# XX数据库处理详细设计

## 数据下载后整体数据处理流程图

## 数据处理的类设计，每个数据库处理的程序在分别放在com/biodata/分数据库名称包下。例如MINT数据库，其所有处理程序都在com/biodata/mint目录下

## 数据处理的详细流程描述，包括数据如何处理，如何解析入库，正确的数据如何处理，错误的数据如何处理，如何处理异常

xml文件的解析，利用xml.etree.ElementTree 来解析xml文件，首先使用xml.etree.ElementTree.parse读入文件，接着使用getroot()方法获取根节点，然后解析根节点下面的子节点。循环遍历获取节点的字段和属性值，放到一个空的字典里。

*# -\*- coding: utf-8 -\*-***import** xml.etree.ElementTree **as** ET  
**def** attr\_gbk(elem, name):  
 **return** elem.attrib.get(name).encode(**"GBK"**)  
**def** attr\_int(elem, name):  
 **return** int(elem.attrib.get(name))  
  
**def** attr\_bool(elem, name):  
 **return** bool(elem.attrib.get(name))  
  
**def** attr(elem, \*args):  
 **if** len(args) > 1:  
 result = []  
 **for** name **in** args:  
 result.append(attr(elem, name))  
 **return** result  
 [name] = args  
 value = elem.attrib.get(name)  
 **if not** value:  
 **return** value  
 **elif** value.isdigit():  
 **return** int(value)  
 **elif** value.lower() **in** [**"true"**, **"false"**]:  
 **return** value.lower() == **"true"  
 else**:  
 **return** value  
  
**def** fix\_attribs(elem):  
 dict\_ = {}  
 **for** key **in** elem.attrib:  
 dict\_.update({key: attr(elem, key)})  
 **return** dict\_  
  
**def** build\_dict(elem):  
  
 **if** elem **is not None**:  
  
 dict\_ = {}  
 **for** subelem **in** elem:  
 *# print(subelem.tag)* **if** subelem.tag **in** dict\_:  
 **if not** isinstance(dict\_[subelem.tag], list):  
 dict\_[subelem.tag] = [dict\_[subelem.tag]]  
 dict\_[subelem.tag].append(build\_dict(subelem))  
 **else**:  
 dict\_.update({subelem.tag: build\_dict(subelem)})  
 **if** subelem.text **and** subelem.text.strip():  
 dict\_.update({subelem.tag: {**"\_text"**: subelem.text}})  
 dict\_.update(fix\_attribs(elem))  
 **return** dict\_  
 **else**:  
 **return** fix\_attribs(elem)  
  
**if '\_\_main\_\_'**==\_\_name\_\_:  
 tree = ET.parse(**"D:/00002\_psimi.xml"**)  
 root = tree.getroot()  
  
 print(build\_dict(root))

txt文件的解析：使用open()方法打开txt文件，逐行读取数据。

*# -\*- coding: utf-8 -\*-***def** load\_txt(fname):  
 dict = {}  
 **with** open(fname, **'r'**) **as** fn:  
 **for** i **in** fn:  
 dict[i[0]]=[i[1:]]  
 **return** dict  
  
**if** \_\_name\_\_==**"\_\_main\_\_"**:  
 result = load\_txt(**'D:/aaa.txt'**)  
 print(result)