**6．文件传输数据包(filetransfer)**

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以3种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有3种类型

file\_data： 服务器对采集进行文件传输

pack\_lost: 服务器查询文件传输缺包情况

pack\_lost\_ack: 采集器回应文件传输缺包的情况

元素有 6 种类型

pack\_total: 文件数据包总数目, 仅在file\_data时有效

pack\_sequence: 传输过程中该包的序号, 仅在file\_data时有效

file\_type: 文件类型，下载表地址是:.meteraddress, 仅在file\_data时有效

file\_data: 文件二进制流数据, 仅在file\_data时有效

pack\_lost\_total: 文件传输缺包的总数，仅在pack\_lost\_ack时有效

pack\_lost\_index: 文件传输缺包的包序号, 仅在pack\_lost\_ack时有效，当pack\_lost\_total 为0时 该项无效

-->

<filetransfer operation="**file\_data/pack\_lost/pack\_lost\_ack** ">

<pack\_total>**100**</pack\_total>

<pack\_sequence>**1**</pack\_sequence>

<file\_type> **.db** </file\_type>

<file\_data> **xxxxxx….xxxxx** </file\_data>

<pack\_lost\_total> **xx** </ pack\_lost\_total >

<pack\_lost\_index> **xx,xx,xx…..xx,xx** </ pack\_lost\_index >

</ filetransfer >

</root>

**7．查询配置数据包(query\_config)**

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

building\_id: AAAAAA

gateway\_id: AAA

以上ID编码为广播查询编号，可对任意采集器进行点对点的配置信息查询

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以2种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有2种类型

Query\_config： 服务器查询采集器额外配置信息，不需要子元素

Reply\_config: 采集器对服务器查询信息的应答

元素有 3 种类型

Period：数据采集周期

Report\_mode: 数据上报模式 1:主动上报；0:被动查询

soft\_ver: 软件版本号

-->

<query operation="**query\_config/reply\_config** ">

<period>**15**</period>

<report\_mode>**1**</report\_mode>

<soft\_ver> **XXX** </soft\_ver>

</query>

</root>

**8．额外配置数据包(ex\_config)**

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以2种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有4种类型

Id\_config: 楼栋和采集器编号设置

Id\_ack: 采集器对服务器配置楼栋和采集器编号的回应

Report\_mode: 设置采集器数据上报模式

Report\_ack: 采集器对服务器设置采集器数据上报模式的回应

元素有 3 种类型

New\_building\_id：楼栋新编号，仅在Id\_config 有效

New\_gateway\_id: 采集器新编号, 仅在Id\_config 有效

Report\_mode: 数据上报模式 1:主动上报；0:被动查询。仅在report\_mode时有效

-->

< ex\_config operation="**id\_config/id\_ack/report\_mode/report\_ack**">

<new\_building\_id> **XXXXXX** </new\_building\_id>

<new\_gateway\_id> **XXX** </new\_gateway\_id>

<report\_mode>**1**</report\_mode>

</ ex\_config>

</root>

**9．密钥更新数据包(update\_secretkey)**

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以3种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有3种类型

Md5\_key：服务器设置md5密钥

AES\_key: 服务器设置AES密钥

Key\_ack: 采集器对服务器密钥设置的应答

元素有 2 种类型

Md5\_keydata: 128位md5校验密钥，仅在md5\_key时有效

Aes\_keydata: 128位AES加密密钥，仅在AES\_key时有效

-->

<update operation="**md5\_key/aes\_key/key\_ack** ">

<md5\_keydata>**xxxx…xxxx**</md5\_keydata>

<aes\_keydata>**xxxx…xxxx**</aes\_keydata>

</update>

</root>

**10．服务器控制指令数据包(control)**

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以2种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有2种类型

reboot：服务器控制采集器重新启动

control\_ack: 采集器对服务器控制的应答

元素有 种类型

-->

<control operation=" **reboot/control\_ack** ">

</control>

</root>

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*杨敏 added -2013.08.15 有讨论的地方联系杨敏\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*TEL:18629400162\*\*\*\*/

**11．服务器查询/上报数据表(report\_srv)**

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

building\_id: AAAAAA

gateway\_id: AAA

以上ID编码为广播查询编号，可对任意采集器进行点对点的配置信息查询

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以下2种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有2种类型

Query\_req: 服务器查询采集器数据功能

Query\_rsp: 采集器查询功能答复

元素有 4 种类型

condition： 数据上报的条件,仅在Query\_req时有效,使用标准的sql语句

flag: 0/1：成功/失败标志，仅在Query\_rsp时有效

result: 上报的查询结果，仅在Query\_rsp时有效,count为查询的数据个数

table\_item: 查询的表项，仅在Query\_rsp时有效

-->

< report\_srv operation=" **Query\_req/ Query\_rsp** ">

<condition> **select \* from Meter\_Info\_Table** </conditon>

<flag> **0** </flag >

<result count= **“20”**>

< table\_item> **15-13-01-01,** **11110071100109,25,1,3,1010, 0001** </ talbe\_item >

< table\_item >  **XXX** </ talbe\_item >

< talble\_item >  **XXX** </ talbe\_item >

</result >

</ report\_srv >

</root>

**12．采集器数据表主动上报(report\_tbl)**

**作用:** 当

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

building\_id: AAAAAA

gateway\_id: AAA

以上ID编码为广播查询编号，可对任意采集器进行点对点的配置信息查询

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以1种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!-- 操作有4种类型

query:服务器查询数据采集器，不需要子元素

reply:采集器对服务器查询的应答

report:采集器定时上报的能耗数据

continuous:采集器断点续传的能耗数据

continuous\_ack:全部续传数据包接收完成后，服务器对断点续传的应答，不需要子元素

元素有5种类型

sequence元素:采集器向服务器发送数据的序号

parser元素:向服务器发送的数据是否经采集器解析过

time元素:数据采集时间

total元素:断点续传时有效，表示需要断点续传数据包的总数

current元素:断点续传时有效，表示当前断点续传数据包的编号

port元素:计量装置的设备号，具有id属性

function元素:每个计量装置的具体采集功能，具有下列3种属性，

id属性:计量装置的数据采集功能编号（供多功能电能表使用）

coding属性:能耗数据分类/分项编码

error属性:该功能出现错误的状态码，0表示没有错误

-->

<!--操作有4种类型

realtime\_qry\_req： 服务器查询数据表请求

realtime\_qry\_rsp: 采集器查询数据表答复

元素有 2 种类型

meter\_addr： 表地址，支持多个表同时查询，仅在realtime\_qry\_req有效

type: 表类型 10：水表 20：热表 21：电表 25：阀门

version：协议类型

result: 成功/失败标志，仅在realtime\_qry\_rsp时有效

meter 上报数据格式为：Electric\_Meter\_Table/ Heat\_Meter\_Table/ Water\_Meter\_Table/ Valve\_Table

例如：上报热表数据格式为：Meter\_addr，Read\_time，Subitem，Acc\_heat

erro: 0:没有错误，1：通讯超时，2：数据验证错误

<data operation="query/reply/report/continuous/continuous\_ack">

<sequence>**XXXXXXXX**</sequence>

<parser>yes/no</parser>

<time> YYYYMMDDHHMMSS </time>

<total>XXX</total>

<current>XXX</current>

<user id="**15-13-01-01**">

<table type=”25”>数据</table>

</ user>

<user id="**15-13-01-02**">

<table type=”25”>数据</table>

</ user>

</data>

</root>

**13．服务器修改报数据表(modify)**

**作用：**当表信息有个别错误时，可根据User\_id修改其中的几项数据，而不需要重新下发表地址

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

building\_id: AAAAAA

gateway\_id: AAA

以上ID编码为广播查询编号，可对任意采集器进行点对点的配置信息查询

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以1种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有4种类型

modify\_req： 服务器修改数据表请求

modify\_rsp: 采集器修改数据表答复

元素有 3 种类型

Content：数据修改内容,仅在modify\_req时有效,使用标准的sql语句

Num: 修改语句的个数，最大支持一次带50个修改语句

flag: 成功/失败标志，仅在modify\_rsp时有效,0为成功，大于0为在执行第几个content时失败

<modify operation=" **modify\_req/modify\_rsp** " num= “6”>

<content id=”1”> **update Meter\_Info\_Table SET Meter\_type ='20' where** **User\_id='15-13-01-01'** </ content >

<content id=”**XX”**> **XXXXXX** </ content >

<flag> **XXX** </flag >

</ modify >

</root>

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**14．服务器实时查询数据(realtime\_qry)**

<?xml version="1.0" encoding="utf-8" ?>

<root>

<!-- 通用部分 -->

<!—

building\_id:楼栋编号

gateway\_id:采集器编号

type:配置信息数据包的类型

building\_id: AAAAAA

gateway\_id: AAA

以上ID编码为广播查询编号，可对任意采集器进行点对点的配置信息查询

-->

<common>

<building\_id>**XXXXXX**</building\_id >

<gateway\_id>**XXX**</gateway\_id >

<type>**以1种操作类型之一**</type>

</common>

<!-- 配置信息 -->

<!--操作有4种类型

realtime\_qry\_req： 服务器查询数据表请求

realtime\_qry\_rsp: 采集器查询数据表答复

元素有 2 种类型

Item:下发的表信息，仅在realtime\_qry\_req时有效

type: 表类型 10：水表 20：热表 21：电表 25：阀门

ProtocolVer：协议类型

Channel: 通讯通道，1-6为mbus通道，7-8为485通道

result: 成功/失败标志，仅在realtime\_qry\_rsp时有效

erro: 0:没有错误，1：通讯超时，2：数据验证错误

meter 上报数据格式为：Electric\_Meter\_Table/ Heat\_Meter\_Table/ Water\_Meter\_Table/ Valve\_Table

例如：上报热表数据格式为：Meter\_addr，Read\_time，Subitem，Acc\_heat

<realtime\_qry operation=" **realtime\_qry\_req/realtime\_qry\_rsp**">

<item type=”**10/20/21**” ProtocolVer=”**0/1**” channel=”2” > **51101211110010**</ item>

< result error= **“0”** > **XXX**</result>

</ realtime\_qry >

</root>