# CountData

--Vexi统计数据保存于显示程序

## 概要

本文档旨在说明为vexi程序开发的统计数据保存与显示的程序,本文档作为用于开发人员,项目维护人员,测试人员等相关人员了解本项目所写.

本文档包含简要的需求说明,概要设计等信息.

## 环境说明

本项目使用 QT4.8.5+VS2010 (QT addlns 1.1.11 + VS2010),编码和编译.

## 需求说明

本项目是用于从vexi的程序得到统计信息,保存到数据库,并在界面显示.

级别按以下划分:

* 1 功能性需求: 指的是满足最基本的设计需求,意思是软件的核心(基本)功能。举个例子：微信 1.0 版本中只有最基本的社交功能，支持发送文字消息，分享照片给好友，对方输入信息状态。
* 2 可靠性需求: 「可靠性需求」跟建立稳定、一致的性能表现有关,本软件包括数据的稳定性,一些错误也不会导致程序崩溃或者退出。举个例子：上述微信 1.0 版本中的功能必须要稳定，尽量避免系统错误，否则达不到可靠性需求，会增加跳出率。
* 3 使用性需求: 「使用性需求」跟设计好不好、包容性高不高有关。举个例子：在微信 2.0 版本中，增加了语音对讲功能.
* 4 熟练度需求:「熟练度需求」指的是能让人们比以前做的更好。举个例子：上面说到微信的语音对讲功能，在 3.6 版本中新增了语音取消功能。
* 5 创意需求:「创意需求」是在其他需求都被满足，而且人们与设计的互动开始出现新意的时候。这个等级的设计常常获得使用者近乎虔诚的忠诚度。举个例子：微信4.0版本开始，Slogan 变 成了「微信，是一个生活方式」，标志着全新的产品价值。

具体需求如下:

|  |  |  |
| --- | --- | --- |
| 需求编号 | 需求描述 | 重要级别 |
| VCD001 | 通过Webservice从vexi程序得到XML格式数据 | 1 |
| VCD0011 | 可调整获取的时间间隔 | 3 |
| VCD002 | 解析XML数据保存到数据库 | 1 |
| VCD0021 | 解析XML时可以过滤一些错误:缺陷字段缺失等 | 2 |
| VCD0022 | 保存数据时可以根据设定策略保存指定的数据 | 3 |
| VCD0023 | 根据配置时间间隔,生成之前未生成的记录 | 1 |
| VCD0024 | 定时检查删除过期的原始数据 | 3 |
| VCD003 | 界面显示数据 | 1 |
| VCD0031 | 界面支持按时间统计 | 1 |
| VCD0032 | 界面支持中英文 | 3 |
| VCD004 | 配置管理 |  |

附:完整xml格式字符串:

<Root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:xsd="http://www.w3.org/2001/XMLSchema"

xmlns="">

<Machine Id="MX">

<Inspected>30</Inspected>

<Rejects>0</Rejects>

<Defects>0</Defects>

<Autoreject>0</Autoreject>

<Mold id="0">

<Inspected>30</Inspected>

<Rejects>0</Rejects>

<Defects>0</Defects>

<Autoreject>0</Autoreject>

<Sensor id="16">

<Rejects>0</Rejects>

<Defects>0</Defects>

</Sensor>

<Sensor id="40">

<Rejects>0</Rejects>

<Defects>0</Defects>

</Sensor>

<Sensor id="41">

<Rejects>0</Rejects>

<Defects>0</Defects>

</Sensor>

<Sensor id="42">

<Rejects>0</Rejects>

<Defects>0</Defects>

</Sensor>

<Sensor id="54">

<Rejects>0</Rejects>

<Defects>0</Defects>

</Sensor>

</Mold>

</Machine>

</Root>

注:

1 Machine标签内的所有数据都可能缺失;

2 暂时可不检查数据完整性.

细节说明表:

|  |  |
| --- | --- |
| 序号 | 描述 |
| 1 | 设备默认是24小时运行, |
| 2 | 时间间隔配置影响后期生成时间,不影响之前已经生成的记录时间间隔 |
| 3 | 如果之前没数据,界面需要提示没有数据 |
| 4 | 界面数据 按照错误占比升序显示 |
|  |  |

## 概要设计

### 程序总体图



### Config

#### 功能说明

本模块的功能是负责读取系统需要的重要配置,..

#### 功能点

* 读取数据模块保存元数据的过滤策略配置
* 提供重要的时间间隔参数
* 保存生成日志的时间间隔接口
* 读取软件显示语言配置

### DataCenter

#### 功能说明

本模块的功能是解析传入的XML字符串,按照策略保存到数据库中.

#### 功能点

* 解析XML字符串
* 读取策略文件
* 保存数据到Sqlite
* 控制日志输出级别

### DataView

#### 功能说明

本模块的功能是从设定的地方获取数据,并按要求显示到界面上..

#### 功能点

* 设定数据来源
* 根据界面设定获取数据(时间间隔(30分钟|10分钟…)
* 过滤某些数据不显示(模板|缺陷代码…)
* 中英文切换

### CountData

#### 功能说明

本模块的功能负责从VEXI获取数据,并传给DataCenter解析.

#### 功能点

* 创建DataCenter,DataView对象,并初始化
* 使用Webservice从VEXI获取数据

附;配置示例及说明:

Config.ini文件:

[strategy]

mode=true

sensorids\size=0

;sensorids\1\0=14

;sensorids\2\1=34

islogall=false

[system]

time\_interval\_generate\_record=0

time\_interval\_get\_src\_data=60

data\_outdate\_days=30

language=1

code\_file\_prefix=CodeTs

如上所示:

strategy.mode: 缺陷代码命中模式:

* true表示白名单模式,意思是此模式下方的sensorids里的缺陷ID会被保存;
* false表示黑名单模式,意思是此模式下方的sensorids里的缺陷ID不会被保存

strategy.islogall:true表示数据中心的日志会记录所有解析的xml字符串.

system. time\_interval\_generate\_record:生成记录数据的时间间隔,

* 0:30分钟,
* 1:60分钟,
* 2:90分钟
* 3:120分钟

system. time\_interval\_get\_src\_data:通过webservice获取数据的时间间隔 单位秒

system.data\_outdate\_days:原始数据过期时间,单位天

system.language:0中文,1英文

system.code\_file\_prefix:使用的代码(包含模板代码和缺陷码)的词条对应文件名前缀

文件放在运行目录下的文件下,如示例配置,完整文件名为CodeTs\_en.ini,如下所示,其中MoldWords下的是模板代码对应的词条,SensorWords下对应的是缺陷码对应的词条.

示例文件:

[MoldWords]

0=Normal

1=Tailand

2=India

[SensorWords]

4=LNMC

12=BOC

16=LNM/LNMP

35=DIMBAG

40=PLUG

41=LEAK

42=CHECK

43=OVALITY

44=CSB

46=THICKNESS

47=NCI/HCI

49=ATLAS

54=EXTENSION1

55=EXTENSION2

110=COMPOSITE COUNTER