km	$S(h) = 1.50 - 1.50 \times 0.80^h$
MMI <sup>1</sup> 烈度	烈度 $i$	$S(i) = 0.74 \cdot \frac{i-1.5}{8} + 0.5$
CSIS <sup>2</sup> 烈度	烈度 $i$	$S(i) = 0.74 \cdot \frac{0.81i+0.45-1.5}{8} + 0.5$

JMA <sup>3</sup> 地震烈度表烈度 (日文: 震度) CWASIS <sup>4</sup> 烈度 (繁体中文: 震度)	烈度 $i$	$S(i) = \begin{cases} 8/16 & \text{if } i = \text{"1"}, \\ 10/16 & \text{if } i = \text{"2"}, \\ 12/16 & \text{if } i = \text{"3"}, \\ 14/16 & \text{if } i = \text{"4"}, \\ 16/16 & \text{if } i = \text{"5-"}, \\ 17/16 & \text{if } i = \text{"5+"}, \\ 18/16 & \text{if } i = \text{"6-"}, \\ 19/16 & \text{if } i = \text{"6+"}, \\ 20/16 & \text{if } i = \text{"7"}, \\ -1 & \text{otherwise.} \end{cases}$
JMA <sup>3</sup> 地震烈度表仪器烈度 (日文: 計測震度) CWASIS <sup>4</sup> 仪器烈度 (繁体中文: 計測震度)	仪器烈度 $i$	$S(i) = \frac{i+3}{8}$
CWASIS <sup>4</sup> 合成 PGA <sup>5</sup> 和合成 PGV <sup>6</sup>	合成 PGA $a$ gal 合成 PGV $v$ kine	$S(a, v) = \begin{cases} 0.11 \cdot \ln a + 0.52 & \text{if } a < 80, \\ 1 & \text{if } v < 15, \\ 0.11 \cdot \ln v + 0.70 & \text{otherwise.} \end{cases}$
JMA <sup>3</sup> 长周期地震动阶 级 (日文: 長周期地震 動階級)	长周期地震动阶 级 $i$	$S(i) = \begin{cases} 0.80 & \text{if } i = \text{"1"}, \\ 0.95 & \text{if } i = \text{"2"}, \\ 1.10 & \text{if } i = \text{"3"}, \\ 1.25 & \text{if } i = \text{"4"}, \\ -1 & \text{otherwise.} \end{cases}$
绝对速度响应频谱最大 值 (日文: 絶対速度応 答スペクトル最大値)	绝对速度响应频谱 最大值 $s$ m s <sup>-1</sup>	$S(s) = 0.56 + 0.15 \cdot \ln s$

<sup>1</sup>修订麦加利地震烈度<sup>2</sup>中国地震烈度表<sup>3</sup>日本气象厅 (日文: 気象庁)<sup>4</sup>台湾交通部中央气象署地震烈度分级 (繁体中文: 交通部中央氣象署地震震度分級)<sup>5</sup>峰值地表加速度 (繁体中文: 峰值地表加速度)<sup>6</sup>峰值地表速度 (繁体中文: 峰值地表速度)

将严重度  $S$  映射到 Oklch 色彩空间的三个分量。

$$\begin{aligned}
 L &= \begin{cases} 0.52 & \text{if } S < 0, \\ -0.03(8S) - 0.09 \cdot |8S - 6| + 1.06 & \text{otherwise.} \end{cases} \\
 C &= \begin{cases} 0.1853(1 + S) & \text{if } S < 0, \\ -0.00112(8S)^3 + 0.0207(8S)^2 - 0.0841(8S) + 0.1853 & \text{otherwise.} \end{cases} \\
 h &= \begin{cases} 270 & \text{if } S < 0, \\ 270 - 30(8S) & \text{otherwise.} \end{cases}
 \end{aligned} \tag{4.1}$$

最后根据 Oklab 标准将 Oklch 颜色分量近似转换为 sRGB 颜色分量即可。

另外, 为了保证前景与背景颜色的对比度, 当表示严重度的颜色作为背景时, 前景文字的颜色取决于该背景颜色的亮度  $L$ 。当  $L^2 < \frac{1}{2}$ , 前景文字颜色为白色, 否则为黑色。

最终效果可参考附录 A。

#### 4.2.2 地图设计

本应用中, 地理元素的呈现方式是使用 Web 墨卡托投影的二维地图。这种投影方式的地图支持广泛, 便于展示不同事件源的各种事件以及导入各种地理信息。考虑到地质灾害的地图呈现需要世界地形作为参考, 地图的底图使用 Esri 提供的 World Ocean Base (世界海底地形图)<sup>1</sup> 叠加 World Hillshade (世界山体阴影)<sup>2</sup>, 如图 4.1。由于本应用面向全球用户, 底图中有意不含有任何的行政边界, 以免引起分歧。

各地图元素的外观规定如下。

- 震源元素: 带白色描边的红色“X”状图形, 低精度震源为环形。
- 标签点元素: 带描边的圆形或圆角矩形, 中间附带文字。

<sup>1</sup><https://www.arcgis.com/home/item.html?id=1e126e7520f9466c9ca28b8f28b5e500>

© Esri, GEBCO, NOAA, Garmin, and other contributors

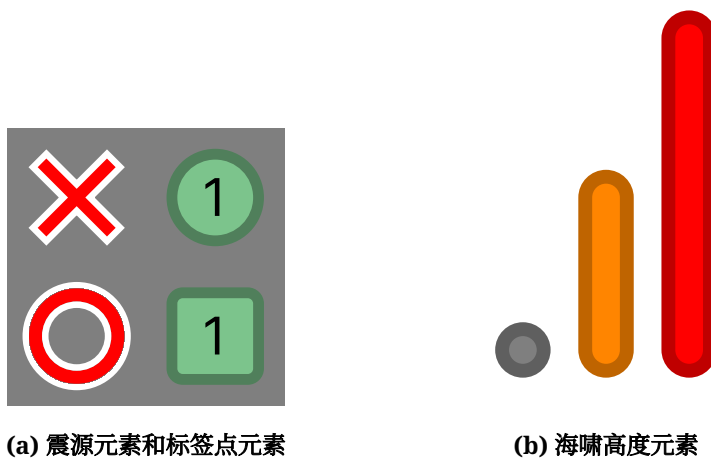
<sup>2</sup><https://www.arcgis.com/home/item.html?id=1b243539f4514b6ba35e7d995890db1d> © Esri, Maxar, Airbus DS, USGS, NGA, NASA, CGIAR, N Robinson, NCEAS, NLS, OS, NMA, Geodatastyrelsen, Rijkswaterstaat, GSA, Geoland, FEMA, Intermap, and the GIS user community



图 4.1 底图样本

Figure 4.1 Basemap Sample

- 通用点元素：仅填充的圆形。
- 海啸高度元素：带描边的棒状图形。
- 地震波元素：仅描边的圆形。



(a) 震源元素和标签点元素

(b) 海啸高度元素

图 4.2 地图元素

Figure 4.2 Map Elements

### 4.3 事件源层简述

本节简要说明各事件源的接口以及事件源层的实现方式。

我们将所有事件源分为两个类型：拉取式和推送式。拉取式指每次需要更新事件时，由客户端主动发出请求，服务器被动响应。这类型的事件源及其 API 接口等信息如表 4.3。

表 4.3 拉取式事件源

Table 4.3 Pulling Event Sources

名称	示例接口 URI	数据格式
日本气象厅地震火山相关防灾信息 Atom 推送	<a href="https://www.data.jma.go.jp/developer/xml/feed/eqvol.xml">https://www.data.jma.go.jp/developer/xml/feed/eqvol.xml</a>	Atom <sup>[6]</sup> XML <sup>[7]</sup>
台湾中央气象署气象资料开放平台	<a href="https://opendata.cwa.gov.tw/api/v1/rest/datastore/E-A0014-001">https://opendata.cwa.gov.tw/api/v1/rest/datastore/E-A0014-001</a> <sup>3</sup>	JSON
美国地质调查局实时 QuakeML 推送	<a href="https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/significant_week.quakeml">https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/significant_week.quakeml</a>	QuakeML <sup>[8]</sup>
美国海啸警报中心	<a href="https://www.tsunami.gov/events/xml/PAAQAtom.xml">https://www.tsunami.gov/events/xml/PAAQAtom.xml</a>	Atom <sup>[6]</sup>
印度尼西亚气象、气候及地球物理局开放地震数据	<a href="https://data.bmkg.go.id/DataMKG/TEWS/autogempa.json">https://data.bmkg.go.id/DataMKG/TEWS/autogempa.json</a>	JSON

推送式指客户端发出连接请求，服务器与客户端建立连接后，在有事件更新时主动向客户端推送。这类型的事件源及其 API 接口等信息如表 4.4。

事件源层的各实现接收到事件后，将其反序列化为各自的原始事件对象，随后下发给核心工作器处理。

<sup>3</sup>台湾中央气象署气象资料需要授权码认证方可访问。

<sup>4</sup>GlobalQuakeServer 的连接为直接的 TCP 连接。

表 4.4 推送式事件源

Table 4.4 Pushing Event Sources

名称	示例接口 URI	数据格式
GlobalQuakeServer API	server-old.globalquake.net :38000 <sup>4</sup>	Java 对象流 <sup>[9]</sup>
欧洲与地中海地震中心 实时通知	wss://www.seismicportal.eu /standing_order/websocket	JSON

#### 4.4 核心层逻辑

核心工作器负责管理事件源工作器，从事件源工作器中接收到原始事件对象时，先利用事件对象的生成器管理器从原始事件对象生成事件对象，并将其下发至应用层。事件对象生成完后，再进行进一步的处理。

核心工作器在生成事件对象并将其下发至应用层后即开始生成语音条目。语音播报采用接入外部文本转语音引擎（英文全称：text-to-speech engine，以下简称“语音引擎”）的方式实现。本应用只负责播报文本的生成和各播报条目的顺序控制。对于用户指定的每一个播报语言，核心工作器利用语音条目的生成器管理器从原始事件对象生成语音条目，并将生成的所有语音条目通过连接成一个链表送入语音工作器。如图 4.3。

语音工作器收到语音条目后先将其音效送入音效播放器。考虑到有些事件源有时会频繁对一个事件单位发报，超出语音播报的速度，为了减少信息的冗余，系统会将要插入的语音条目与语音条目队列中已有的条目进行信息涵盖检查，删除冗余的条目。最后，只有新的条目没有被已有的条目涵盖时，才将其插入队列。如图 4.4。

语音条目队列由语音工作器在单独的后台线程中维护。其负责从队列中取出语音条目送入语音引擎。有时遇到优先级比当前正在播报的条目更高的条目时，语音工作器会打断当前正在播报的条目，并视情况将其重新插入队列。如图 4.5。

分组逻辑如图 4.6 和图 4.7。

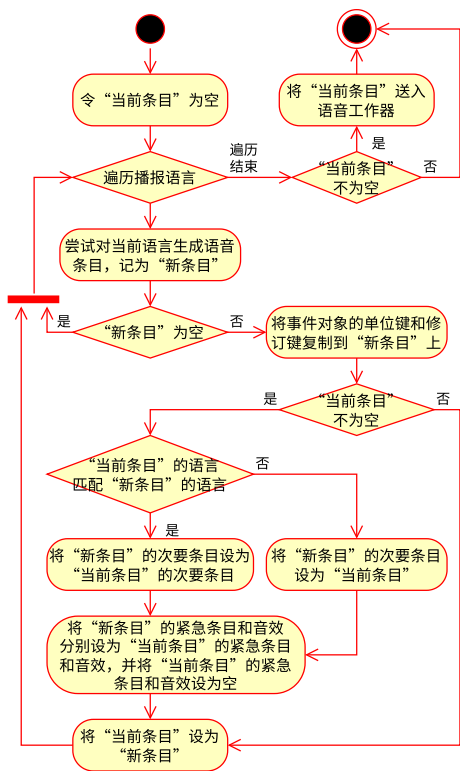


图 4.3 生成语音条目

Figure 4.3 TTS Entry Generation

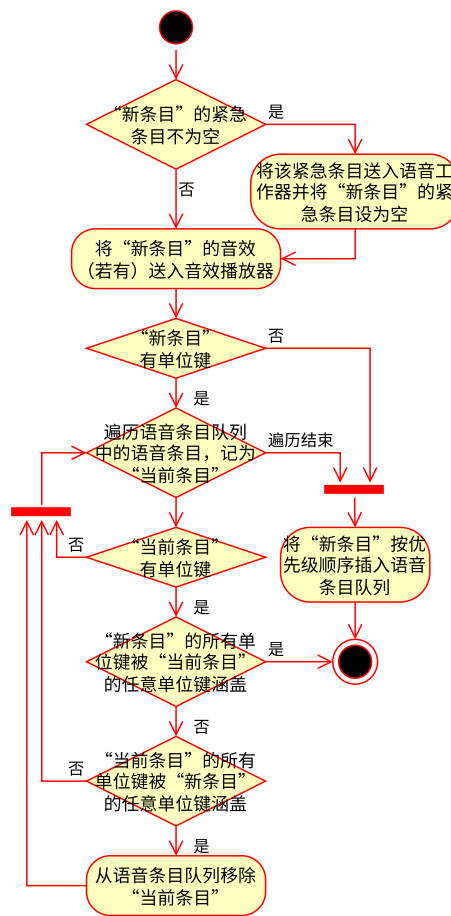


图 4.4 插入语音条目

Figure 4.4 Pushing TTS Entry

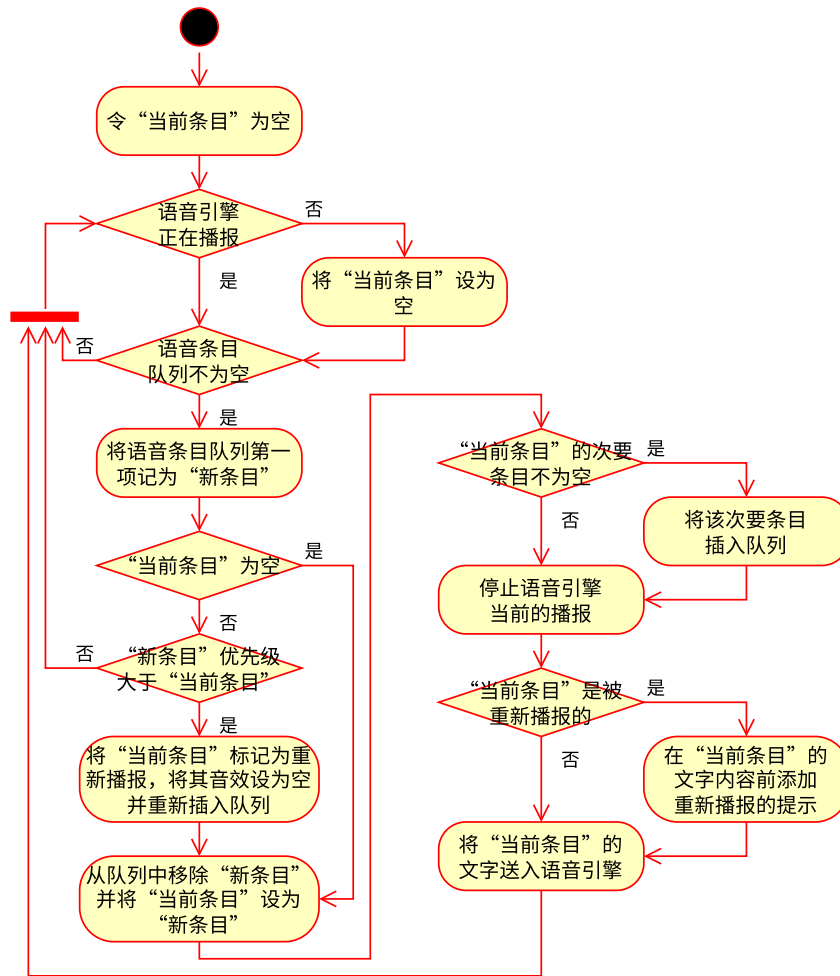


图 4.5 语音工作器

Figure 4.5 TTS Worker

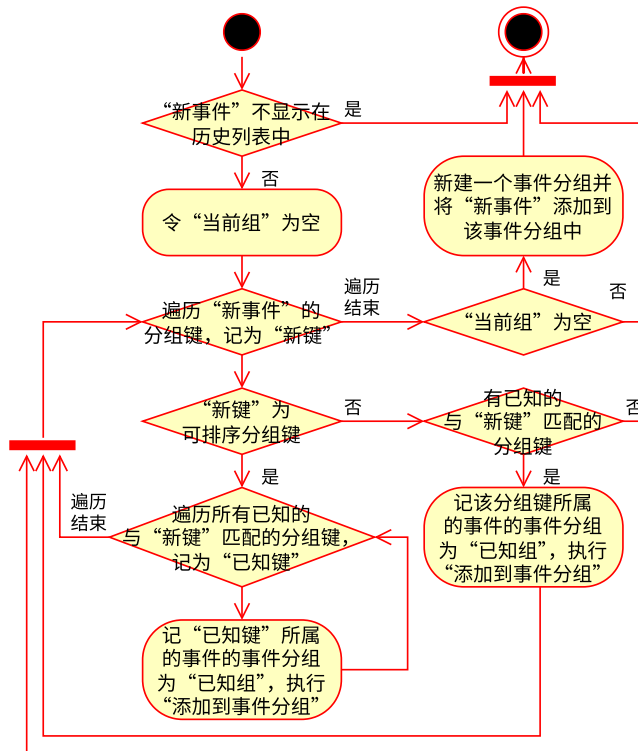


图 4.6 事件分组

Figure 4.6 Event Grouping

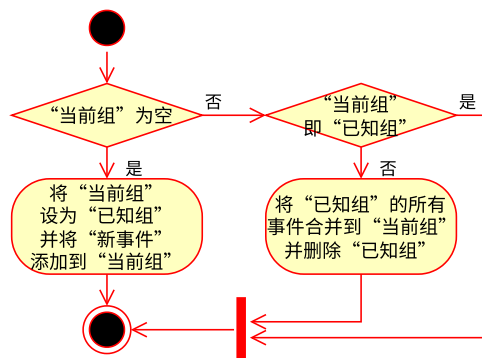


图 4.7 “添加到事件分组”

Figure 4.7 “Add Event to Group”

## 4.5 音效设计

本应用的预警功能和播报功能需要用到一系列的音效，用于在事件发生时先于语音引起用户的注意。对于不同性质的事件，需要专门对其音效进行设计。设计的音效如表 4.5。

其中，两个 `eew_local` 前缀的音效为循环若干次的、频率（音调）上滑的锯齿波（英文：saw wave）。锯齿波听感较为刺耳，容易引起警觉。`eew_local_4` 相较 `eew_local_2` 多一个复音，且音调上滑更快、循环间隔更短。

除上述两个音效，其余音效为使用颤音琴（英文：vibraphone）音色合成的乐音。`eew_update` 前缀的音效为单音或简单的音程。其它 `eew` 前缀的音效为以 A4（440Hz）为根音的减和弦琶音。减和弦是一种不和谐的和弦，能够更好地引起用户的警觉。`eq` 前缀的和 `ev` 音效为以 C4（约 261.6Hz）为根音的三和弦琶音，较为和谐，与其它音效区分度较大。各音效的具体乐谱参见附录 B。

此外，系统内还有两个硬编码的音频流片段。其中一个是中国国家地震烈度速报与预警工程项目地震预警信息发布规程中规定的地震预警警报声音<sup>[10]</sup>，由于没有国内地震预警信息源而暂未使用。另一个是日本邮政省规定的紧急警报信号<sup>[11][12]</sup>，在接收到日本的海啸警报或大海啸警报时触发。

## 4.6 本章小结

本章描述了系统的组件结构，设计了应用的配色方案、地图元素和音效，简要说明了事件源层的实现，并整理和详细设计了核心层的业务逻辑。

表 4.5 音效

Table 4.5 Sound Effects

名称	对应事件
eew_1	地震预警 严重程度从低到高，分 1 到 5 共五级音效
eew_2	
eew_3	
eew_4	
eew_5	
eew_local_2	用户所在地有震感的地震预警（有感）
eew_local_4	用户所在地有震感的地震预警（强有感）
eew_update	地震预警的更新报
eew_update_final	地震预警的最终更新报
eew_update_cancel	地震预警的取消报
eq	仅带震源信息的地震信息
eq_a	自动测定的仅带震源信息的地震信息
eq_i	仅带烈度信息的地震信息
eq_d	带烈度信息和震源信息的地震信息
eq_c	地震信息的取消报
ev	其它地震相关的信息

## 第 5 章 实现与测试

### 5.1 实现

本系统使用 Visual Studio 作为开发工具。应用层分双端实现：安卓端于安卓系统运行，使用 .NET for Android 框架进行开发，并启用 AOT 编译；Unity 端<sup>1</sup>于 Windows 系统运行，使用 Unity 引擎进行开发，编译时使用 IL2CPP 脚本后端。编程语言使用 C#，所有平台共用一份底层代码，底层框架为 .NET 8 和 .NET Standard 2.1。部分数据预处理使用 Python 脚本进行。版本管理工具使用 git。

安卓端对安卓系统的运行环境要求如下。

- 安卓 API 版本：21 或以上（安卓 5.0 或以上）；
- CPU 架构：ARMv7 或 ARM64；
- 不支持在容器或模拟器中运行。

Unity 端对 Windows 系统的运行环境要求如下。

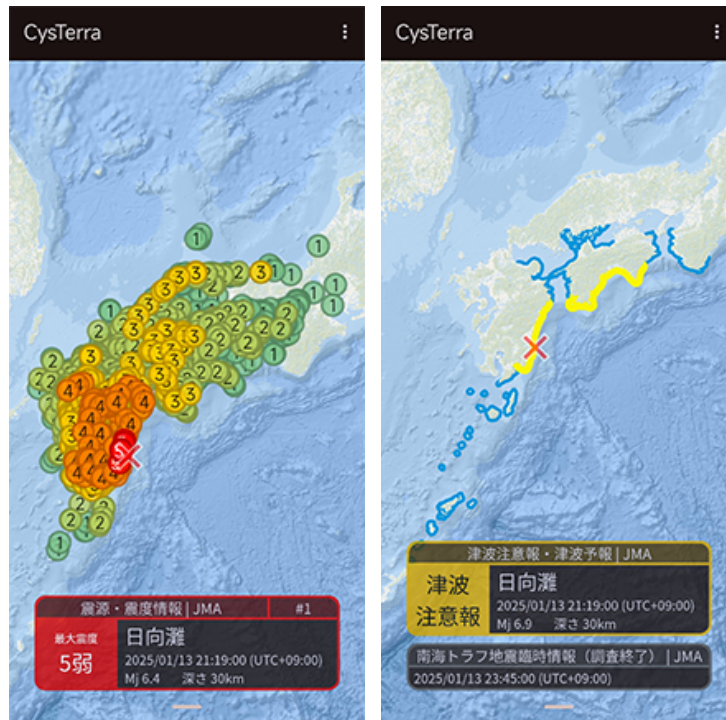
- 系统版本：Windows 7（SP1 或以上）、Windows 10 或 Windows 11；
- CPU 架构：x86 或 x64；
- 图形 API：支持 DX10、DX11 或 DX12；
- 不支持在容器或模拟器中运行。

应用取名为“CysTerra”，发音为 /saɪs'tɛɪə/ 或 /saɪstɛɪə/。本名字来源于“Cys”+“Terra”：“Cys”来源于英语“seism”（“地震”）；“Terra”来源于拉丁语“terra”（“土地”）。

截至本文定稿，应用的用户界面如图 5.1 和图 5.2 所示。

---

<sup>1</sup>Unity 端在工具链层面上并不仅限于能在 Windows 系统下运行，因此该端不称作“PC 端”“Windows 端”或其它类似的名称。



(a) 地震信息

(b) 海啸信息



(c) 事件历史

(d) 应用设置

图 5.1 安卓端应用界面

Figure 5.1 User Interface of the Android Build

截图中启用了“使用事件源语言”设置，所有信息以事件源语言显示

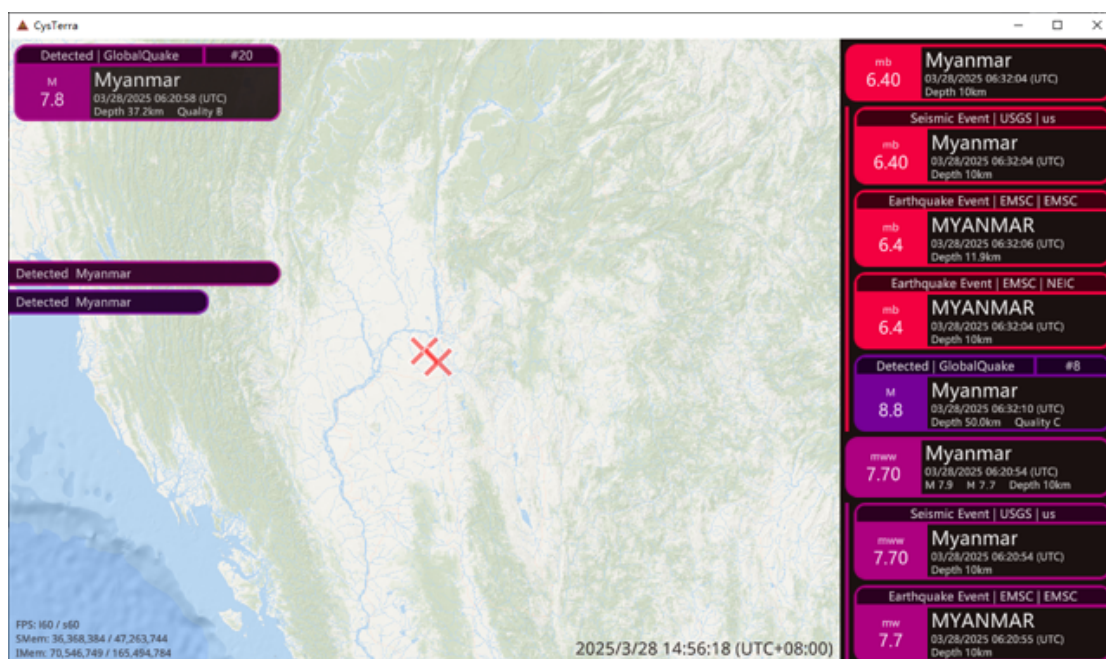


图 5.2 Unity 端应用界面

Figure 5.2 User Interface of the Unity Build

截图中启用了“使用事件源语言”设置，所有信息以事件源语言显示

## 5.2 代码分析

在编码应用的过程中，有一些问题可以通过分析代码在编译之前就及时发现，减少后续测试成本。 .NET 的开发工具使用 Roslyn 编译平台进行代码编译。Roslyn 是 C# 和 Visual Basic 编译器的开源实现，附带构建代码分析工具的 API。 .NET 本身内置了很多代码分析器，能够针对相对全面的应用场景对各种代码问题提出建议和警报，但由于 .NET 框架本身有些逻辑存在缺陷，抑或是某些代码分析器本身存在问题，有时可能需要开发者自行开发几个代码分析器。

例如，在实现本应用的过程中，编者发现，在安卓平台上，对于某些特定的区域性， .NET 框架给出的时间格式是不带上下午标识的十二小时制时间，这显然会引起重大歧义。<sup>2</sup>针对该问题，我们在接口层添加一个与 .NET 框架区域性接口等价的接口，并对其时间格式予以修复。此时就有必要开发一个相关的代码分析器，防止将来开发过程中无意间直接使用 .NET 框架本身的区域性接口，从源头上阻断问题的发生。

<sup>2</sup>编者已提交该问题和相关拉取请求到 .NET 框架代码仓库，修复代码已于 2024 年 6 月 22 日合并。

<https://github.com/dotnet/runtime/issues/103592>

<https://github.com/dotnet/runtime/pull/103681>

### 5.3 测试

本应用的测试分为单元测试、集成测试和系统测试三个步骤。

单元测试使用 NUnit, 一个 .NET 语言专用的单元测试框架, 对应用的各个模块的各业务逻辑编写单元测试。图 5.3 和图 5.4 展示了部分单元测试。

Test	Duration
▲ Cryville.EEW.Analyzer.Test (4)	2.7 sec
▲ Cryville.EEW.Analyzer.Test (4)	2.7 sec
▲ SpecifyFormatProviderTest (2)	2.6 sec
● Test	2.6 sec
● TestExcluded	47 ms
▲ UseSharedCulturesTest (2)	52 ms
● Test	41 ms
● TestExcluded	11 ms

图 5.3 代码分析器部分单元测试

Figure 5.3 Part of the Unit Tests of the Analyzers

Test	Duration
▲ Cryville.EEW.Test (16)	30.3 sec
▲ Cryville.EEW.Test (16)	30.3 sec
▲ AudioTests (4)	27.7 sec
● CENCWarningTest	2.5 sec
● GenericTsunamiSignalTest	9 sec
● JapanTsunamiBeginSignalTest	8.1 sec
● JapanTsunamiEndSignalTest	8 sec
▲ Tests (11)	1.6 sec
● GlobalQuakeWorkerInvalidatedTimeTest	120 ms
● TestAtomDeserialization	124 ms
● TestCoordinateParsing	6 ms
● TestFeRegions	16 ms
● TestGeoJSONDeserialization	73 ms
● TestJmaEqListDeserialization	24 ms
● TestJmaForecastArea	74 ms
● TestJmaReportDeserialization	1.2 sec
● TestJmaTsunamiArea	39 ms
● TestLocalizedResource	4 ms
● TestLocalizedResourceParent	< 1 ms

图 5.4 核心层部分单元测试

Figure 5.4 Part of the Unit Tests of the Core Layer

其中, 事件源层的测试用例主要从三个途径获取: (1) 事件源文档有样例的, 直接将其作为测试用例; (2) 事件源文档没有样例的, 直接从其接口获取具有代表性的以及处于各种边缘情形下的事件; (3) 事件源的事件中找不到某种边缘情形下的事件的, 手动编写测试用的事件。

集成测试分为两个阶段, 首先将接口层和核心层组装, 针对其之间的交互进行测试。该测试通过后再加入事件源层, 针对事件源层, 测试其在接口层和核心层的表现是否正确。

系统测试即直接对构建出来的应用以用户的角度使用。除了开发人员自己进行测试，本应用开发过程中还发布了若干 Alpha 版本和 Beta 版本，供用户在实际环境下公开测试。考虑到用户不一定会进行反馈或反馈时未必能提供足够详细的信息，测试版本发布前应用层中加入了一些错误报告逻辑，遇到未处理的异常时将异常信息和系统信息发送至报告服务器储存，供开发人员调查。本应用的版本历史参见附录 C。

经测试，应用中已完成的功能运行良好，能够在事件发生时迅速、准确、详细地呈现信息，符合需求分析中的功能性需求和性能需求。

#### 5.4 本章小结

本章描述了系统在实现上的一些细节和测试方法，同时介绍了代码分析器的使用。



## 第 6 章 总结

本文对一个多事件源融合的地质灾害信息应用展开设计。经过需求分析、概要设计、详细设计，最终实现了一个可用性和稳定性较高的应用。

在需求分析中，本文分析了应用的目标用户群体，并根据其用例设计了应用具体的需求。在系统概要中，本文描述了系统分为接口层、事件源层、核心层和应用层四层的宏观架构，辅以类图，具体设计了接口层中事件、生成器、事件源、本地化资源、语音接口、地图接口，以及核心层中生成器管理器、语音音效、核心工作器等各个组件及其之间的交互关系，明确了各个组件的对象模型。在详细设计中，本文具体设计了应用的配色、地图和音效，并着重描述了核心层核心工作器和语音工作器的业务逻辑。在实现与测试中，本文说明了应用实现使用的开发工具、应用在各平台的运行环境要求以及系统的测试方法，展示了应用实际运行时的表现并对其进行了评估。

在开发环境的测试和实际环境的使用下，应用中已完成的功能运行良好，能够在事件发生时迅速、准确、详细地呈现信息，较好地达成了本文的研究目的。截至本文定稿，本应用的最新版本为 0.5.7。目前开发仍处在主版本 0，即测试版阶段。今后本应用未实现功能的开发仍会继续推进，并以逐渐面向一般大众为目标。



## 参考文献

- [1] Geller R J. Earthquake prediction: a critical review [J/OL]. *Geophysical Journal International*, 1997, 131(3): 425-450. <https://doi.org/10.1111/j.1365-246X.1997.tb06588.x>.
- [2] Španger J. GPU acceleration of GlobalQuake application [J/OL]. 2024. <https://is.muni.cz/th/md7hy/>.
- [3] Crotwell H P, Owens T J, Ritsema J. The TauP Toolkit: Flexible Seismic Travel-time and Ray-path Utilities [J/OL]. *Seismological Research Letters*, 1999, 70(2): 154-160. <https://doi.org/10.1785/gssrl.70.2.154>.
- [4] Preston-Werner T. Semantic Versioning 2.0.0 [S/OL]. 2013. <https://semver.org/spec/v2.0.0.html>.
- [5] Ottosson B. A perceptual color space for image processing [J/OL]. 2020. <https://bottosson.github.io/posts/oklab/>.
- [6] Nottingham M, Sayre R. Request for comments: number 4287 The Atom Syndication Format [M/OL]. RFC Editor, 2005. <https://www.rfc-editor.org/info/rfc4287>. DOI: 10.17487/RFC4287.
- [7] 気象庁. 気象庁防災情報 XML フォーマット 技術資料 [EB/OL]. [https://xml.kishou.go.jp/tec\\_material.html](https://xml.kishou.go.jp/tec_material.html).
- [8] Euchner F, Schorlemmer D. QuakeML—An XML Representation of Seismological Data Basic Event Description [EB/OL]. 2013. <https://quake.ethz.ch/quakeml/Documents>.
- [9] Oracle. Object serialization stream protocol [M/OL]//Java Object Serialization Specification. 2010. <https://docs.oracle.com/javase/8/docs/platform/serialization/spec/protocol.html>.
- [10] 中国地震台网中心. 地震预警信息发布规程 [EB/OL]. 2023. [https://www.cenc.ac.cn/cenc/\\_300647/407193/index.html](https://www.cenc.ac.cn/cenc/_300647/407193/index.html).
- [11] 電波監理委員会. 昭和二十五年電波監理委員会規則: number 18 無線設備規則 [EB/OL]. 1950. <https://laws.e-gov.go.jp/law/325M5008000018>.
- [12] 郵政省. 昭和六十年郵政省告示: number 405 無線設備規則第九條の三第五号の規定に基づく緊急警報信号の構成 [EB/OL]. 1985. [https://www.tele.soumu.go.jp/horei/law\\_honbun/72103000.html](https://www.tele.soumu.go.jp/horei/law_honbun/72103000.html).



## 附录 A 默认配色参考

### 基准



### 震度 / 震度



### 計測震度 / 計測震度



### 長周期地震動階級



### 絶対速度応答スペクトル最大値



### 烈度



### 震源の深さ / 震源深度 / 震源深度 (km)



### マグニチュード / 震級 / 規模



### 津波の高さ / 海啸高度 / 海啸高度 (m)



### 噴煙高度 (km)



图 A.1 默认配色

Figure A.1 Default Color Scheme

需彩色印刷 纸面呈现效果会与屏幕有一定差异







## 附录 C 版本历史

本附录仅列出主要更改和重大更改。

Unity 端发布后（0.4.7 及后续版本），标注的版本从安卓端版本转为发布周期版本。发布周期版本没有“Beta”标识不代表应用已经脱离 Beta 开发阶段。

版本	发布日期	更改
0.5.7	2025-06-16	实装 USGS GeoJSON 推送（详细推送）事件源。
0.5.6	2025-06-06	优化网络连接的流量消耗。
0.5.5	2025-05-11	实装 GeoNet 事件源。
0.5.4	2025-05-07	（Unity）实装语音播报。
0.5.3	2025-04-29	添加 JMA 海啸警报结束信号。
0.5.2	2025-04-08	（安卓）修复因不信任的证书导致 CWA 事件源连接失败的问题。
0.5.1	2025-04-08	实装显示信息本地化。
0.5.0	2025-03-23	修复在区域性不使用点（.）作为小数点的系统上，部分数字解析不正确的问题。
0.4.8	2025-02-26	添加 BMKG 开放数据事件源。 添加 EMSC 实时通知事件源。 添加 USGS QuakeML 推送事件源。
0.4.7	2025-02-22	（Unity）首个发布版本。
Beta 0.4.6	2025-02-04	修复播放音效时应用有小概率崩溃的问题。
Beta 0.4.5	2025-02-04	添加音效和语音的音频频道的设置。 添加过滤延迟事件的设置。
Beta 0.4.4	2025-01-10	增补 JMA 报文细分开关。 添加 JMA 南海海槽地震相关信息的语音播报支持。 添加 JMA 火山喷发速报的语音播报支持。 添加 JMA 地震次数信息的语音播报支持。
Beta 0.4.3	2025-01-01	修复自动切换至历史启用时，应用启动后会崩溃的问题。

版本	发布日期	更改
Beta 0.4.2	2025-01-01	<p>添加 NOAA 事件源, 包含 NTWC 海啸警报和 PTWC 海啸警报。</p> <p>修复只有取消报的事件单元会导致崩溃的问题。</p> <p>添加自动切换至历史的设置。</p>
Beta 0.4.1	2024-12-23	<p>修复地图自动聚焦跨越 180° 经线的元素时会错误地聚焦到 0° 经线附近的问题。</p> <p>修复触发额外音效时有小概率发生错误, 使事件源连接异常退出的问题。</p> <p>修复生成 JMA 降灰预报的地图时崩溃的问题。</p>
Beta 0.4.0	2024-12-02	<p>实装事件分组。</p> <p>实装震源深度的颜色主题。</p> <p>添加事件属性的附加状态显示。</p> <p>添加对 JMA 警报类固定附加文的语音播报支持。</p> <p>现在查看过往的地震预警时, 地图上会使用虚线绘制发报时刻的地震波二圆。</p> <p>添加使用兼容地图渲染的设置。</p> <p>添加字体大小设置。</p>
Beta 0.3.3	2024-10-20	<p>优化语音条目之间的覆盖和优先级逻辑。</p> <p>修复粤语翻译完全丢失的问题。</p> <p>添加 JMA 火山相关报文的具体现象显示。</p> <p>添加 JMA 火山相关报文的警戒种类显示。</p> <p>添加 JMA 火山警戒种类的语音支持。</p>
Beta 0.3.2	2024-09-27	<p>调整 CWA 海啸警报和 JMA 地震相关报文的覆盖逻辑。</p> <p>调整部分音效。</p> <p>现在海啸警报岸段线条会被绘制在陆地区域外侧。</p> <p>调整 JMA 海啸高度上升中和超出观测范围的呈现方式。</p> <p>限制地图的最大缩放, 现在地图不能再无限放大。</p>
Beta 0.3.1	2024-09-19	<p>调整部分音效。</p> <p>微调 JMA 火山口上方喷烟高度的语音播报, 并添加对状态“以上”的语音播报支持。</p>

版本	发布日期	更改
Beta 0.3.0	2024-08-19	<p>实装海啸高度的颜色主题。</p> <p>优化地图图块的加载逻辑，使其在被放大时能够在加载新图块的同时临时渲染上几个缩放级别的图块。</p> <p>添加 JMA 地震相关报文中各细分区域的震度和长周期地震动阶级的语音播报支持。</p> <p>添加并实装 JMA 火山喷发警报的颜色主题。</p> <p>实装免打扰模式。</p>
Beta 0.2.3	2024-07-05	<p>界面添加粤语支持。</p> <p>添加四个色觉障碍者专用的颜色主题。</p> <p>优化事件源的切换逻辑，现在开关事件源不再需要重启应用。</p>
Beta 0.2.2	2024-07-03	<p>修复“覆盖字体”打开的情况下，系统中存在的不能被读取的字体导致崩溃的问题。</p> <p>添加自动聚焦生效事件的设置。</p> <p>添加 JMA 火山相关报文中喷烟高度“上升中”的语音播报支持。</p>
Beta 0.2.1	2024-06-28	<p>添加对 ARM 32 位架构的支持。</p>
Beta 0.2.0	2024-06-28	<p>使用绝对速度响应频谱最大值作为 JMA 长周期地震动阶级的连续颜色替代参数。</p> <p>添加高分辨率图块设置。</p> <p>修复在安卓 13 及以上，繁体中文的信息中，时间显示为十二小时制，却没有上下午标识的问题。</p> <p>实现对等语言映射和非对等地区转换。非对等地区转换实装 F-E 区域长短名称两个版本。</p> <p>语音播报添加英语（美国）支持。</p> <p>添加额外的开屏声明。</p>
Beta 0.1.2	2024-06-07	<p>修复系统中存在的无效字体导致崩溃的问题。</p>
Beta 0.1.1	2024-06-06	<p>修复安卓 14 及以上无法启动应用的问题。</p>
Beta 0.1.0	2024-06-06	<p>添加 JMA 火山相关报文的语音播报支持。</p> <p>添加 JMA 火山相关报文的的事件源开关。</p> <p>实现 JMA 各事件源的细分开关。</p> <p>语音播报添加现代标准汉语（台湾）支持。</p>

版本	发布日期	更改
Alpha 0.0.19	2024-05-23	添加 JMA 火山相关报文中火口上喷烟高度和喷烟流向的显示。
Alpha 0.0.18	2024-05-22	添加颜色主题设置。 添加中文（繁体）界面本地化。 对 JMA 事件源添加总开关。 添加“覆盖字体”功能。
Alpha 0.0.17	2024-05-12	修复 JMA 海啸相关信息中，若有多个关联地震，只会显示和播报第一个地震的问题。
Alpha 0.0.16	2024-05-11	添加“忽略语言变体”功能。 现在接收到新的报告时，若该报告在地图上无内容可显示，则地图缩放不会再被重置。 添加 JMA 潮位观测点的翻译。
Alpha 0.0.15	2024-04-29	添加更新检测。
Alpha 0.0.14	2024-04-28	改进并统一了事件中信息源的显示方式。 添加“使用连续的颜色”功能。 将应用重命名为“CysTerra”。 修复地图自动缩放时只考虑 P 波距离而不考虑 S 波距离的问题。 若有选中的无时效性事件，则地图自动缩放时不再以其它事件为缩放依据。
Alpha 0.0.13	2024-04-22	添加 GlobalQuake 事件源。 修复地图元素在 180° 经线处的不连续问题。
Alpha 0.0.12	2024-04-20	为海啸预报线添加闪动效果。 区分 JMA 的训练报。
Alpha 0.0.11	2024-04-20	添加 JMA 远地地震海啸相关固定附加文的翻译。 统一使用汉字「強」「弱」而不用符号「+」「-」显示 JMA 震度。 点击通知栏通知时打开应用。 具有时效性的重要报文会保持显示，多个此类报文同时生效时会同时并列显示。
Alpha 0.0.10	2024-04-17	首个发布版本。

## 致 谢

感谢从本项目发布以来提供支持和反馈的所有用户。他们的反馈不仅对于项目开发本身十分珍贵，同时也是本人开发的主要动力来源之一。其中，特别感谢 GlobalQuake 社区对本项目早期开发的支持。GlobalQuake 社区管理对本项目的宣传使本项目从早期开始便在全球范围获得了一定的用户。本项目的发展离不开所有用户的支持。

感谢给本项目中 F-E 区域的中文翻译工作提供协助的国内同好者 \*\*，以及校对印度尼西亚语翻译的印度尼西亚友人 \*\*\*\*\* 和 \*\*\*\*\*。本项目的本地化工作离不开他们的贡献。

感谢全球公开提供地质灾害数据和地理数据的机构，没有他们的努力，本项目将无从以现在的方式实现。

最后，感谢导师 \*\*\* 对本文的检查和建议。



## 诚信声明

我声明，所提交的毕业论文是本人在老师指导下进行的研究工作及取得的研究成果。据我查证，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得其他教育机构的学位或证书而使用过的材料。我承诺，论文中的所有内容均真实、可信。

毕业论文作者签名：\*\*\*

签名日期：2025年6月17日

