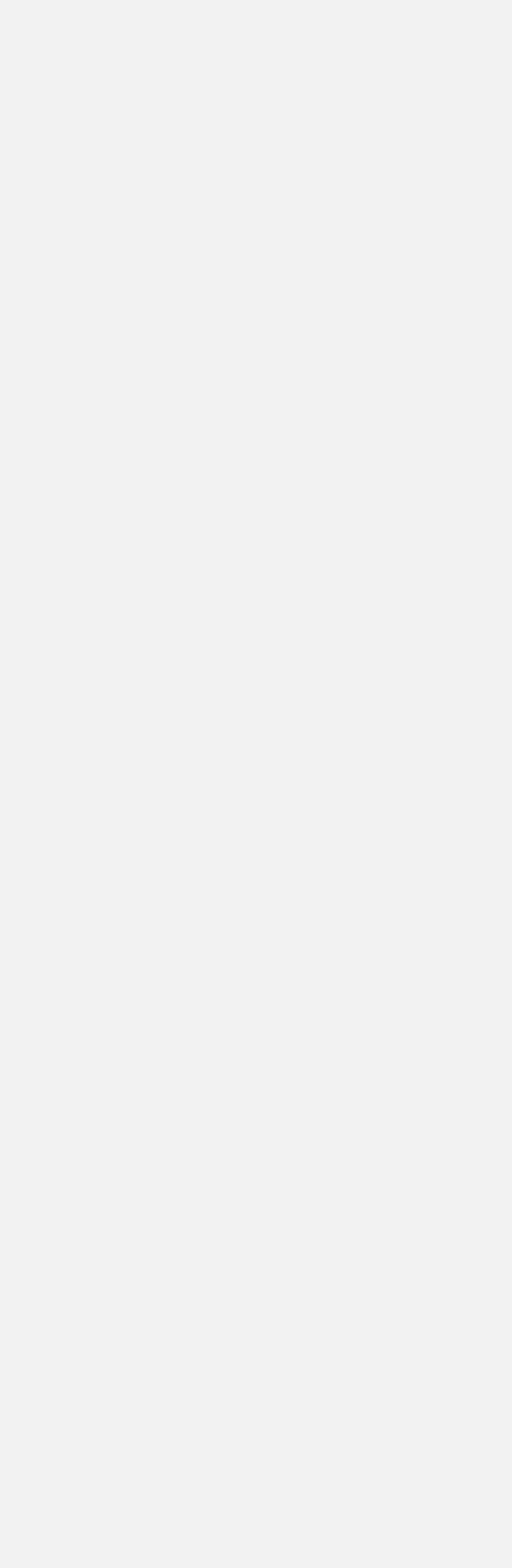
**【重要提醒】 字数超过限制, 部分内容不再显示出来.**



**Smartone-沃达丰**

**ADW项目**

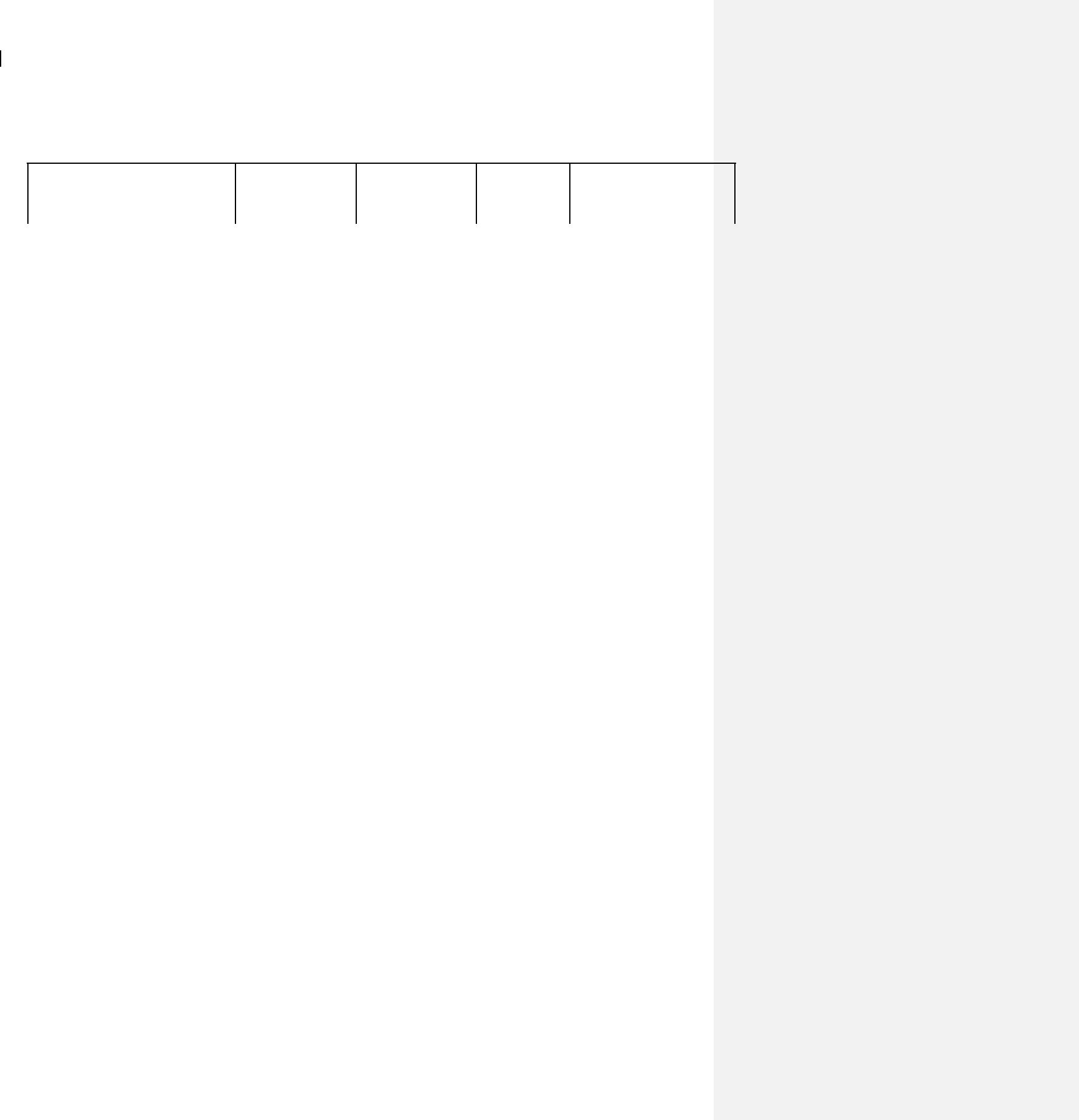
**ETL技术规范**

**版本1.154**

**2005年11月15日至4日**



**Smartone-Vodafone&NCR机密**



ETL技术规范2005年11月15日至4日

ETL技术规范

**文档准备信息**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 项目名称 | 编制 | 依据 | 签名 | 日期 | 文档ID |
|  | （打印） |  |  | 编制 |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Smartone-沃达丰ADW | 李彦宏 | | | | 9/28/2005 | | | ETL技术 | | | |
| 项目 |  |  |  |  |  |  |  | 规格 | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **客户信息** | | | | |  |  |  |  |  |  |  |
| 客户 |  |  | 联系人 | |  |  |  |  | 客户帐户 | |  |
| Smartone-沃达丰 |  |  | 陈冯富珍 | |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **一般信息** |  |  |  |  |  |  |  |  |  |  |  |
| 项目经理 |  |  |  | 安全分类 |  |  |  |  |  |  |  |
| 陈健 |  |  |  | NCR和Smartone-沃达丰保密 | | | | | |  | |
|  |  |  | |  |  |  |  |  |  |  |  |
| **发起人信息** | | | | |  |  |  |  |  |  |  |
| 发起人 |  |  |  | NCR组织名称 |  |  |  |  |  |  |  |
| 陈健 |  |  |  | Teradata分部 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| NCR组织地址 |  |  |  |  |  |  |  |  |  |  |  |
| 香港铜锣湾时代广场RBS大厦22楼 | | | | |  |  |  |  |  |  |  |
|  |  |  | | |  |  |  |  |  |  |  |
| 电话号码 |  | 传真号码 | | |  |  | 电子邮件 |  |  |  |  |
| 2859-6048 |  | 2548-4679 |  |  |  |  | [ken.chan@ncr.com](mailto:ken.chan@ncr.com) | |  |  | |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **版本信息** |  |  |  |  |  |  |  |  |  |  |  |
| 上次更新时间 | 主版本位置\* | | | |  |  |  |  |  |  |  |
| 版本1.14 | NCR/Smartone-沃达丰项目办公室 | | | |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

\*详情见下文准则。

***注意:在使用之前，读者必须验证这是当前版本。***

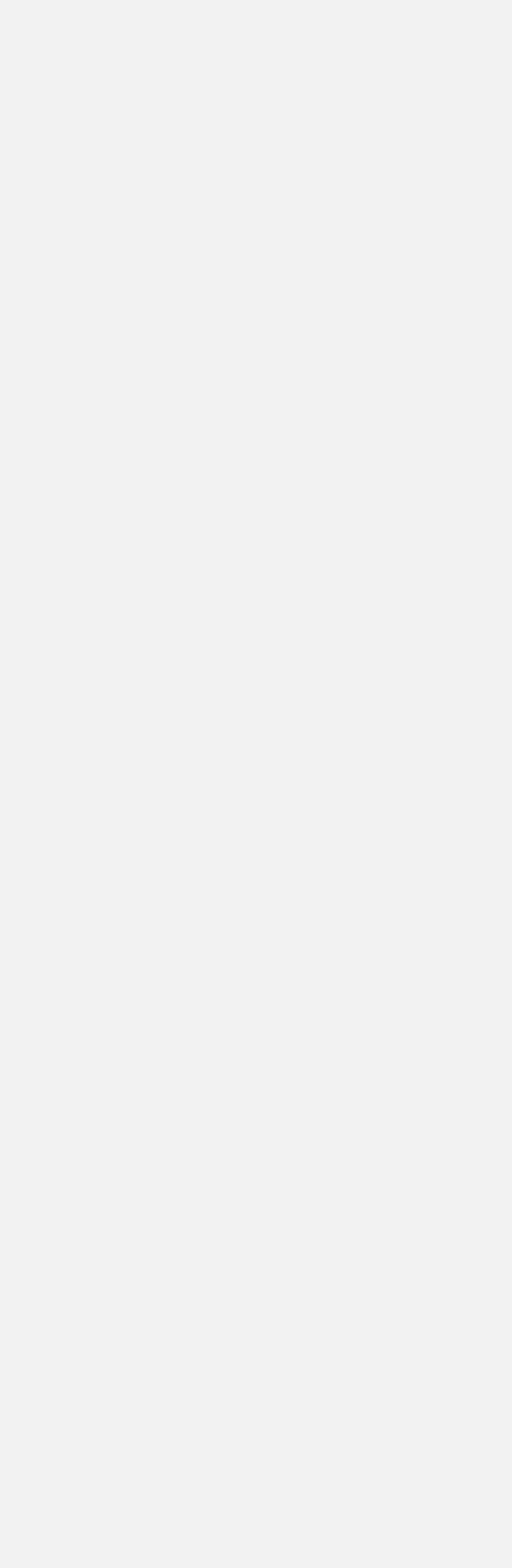
**客户接受**

|  |  |  |  |
| --- | --- | --- | --- |
| 名称(PRINT) | 签名 |  | 日期 |
|  |  |  |  |
| 电话 | 传真 | 电子邮件 | |
|  |  |  |  |
| 名称(PRINT) | 签名 |  | 日期 |
|  |  |  |  |
| 电话 | 传真 | 电子邮件 | |
|  |  |  |  |

**指导方针**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **师傅** |  |  | [\\SHK\_DEMO0CMS\CMSProjs\Telecom\SmartOne ADW Project\Document\SMC-V ADW Project-](//SHK_DEMO0CMS/CMSPROJS/Telecom/SmarTone ADW Project/Document/SMC-V ADW Project - ETL Technical Specification.doc) | |  |  |
| **版本** | [ETL技术规范。doc](//SHK_DEMO0CMS/CMSPROJS/Telecom/SmarTone ADW Project/Document/SMC-V ADW Project - ETL Technical Specification.doc) | | | |  |  |
|  |  |  |  |  |  |
| **位置** |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

ETL技术规范 第2页，共46页



ETL技术规范 15~~4。~~ 2005年11月

文件控制

**文件分发**

|  |  |  |
| --- | --- | --- |
| **名称** | **标题** | **组织机构** |
|  |  |  |

Smartone-沃达丰

Smartone-沃达丰

Smartone-沃达丰

Smartone-沃达丰

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 萧炯柱 | | | |  |  |  | NCR |  |
|  |  | |  |  |  |  |  |  |
| 陈健 | | | |  |  |  | NCR |  |
|  |  | |  |  |  |  |  |  |
| 小圆王 | | | |  |  |  | NCR |  |
|  |  | |  |  |  |  |  |  |
|  |  | |  |  |  |  |  |  |
|  |  | |  |  |  |  |  |  |
| **更改历史记录** | | | |  |  |  |  |  |
|  |  | **日期** | |  | **更改说明** | | **核准人** |  |
|  |  | |  |  |  |  |  |  |
| 2005年9月28日 | | | |  | TOC评审版本 | |  |  |
|  |  |  |
|  |  |  |  |  |  |  |  |  |
| 22 | | 2005年10月 | |  | 已将详细信息添加到所有项目 | |  |  |
|  |  |  |  |  |  |  |  |  |
| 26 | | 2005年10月 | |  | 内部审查后的修改 | |  |  |
|  |  | |  |  |  |  |  |  |
| 11月4日 | | | |  | SMC-V反馈后的修改 | |  |  |
|  | |  | |  |  | |  |  |
| 15 | | 11月 | |  | 添加特殊处理脚本参数 | |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

ETL技术规范 第3页，共46页

ETL技术规范 15~~4。~~ 2005年11月

**目录目录**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1。** | **ETL概述。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。** | | **6。** | | |
| **2。** | **逻辑工艺流程。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。** | | **6。** | | |
|  | 2.1 | 数据源 …………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。 | 7 | | |
|  | *2.1.1* | *拉…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | *8* | | |
|  | *2.1.2* | *推………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | *8* | | |
|  | *2.1.3* | *正在提取………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | *8* | | |
|  | *2.1.4* | *中转区……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。 9* |  |  | *~~8~~* |
|  | 2.2 | d阿塔F ILES预-加工………………………………………………………………………………………。 | 9 | | |
|  | *2.2.1* | *预处理后台进程的典型功能。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *9* | | |
|  | *2.2.2* | *预处理配置表。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *9* | | |
|  | *2.2.3* | *数据文件通配符匹配。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *15* | *~~14~~* | |
|  | *2.2.4* | *控制文件和数据文件结构。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *16* | *~~15~~* | |
|  | *2.2.5* | *使用特殊处理脚本。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *16* | | |
|  | *2.2.6* | *可重新启动的拉拔过程。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *18* | *~~16~~* | |
|  | *2.2.7* | *加载差异系数-检查类型。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *18* | *~~16~~* | |
|  | *2.2.8* | *加载方差检查机制…………………………………………………………………………………………………………………………………………………………………………………………………………………………………….* | *19* | *~~17~~* | |
|  | *2.2.9* | *编程语言…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | *19* | *~~17~~* | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *2.2.10* | | *工艺流程设计。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *19* | | *~~17~~* | |
| 2.2.10.1 | | 预处理后台进程。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。 20 |  |  | | ~~17~~ |
| *2.2.11* | | *预处理相关目录。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *25* | | *~~23~~* | |
| 2.3 | 数据测试改造以及 装载方法……………………………………………………………………… | | 26 | | ~~24~~ | |
| *2.3.1* | *向上插入………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | | *27* | | *~~24~~* | |
| *2.3.2* | *取代…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | | *27* | | *~~25~~* | |
| *2.3.3* | *历史…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | | *27* | | *~~25~~* | |
| *2.3.4* | *追加…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。* | | *28* | | *~~26~~* | |
| *2.3.5* | *表映像切换方法。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | | *28* | | *~~26~~* | |
| *2.3.6* | *CDR表实现。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | | *29* | | *~~26~~* | |
| *2.3.7* | *重复的行处理………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………….* | | *29* | | *~~27~~* | |
| 2.4 | POST处理………………………………………………………………………………………。 | | 30 | | ~~27~~ | |
| *2.4.1* | *输出方差检查……………………………………………………………………………………………………………………………………………………………………………………………………………….* | | *30* | | *~~27~~* | |
| *2.4.2* | *表AS\_OF\_DATE和REFRESH\_DATE.。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | | *30* | | *~~28~~* | |
| *2.4.3* | *表收集统计数据………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………….* | | *31* | | *~~29~~* | |
| **3.ETL自动化。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。** | | | **32** | | **~~29~~** | |
| 3.1 | bATCHW 印多…………………………………………………………………………………………。 | | 32 | | ~~29~~ | |
| *3.1.1* | *夜间批量装载…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………….* | | *32* | | *~~29~~* | |
| *3.1.2* | *立即装货……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………….* | | *32* | | *~~30~~* | |
| *3.1.3* | *文件目录流。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | | *32* | | *~~30~~* | |
| *3.1.4* | *每天在ETL Automation中多次运行作业............................................................................................* | | *34* | | *~~31~~* | |
| *3.1.5* | *作业重新运行。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | | *35* | | *~~32~~* | |
| *3.1.6* | *ETL自动化目录。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | | *35* | | *~~32~~* | |
| 3.2 | 密码加密 …………………………………………………………………………………………。 | | 36 | | ~~33~~ | |

**他说:**



**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

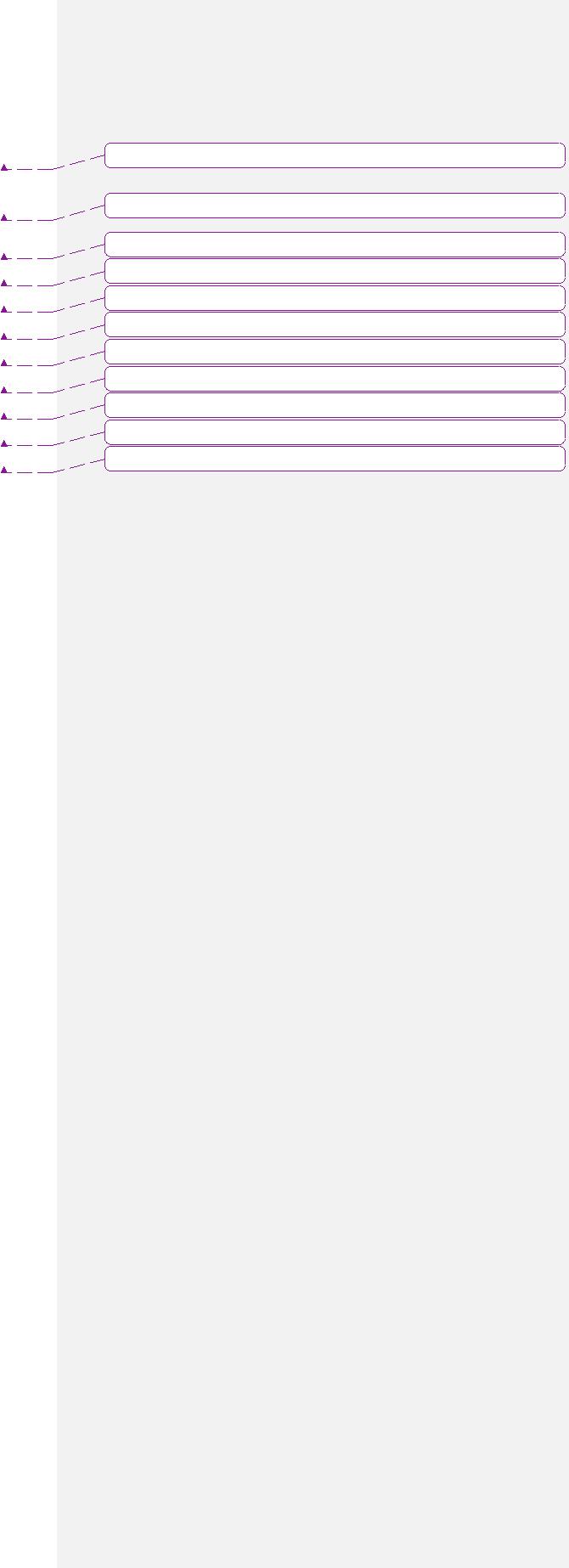
**他说:**

ETL技术规范 第4页，共46页

ETL技术规范 15~~4。~~ 2005年11月

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4。** | **HP OpenView.。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。** | | **36** | **~~33~~** |
| **5。** | **附录……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………。** | | **37** | **~~33~~** |
|  | 5.1 | ETL自动化操作游戏攻略 ………………………………………………………………………… | 37 | ~~33~~ |
|  | *5.1.1* | *概览……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………….* | *37* | *~~34~~* |
|  | *5.1.2* | *每日和每月作业设置。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *42* | *~~38~~* |
|  | *5.1.3* | *环境参数文件。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *43* | *~~39~~* |
|  | 5.2 | d航海作业游戏攻略………………………………………………………………………… | 43 | ~~40~~ |
|  | 5.3 | ETL P性能调谐………………………………………………………………………………………。 | 44 | ~~40~~ |
|  | *5.3.1* | *分区主索引。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。* | *45* | *~~41~~* |
|  | *5.3.2* | *次级索引................................................................................................................* | *45* | *~~41~~* |
|  | 5.4 | 我可辨认的E 错误M 散文…………………………………………………………………………………………。 | 45 | ~~41~~ |

**他说:**



**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

**他说:**

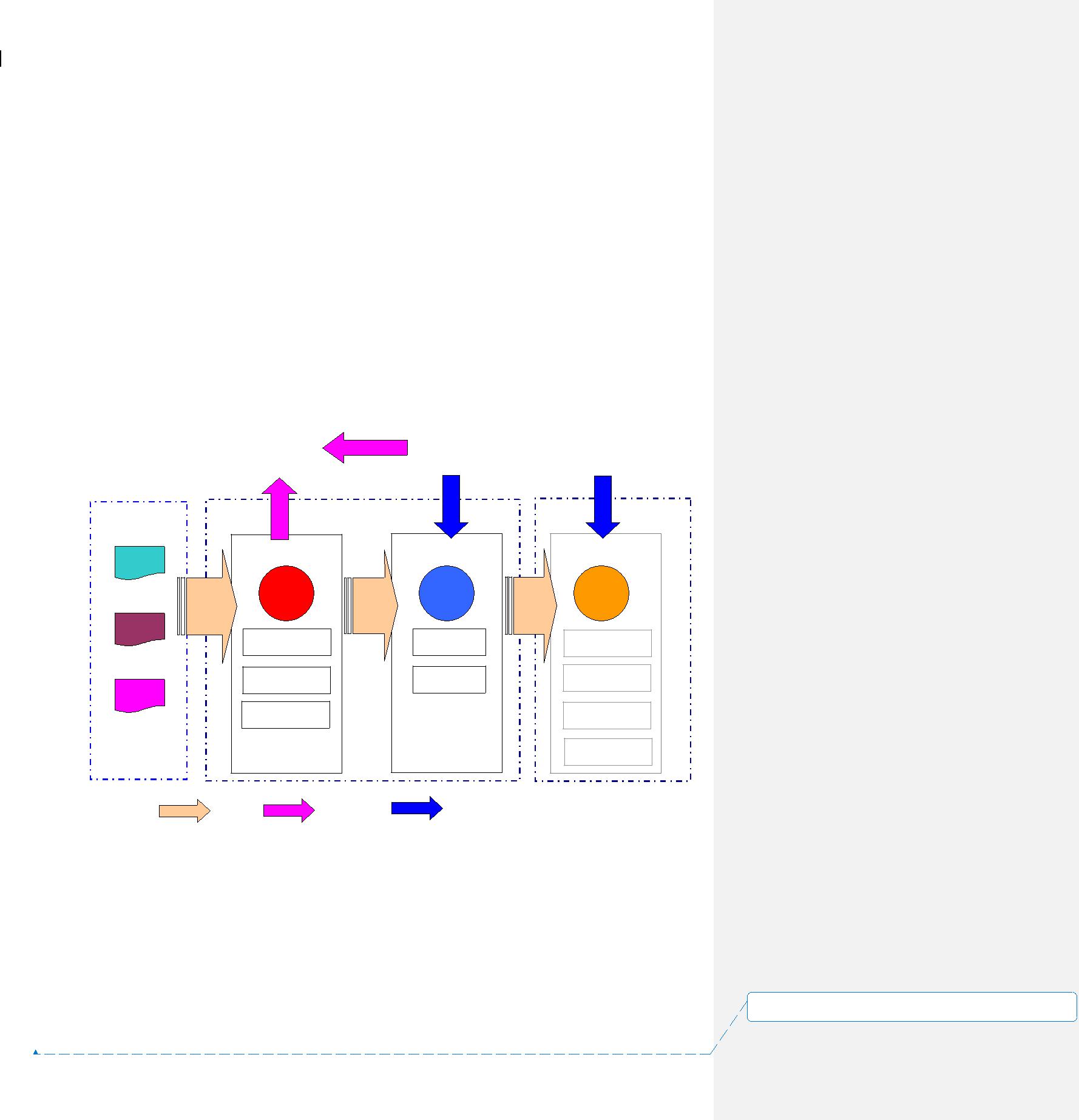
**他说:**

**他说:**

**他说:**

**他说:**

ETL技术规范 第5页，共46页



ETL技术规范 15~~4。~~ 2005年11月

1. **ETL概述**

ETL在现代数据仓库技术中占有重要地位。 这包括从外部源系统提取数据、将数据转换为适合数据模型的数据以及将数据加载到数据仓库中。 在提取过程中，数据图像被捕获到数据文件中。 数据文件以固定宽度或分隔符记录的形式存在。 对数据文件进行进一步的操作，并将其转换为适当的格式。 将进行代码转换、合并和转换。 最后阶段加载DAT

1. **a根据数据的性质和业务需要采用不同的方法。**

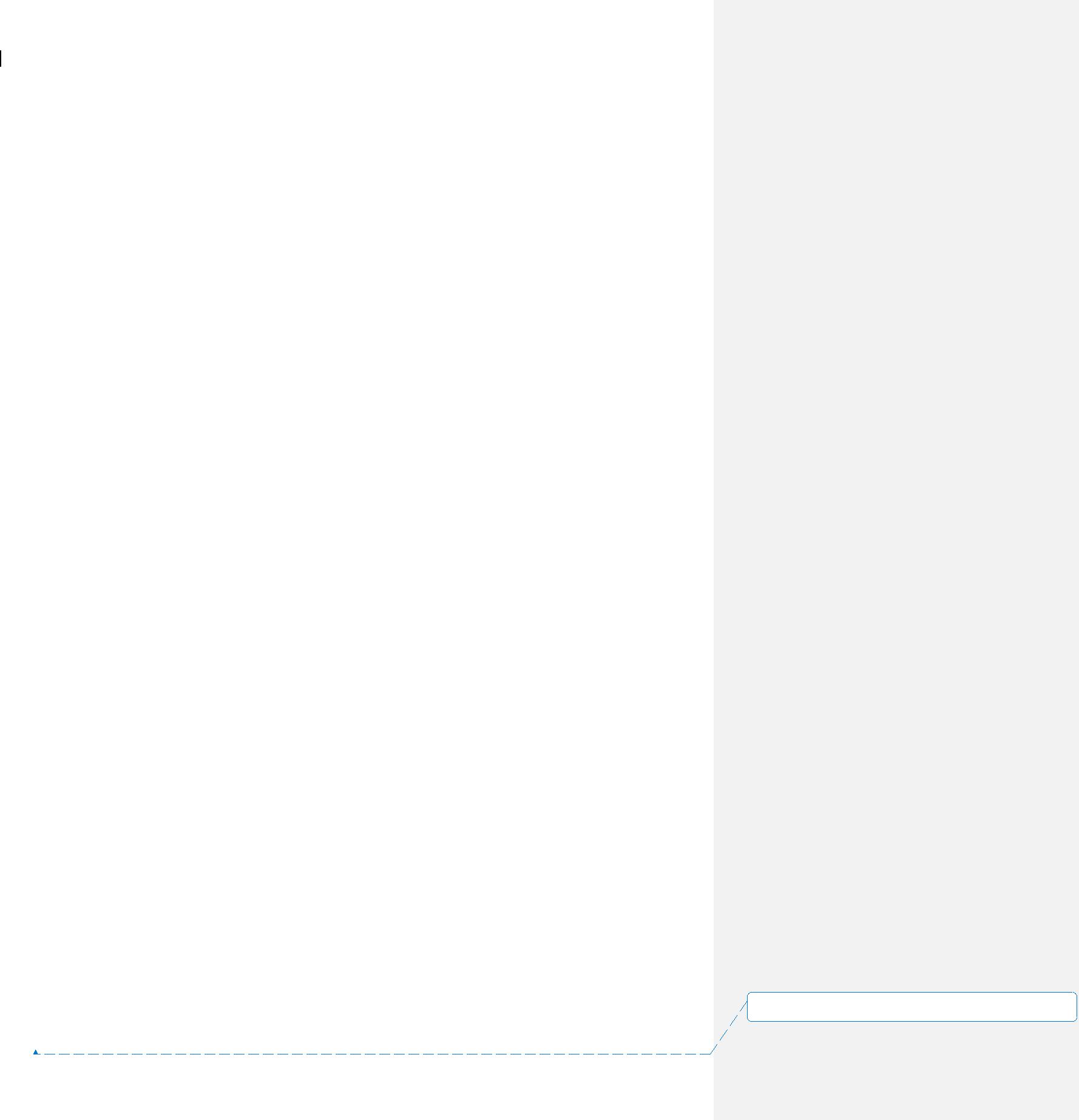
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **逻辑工艺流程** | | **惠普** |  |  |
|  | **ETL** | | **OpenView视图** |  |  |
| **自动化** |  | **数据文件** |  |  |  |
|  | **Sun Solaris** | | **预处理** | **数据存储** |  |
|  | **Teradata SQL** | **推** | **P1** | **P2** |  |
|  | **P3** |  |  |  |  |
|  | **拉** |  |  |  |  |
|  |  | **提取物** | **输入** | **简单** |  |
|  | **复合体** | | **差异检查** | **变换** |  |
|  |  | **变换** | **特殊文件** | **负载** |  |
|  |  | **输出** | **搬运** |  |
|  |  |  |  |
|  |  | **差异检查** |  | **发送控制** |  |
|  |  | **收集** |  | **文件** |  |
|  |  |  |  | **统计数字** |  |
|  |  |  |  | **更新** |  |
| **截止日期** | **图例:** | **数据** | **发送** |  |  |
| **控制和** | **流动** | **消息** |  |  |
|  |  |  |

* 监视器
* 数据源来自不同的机器。 它们中的一些周期性地被推送到分段区域，而另一些则可能需要拉出和提取。

预处理后台进程将处理数据文件的到达。 它强制检查文件并保存结果以进行差异检查。 它还调用特殊的处理脚本来执行特定的解密、转置处理。 最后，生成控制文件来通知ETL作业的启动。~~0\_1.单据~~ SMC-V ADW项目-ETL技术规范。doc

佩奇

**46个中的6个** 大小:（）（）（）（）（）（）），



身高 ETL技术规范~~15~~ 4。

* 2005年11月
* DataStage参与一些简单的转换和加载作业。
* 对于更复杂和更繁重的转换，它将由Teradata SQL完成。 数据差异检查和拒绝记录检查也将在后处理阶段在Teradata中进行。 毕竟，该表是收集统计信息的。
* ETL自动化将控制ETL作业的整个过程。 它们包括DataStage转换和加载作业、Teradata转换作业以及统计日志记录。 它根据存储库中预定义的依赖项执行这些作业。

HP Overview将是所有组件的主要监控控制台。 有关作业失败的详细故障排除，请参阅ETL Automation和详细的作业日志。

* ETL过程主要分为3个阶段:
* P1-预处理阶段处理不同的源，并验证文件是否适合加载。
* P2-DataStage阶段负责将数据简单地转换和加载到Teradata中。

P3-Teradata SQL阶段进一步操作加载的表，并将数据集成到现有的表映像中。

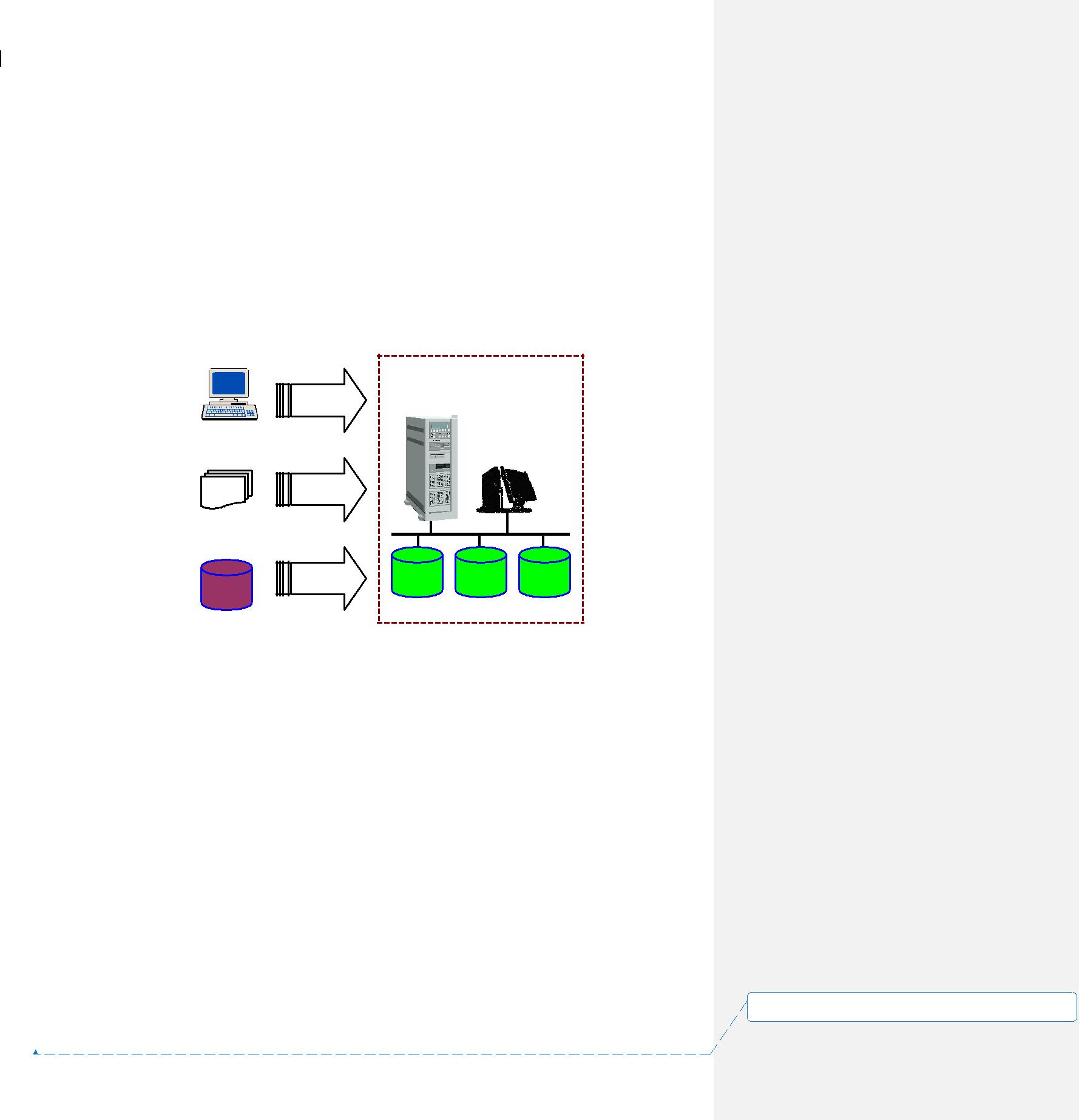
**以下各节将详细解释这些阶段。** **2.1**

数据源

Smartone-Vodafone有多种源系统。 为了便于设计和实现，将使用一组应用程序缩写代码来表示它们。 这提供了脚本、数据文件的逻辑分组，并有助于避免来自不同源系统的文件名重复。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **缩写代码概述如下:** | **数据源** |  |
|  |  |  | **数据源** |  |
|  |  | 代码 | 应用服务器 |  |
|  |  | 作为 | 计费系统 |  |
| BLS | ~~0\_1.单据~~ | |  | SMC-V ADW项目-ETL技术规范。doc |
| 佩奇 | | |  |  |

**46个中的7个** 大小:（）（）（）（）（）（）），



身高 ETL技术规范~~15~~ 4。

|  |  |
| --- | --- |
| 2005年11月 | 前端系统 |
| FeS | 中介装置 |
| 医学博士 | 移动电话号码便携性 |
| MNP | OM |
| OM | Oracle财务 |
| ORF | 预付费控制台 |
| PPC | 用户 |

USR

**来自不同源系统的数据文件应到达Sun Solaris临时区域。 它们被三种方式捕获:**

Sun Solaris

拉

推

提取物

**中转区** **2.1.1**

拉拔

**某些数据文件存在于FTP服务器或受信任的UNIX服务器中。 需要通过文件传输协议(FTP)或远程文件复制(RCP)获取它们。 ftp需要一对登录名和密码，而rcp只需要一个受信任的登录名。 拉拔将在预处理中进一步提及。** **2.1.2**

推送

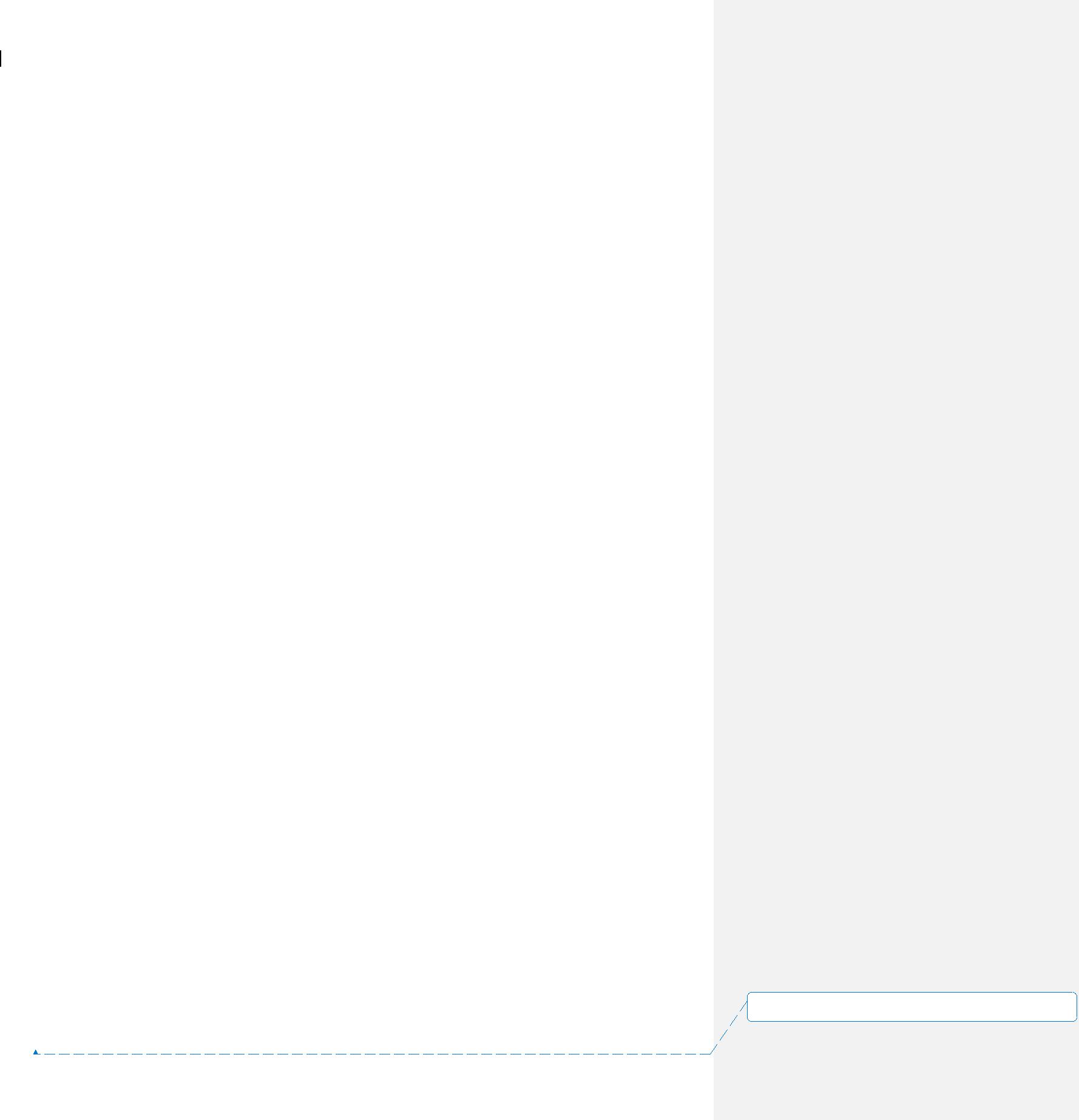
**这种方法主要用于来自中介设备的文件。 这些设备将文件推送到临时区域。** **2.1.3**

提取

源数据作为一个表或一组表存在于OLTP系统中。 需要一个提取作业来登录数据库以获取所需的数据。 提取作业~~0\_1.单据~~ SMC-V ADW项目-ETL技术规范。doc

佩奇

**46个中的8个** 大小:（）（）（）（）（）（）），



身高 ETL技术规范~~15~~ 4。

**2.2** **DATA FILES PRE-PROCESSING**

In order to capture data from all sources, there is a need to have a central process to manage different arrival patterns of data files. A pre-process module is designed to act as the primary data file handler. All data files should undergo it prior to ETL Automation loading. As this pre-process is required to support monitoring capability, this should be implemented as daemon program.

**2.2.1** **Typical Functionalities of the Pre-process Daemon**

The daemon provides the following basic functions:

* Monitor arrival of data files and control files
* Trigger file pulling and extraction sub-task on dedicated schedule
* Verify file by variance and determine if the file is ready to load
* Truncate and validate header/trailer for loading jobs
* Massage data files (E.g. Transpose, Decrypt)
* Generate standardized control files and data files for ETL Automation processing
* Log file statistics for calculating variance.

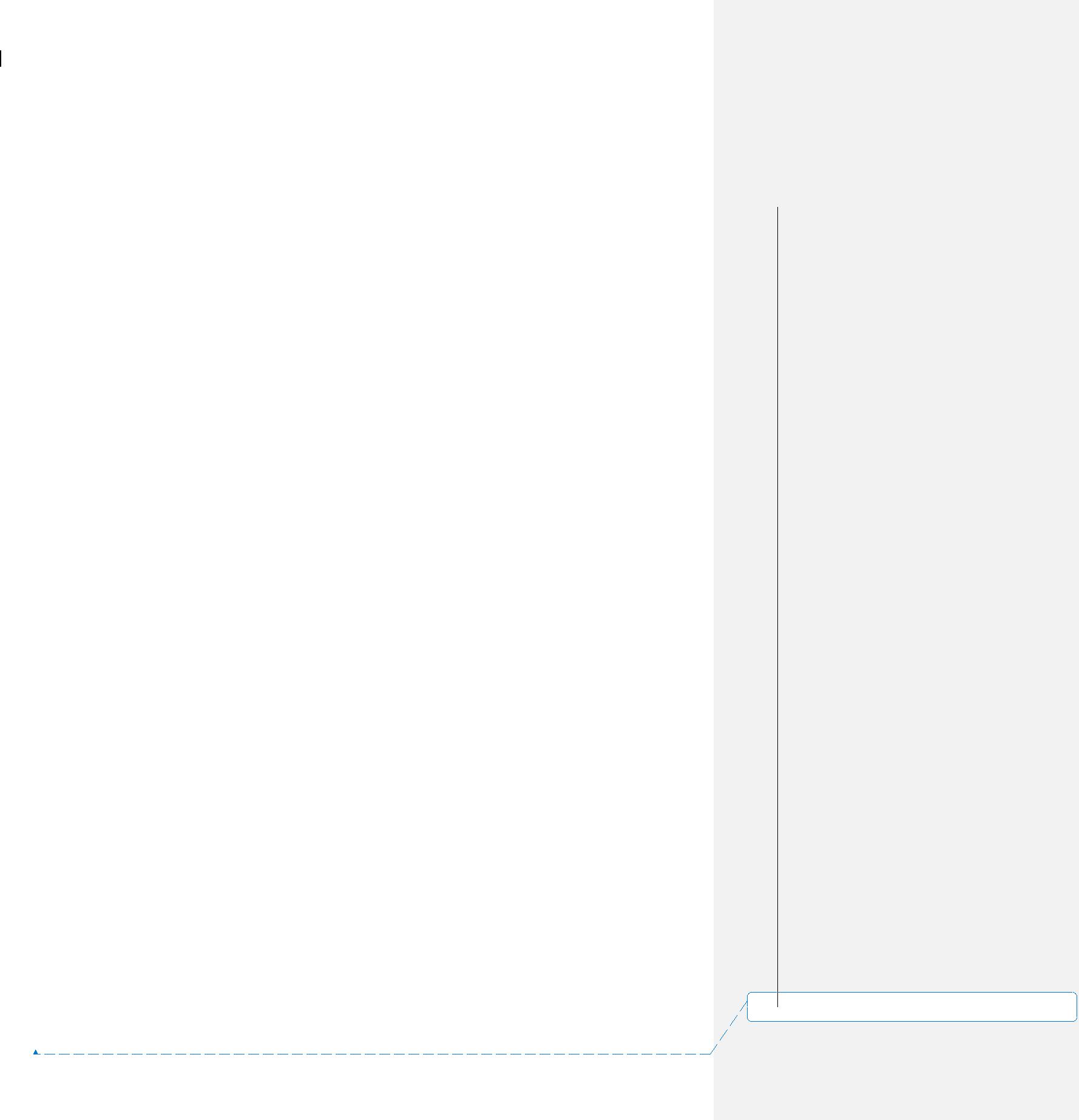
**2.2.2** **The Pre-process Configuration Tables**

Three tables will be used to define the source file characteristics and file checking rules. The pre-process daemon will handle the file according to the values of these attributes, and finally determine if the corresponding EA loading job can start, or generate necessary messages to notify relevant parties.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

9 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法

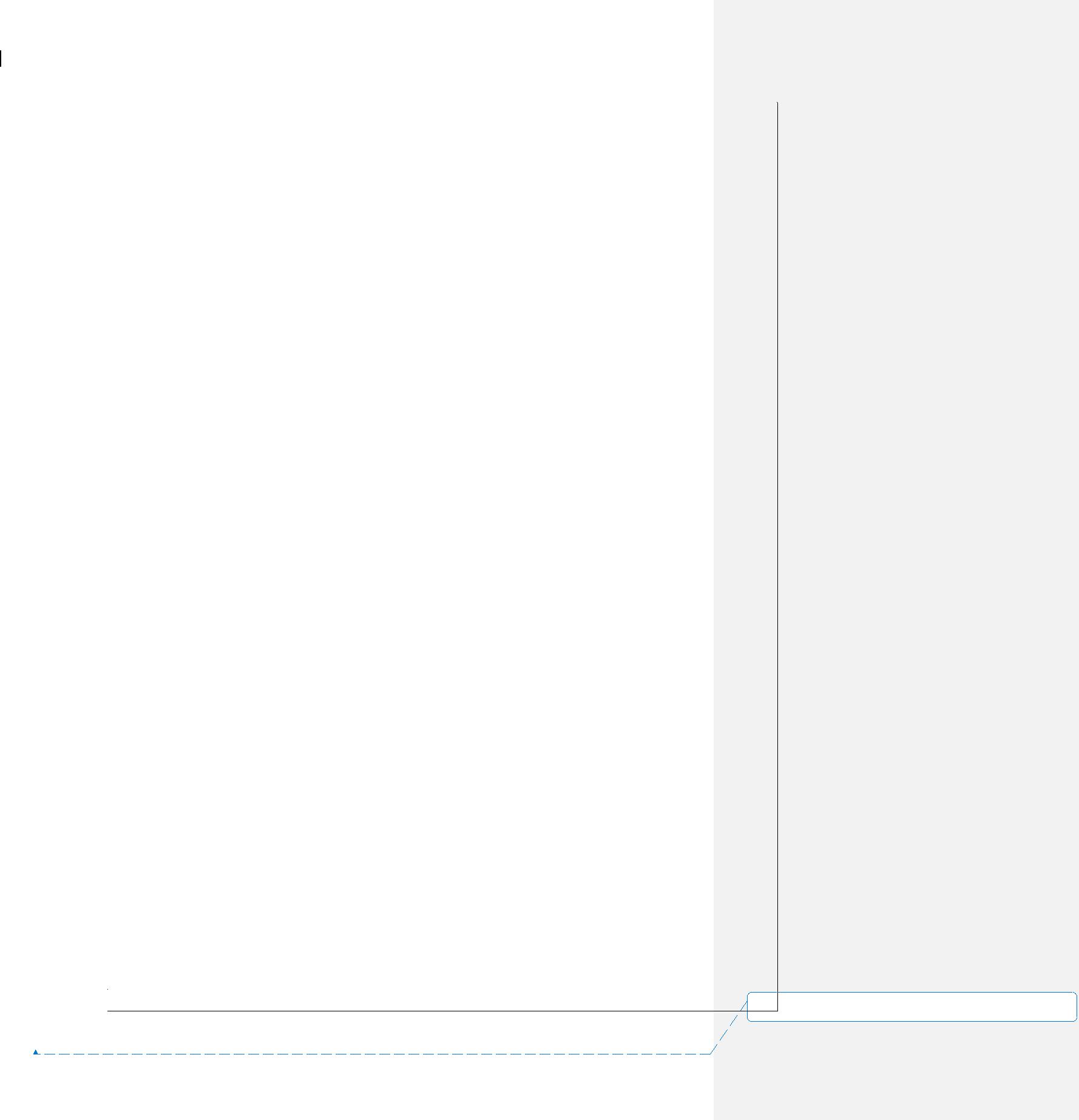


ETL Technical Specification 15~~4~~ November 2005

**Table name: ETL\_SRC\_FILE**

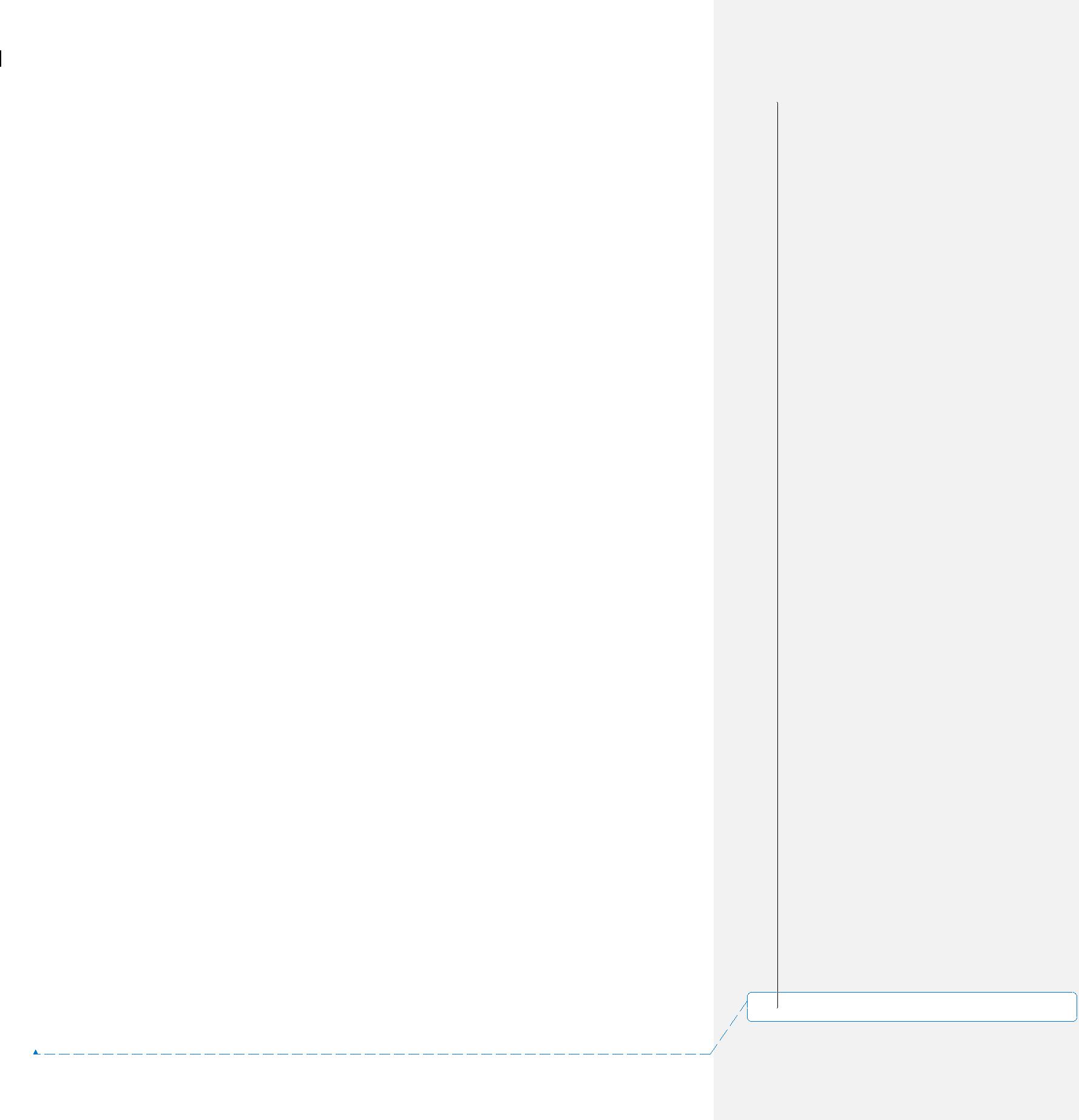
The table stores the attributes to tell the daemon how to handle data file arrival and call generic child module to pre-process the file case by case.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Attribute Name** | **Description** | **Format and Possible** |  |
|  |  |  |  | **Values** |  |
|  |  | Filename\_Mask | For pulling or pushing data source, it is the | E.g. |  |
|  |  |  | file name mask to match a set of files based |  |  |
|  |  |  | on the defined pattern. | [date].card.z |  |
|  |  |  | It allows the following 3 types of pattern | zone\_reg\_[date].txt |  |
|  |  |  | variables: |  |  |
|  |  |  |  | [date]\*.cli |  |
|  |  |  | [date] – date portion |  |  |
|  |  |  | \*– Multiple alphanumeric chars | billing\_[date].ready |  |
|  |  |  | [n] – One digit |  |  |
|  |  |  | [n][n] – Two digits | ds\_load\_rateplan |  |
|  |  |  | [n][n][n] – Three digits |  |  |
|  |  |  | [date] will be substituted by the system |  |  |
|  |  |  | date to match data files. This allows one |  |  |
|  |  |  | occurrence only. |  |  |
|  |  |  | \* is the wildcard for the data filename. This |  |  |
|  |  |  | allows one occurrence only. |  |  |
|  |  |  | [n] represents a digit that allows a number |  |  |
|  |  |  | of occurrences. |  |  |
|  |  |  | For Billing system, this attribute will |  |  |
|  |  |  | contain a control file name to specify |  |  |
|  |  |  | readiness of multiple file sets in the same |  |  |
|  |  |  | remote directory. |  |  |
|  |  |  | For data source extracted directly from |  |  |
|  |  |  | database, it is the DataStage job name. |  |  |
|  |  |  | These jobs are located in a predefined |  |  |
|  |  |  | directory. |  |  |
|  |  | Filename\_Date\_For | It is the argument used in Unix date | E.g. |  |
|  |  | mat | command to tell how the variable date |  |  |
|  |  |  | portion “[date]”in the Filename\_Mask is | %Y%m%d |  |
|  |  |  | substituted by a system date. | **设置了格式:** (中文) (不作校对), (其他) (不作校对),不检 | |
|  |  |  |  |  |  |
|  |  |  |  | 查拼写或语法 | |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | | | Page | |
| 10 of 46 | | |  |  |  |



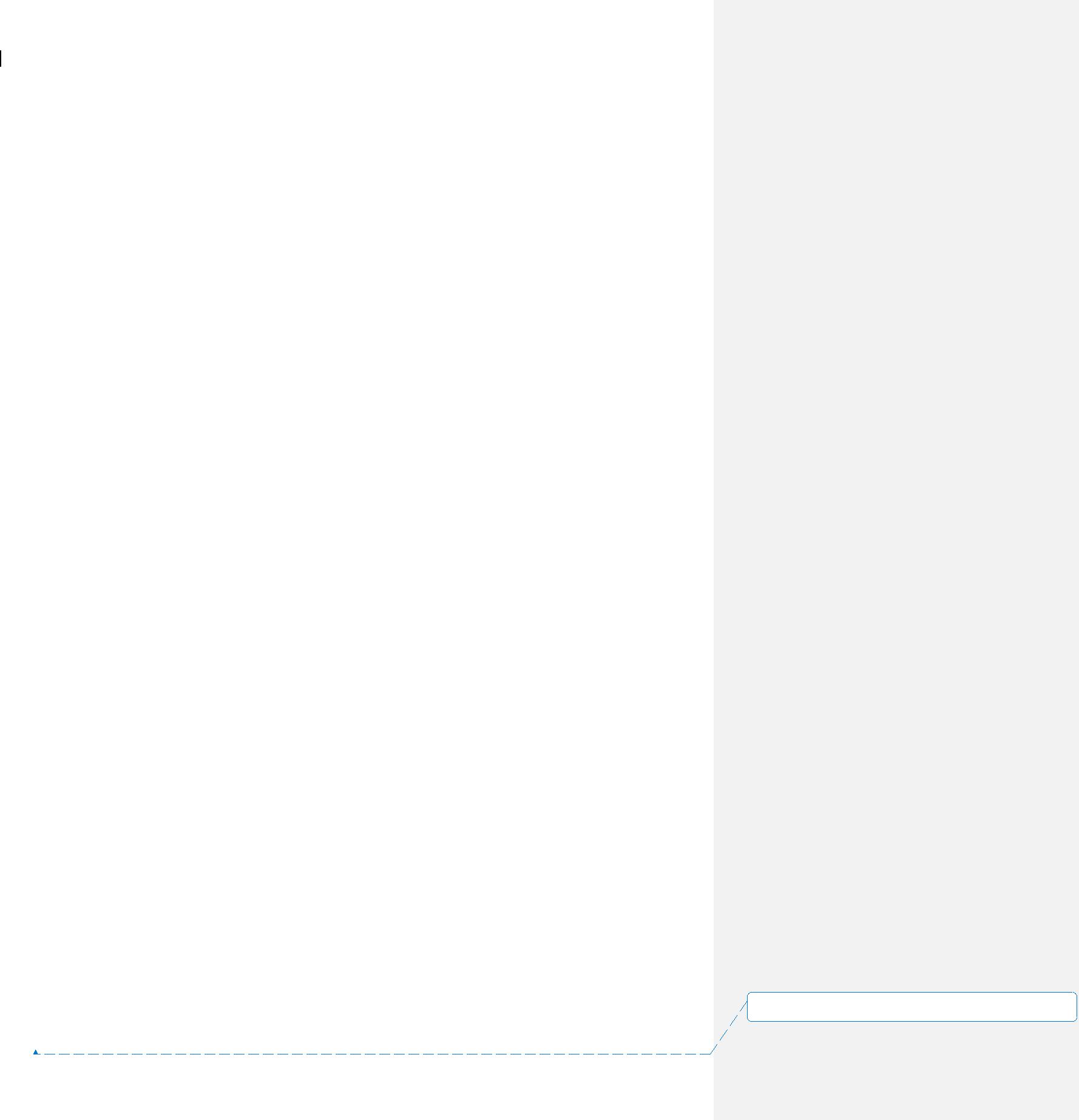
ETL Technical Specification 15~~4~~ November 2005

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
|  |  | Data\_Path | Directory location for the data files | | E.g. |  |  |  |
|  |  |  | For pulling, data files are pulled from this | | /opt/etl/data/arrive/fes |  |  |  |
|  |  |  | directory | |  |  |  |  |
|  |  |  | For extracting, data is extracted to put on | |  |  |  |  |
|  |  |  | this directory | |  |  |  |  |
|  |  |  | For pushing, this is the monitoring | |  |  |  |  |
|  |  |  | directory | |  |  |  |  |
|  |  |  |  | |  |  |  |  |
|  |  | Job\_Name | The name used to derive data file name | | E.g. md\_loadcdr |  |  |  |
|  |  |  | and control file name for ETL Automation. | |  |  |  |  |
|  |  |  | The Job\_Name should compose of a | |  |  |  |  |
|  |  |  | meaningful name related to the table and | |  |  |  |  |
|  |  |  | its data source abbreviation at the | |  |  |  |  |
|  |  |  | beginning. | |  |  |  |  |
|  |  |  | If this field is null, no control file and data | |  |  |  |  |
|  |  |  | file will be generated to EA. This is the case | |  |  |  |  |
|  |  |  | for special condition, that Billing System | |  |  |  |  |
|  |  |  | source is putting control file to indicate | |  |  |  |  |
|  |  |  | readiness for a set of data files. Please refer | |  |  |  |  |
|  |  |  | to the Using Special Handling Script for | |  |  |  |  |
|  |  |  | details. | |  |  |  |  |
|  |  | Date\_Offset | The number of offset days used to make | | 0 |  |  |  |
|  |  |  | adjustment on the system date for locating | |  |  |  |  |
|  |  |  | valid data files. Sign is meaningful to this | |  |  |  |  |
|  |  |  | field. | |  |  |  |  |
|  |  | File\_Locate\_Type | Method of locating the data files | | PF – Pull FTP |  |  |  |
|  |  |  |  |  | PR – Pull RCP |  |  |  |
|  |  |  |  |  | PS – Push |  |  |  |
|  |  |  |  |  | EX – Extract |  |  |  |
|  |  |  |  |  | CP – Copy |  |  |  |
|  |  | Special\_Handling\_ | The file name of specific special handling | | dw\_load\_ringcity.pl |  |  |  |
|  |  | Script | script required to run. (E.g. transpose, | | bs\_ftp\_billing.ksh |  |  |  |
|  |  |  | decrypt, header-trailer validation). | |  |  |  |  |
|  |  |  |  | |  |  |  |  |
|  |  | Pull\_Ip\_Address | IP address or DNS name for locating the | | 172.20.252.100 |  |  |  |
|  |  |  | file for ftp and rcp pulling | |  |  |  |  |
|  |  | Pull\_Login | Login user id used to get file via ftp or rcp | | adwbat |  |  |  |
|  |  | Pull\_FTP\_Port | Special port required for FTP other than | | 6006 | **设置了格式:** (中文) (不作校对), (其他) (不作校对),不检 | |  |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | | |  | Page | 查拼写或语法 | |  |
|  |  |  |  |
| 11 of 46 | | |  |  |  |  |  |  |



ETL Technical Specification 15~~4~~ November 2005

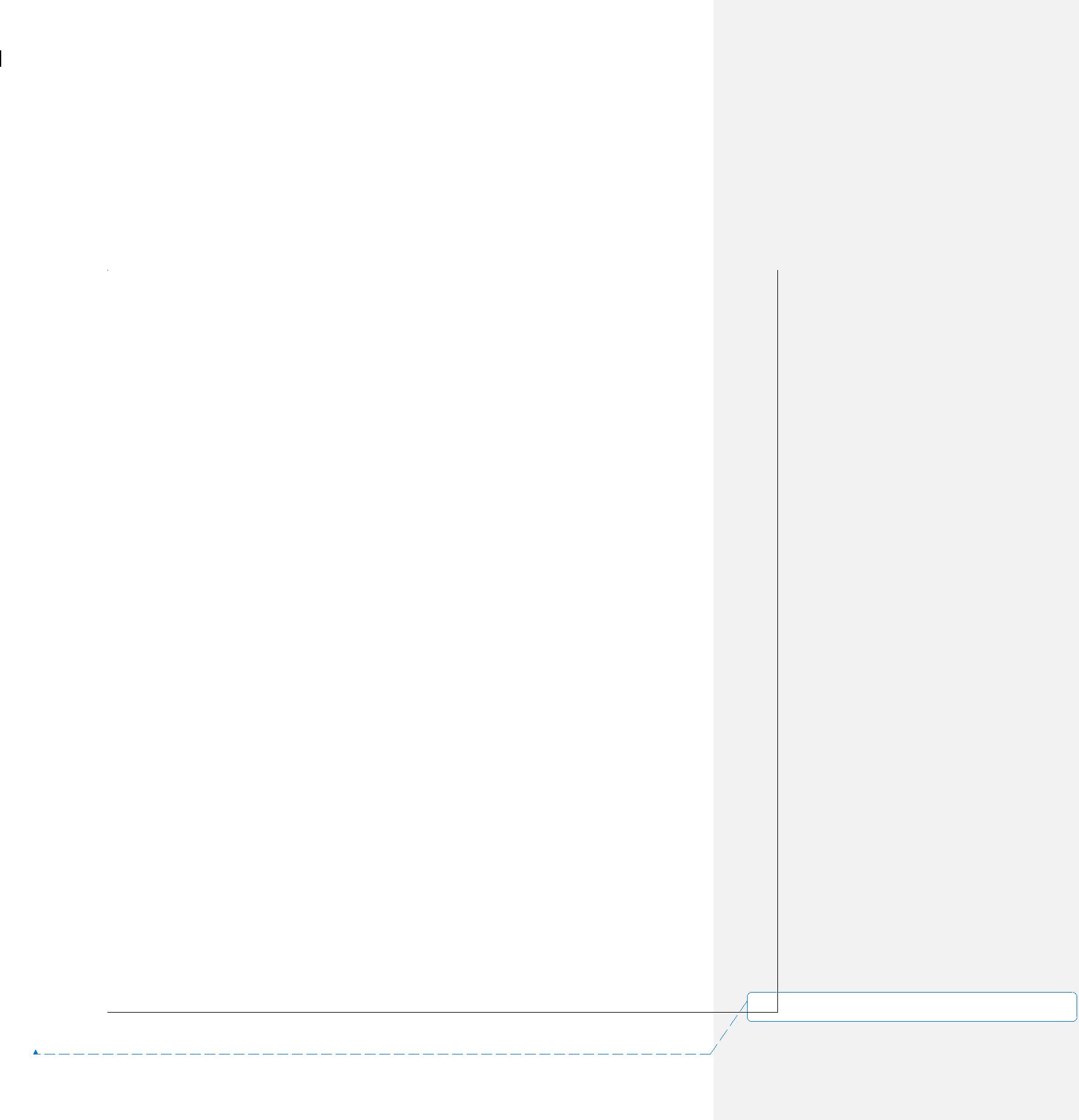
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | standard port |  |  |  |
|  |  | Pull\_FTP\_Mode | File transfer mode | B – Binary |  |  |
|  |  |  |  | A – ASCII |  |  |
|  |  | Pull\_Remote\_Path | Remote directory storing the data files to | /log/logbak/CDR/[date] |  |  |
|  |  |  | pull. |  |  |  |
|  |  |  |  | /xera/warranty |  |  |
|  |  |  | This value should NOT contain “/” at the |  |  |  |
|  |  |  | end. |  |  |  |
|  |  |  | If “[date]” exists in the path, it will be |  |  |  |
|  |  |  | replaced by system date according to the |  |  |  |
|  |  |  | format specified in |  |  |  |
|  |  |  | Remote\_Path\_Date\_Format |  |  |  |
|  |  |  |  |  |  |  |
|  |  | Remote\_Path\_Date | It is the argument used in Unix date | %Y%m%d |  |  |
|  |  | \_Format | command to tell how the variable date |  |  |  |
|  |  |  | portion “[date]”in Pull\_Remote\_Path is |  |  |  |
|  |  |  | substituted by a system date. |  |  |  |
|  |  |  |  |  |  |  |
|  |  | Pull\_Start\_Date\_Lis | It is a list of execution frequency date codes | 6:1:\*,6:4:\*,6:7:\*,6:10:\* |  |  |
|  |  | t | controlling data file pulling. Each |  |  |  |
|  |  |  | frequency code is delimited by a comma. |  |  |  |
|  |  |  | Pulling starts if any one code is matched. |  |  |  |
|  |  |  | The format is: |  |  |  |
|  |  |  | <Freq\_Code1>,<Freq\_Code2> |  |  |  |
|  |  |  | A frequency code is comprised of 3 parts. |  |  |  |
|  |  |  | Each part is delimited by a colon, and |  |  |  |
|  |  |  | follows the date specification of crontab in |  |  |  |
|  |  |  | Unix. |  |  |  |
|  |  |  | The specification is: |  |  |  |
|  |  |  | <Day in month>:<Month>:<Week day> |  |  |  |
|  |  |  | Each part allows an asterisk to represent a |  |  |  |
|  |  |  | wildcard. |  |  |  |
|  |  |  | E.g. |  |  |  |
|  |  |  | \*:\*:\* means pulling file everyday |  |  |  |
|  |  |  | 30:\*:\* means pulling file on 30th every |  |  |  |
|  |  |  | month. (Note that February won’t have 30th |  |  |  |
|  |  |  | day, so there will not be February run) |  |  |  |
|  |  |  |  | **设置了格式:** (中文) (不作校对), (其他) (不作校对),不检 | | |
|  |  |  |  |  |  | |
|  |  |  |  | 查拼写或语法 | | |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | | | Page | | |
| 12 of 46 | | |  |  |  |  |



ETL Technical Specification 15~~4~~ November 2005

|  |  |  |
| --- | --- | --- |
|  | 6:1:\*,6:4:\*,6:7:\*,6:10:\* represents a file pull |  |
|  | on the 6th day of Jan, Apr, Jul and Oct. |  |
| Pull\_Start\_Time\_Li | It is a list of expected time in HH:MM 24 | 01:00,13:00 |
| st | hour format up to minute. Hour part and |  |
|  | minute part is delimited by a colon. |  |
|  | This will be used together with |  |
|  | Pull\_Start\_Freq\_List to determine if the file |  |
|  | needs to pull. |  |
| Pull\_Retry\_Count | The number of retry allowed before an | 5 |
|  | error message is sent to HP Openview |  |
| Retry\_Wait\_Minute | The wait time in minutes to wait before the | 10 |
|  | next retry. |  |
| Data\_File\_Retentio | The number of days where data files are | 7 |
| n\_Period | kept. Data files older than this number of |  |
|  | days will be cleaned up. |  |
| Unzip\_Command | The command with argument to be called | gunzip %1 –c > %2 |
|  | to unzip a file. |  |
|  | Complex combination of different unzip |  |
|  | types (E.g. gunzip, tar, zcat) will be done |  |
|  | by the script specified in |  |
|  | Special\_Handling\_Script field. |  |
| Check\_EA\_Fail\_Dir | Y – The pre-process will check for previous | Y |
| \_Flag | job failure in EA Fail directory before |  |
|  | sending control file to EA. If control file is |  |
|  | found there, pre-process does not trigger |  |
|  | the EA loading job. It will then wait for |  |
|  | minutes in Retry\_Wait\_Minute and retry. |  |
|  | N – The pre-process does not check EA Fail |  |
|  | directory. |  |
|  |  |  |
| As\_Of\_Date\_Deter | The way to determine the data As\_Of\_Date | N – From file name itself |
| mine\_Method |  | T – From file timestamp |
| Last\_Process\_Time | System date time for last processing on this | 2005-10-24 10:12:25 |
|  | file set |  |
| Last\_Process\_Statu | The status for last processing | S - Success |
| s |  | F – Fail |
|  |  | P - Processing |
| Last\_File\_As\_Of\_D | The last valid file As\_Of\_Date determined | 2005-10-24 00:00:00 |
| ate | during file pre-process |  |

|  |  |  |
| --- | --- | --- |
|  |  | **设置了格式:** (中文) (不作校对), (其他) (不作校对),不检 |
|  |  | 查拼写或语法 |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 13 of 46 | |  |

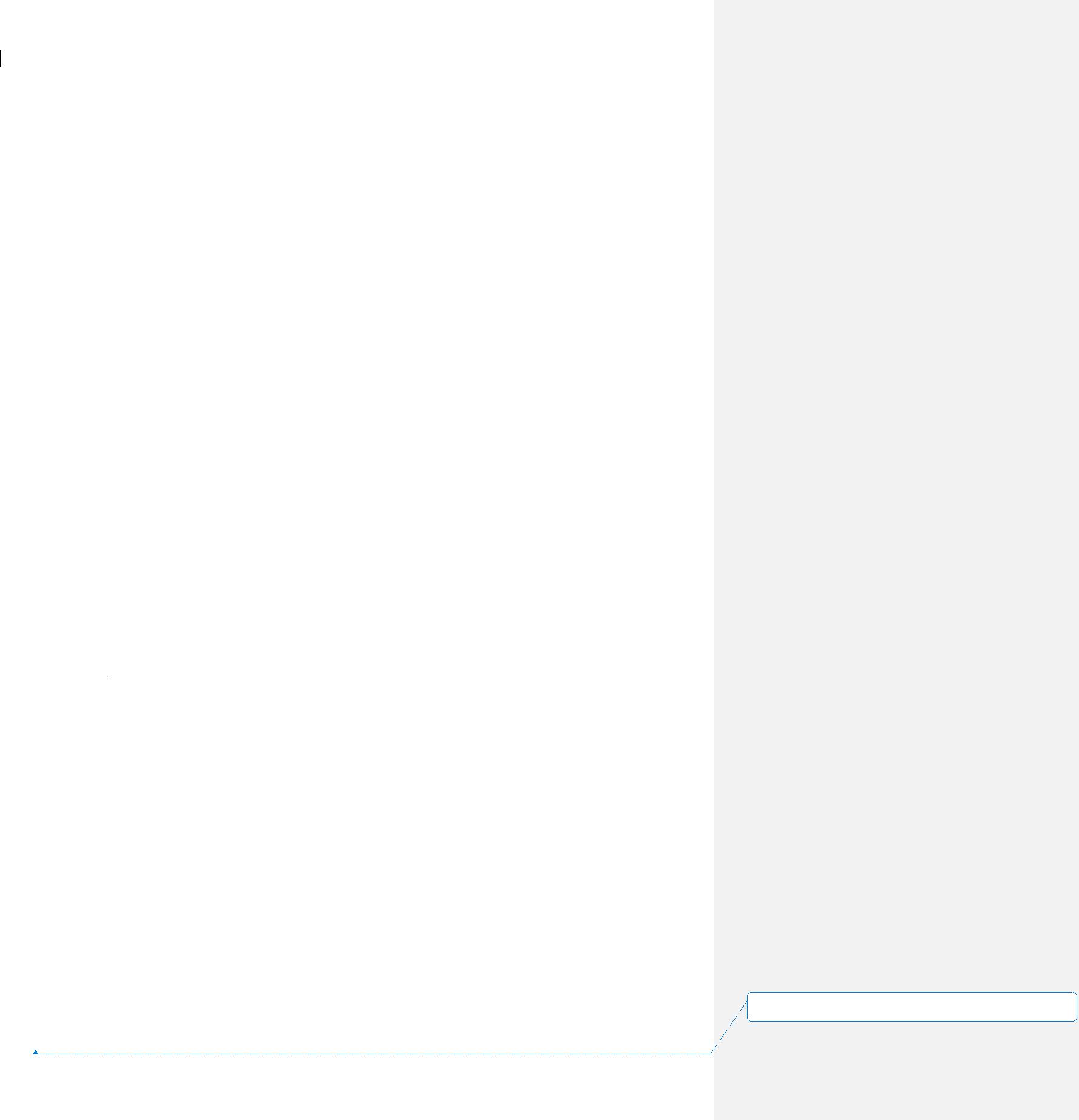


ETL Technical Specification 15~~4~~ November 2005

**Table name: ETL\_FILE\_VALIDATION\_RULE**

Per each data file set, checking rules can be defined to validate it. Each rule is designed to generate a file validation failure message if its condition is fulfilled. In short, all rules are OR’ed to determine the file validity.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Attribute** | **Description** | **Possible Value** |  |
|  |  | Filename\_Mask | This is foreign key to | Refer to foreign key |  |
|  |  |  | ETL\_SOURCE\_FILE.Filena | ETL\_SOURCE\_FILE.Filename\_Mask |  |
|  |  |  | me\_Mask |  |  |
|  |  | Data\_Path | This is foreign key to | Refer to foreign key |  |
|  |  |  | ETL\_SOURCE\_FILE.Data\_ | ETL\_SOURCE\_FILE.Data\_Path |  |
|  |  |  | Path |  |  |
|  |  | Check\_Type | Types of checking rules | RC – File record count |  |
|  |  |  |  | FC – File count |  |
|  |  |  |  | FS – File size in bytes |  |
|  |  |  |  | DV – Data Variance |  |
|  |  |  |  | RR – Rejected record sent to error tables |  |
|  |  |  |  | after loading |  |
|  |  |  |  | OC – Output Record Count |  |
|  |  | Percentage\_Fixed\_ | The value used for the | F - Fixed Value |  |
|  |  | Value | checking | P – Percentage |  |
|  |  |  |  | A – Both Fixed Value and Percentage are |  |
|  |  |  |  | checked |  |
|  |  | Rule\_Activate\_Flag | The flag to activate or | Y – This rule is activated and in effect |  |
|  |  |  | deactivate this rule | N – This rule is deactivated |  |
|  |  | DV\_Check\_Field\_ | The numeric field that | Total\_Call\_Minute |  |
|  |  | Name | needs to impose |  |  |
|  |  |  | aggregation check. |  |  |
|  |  | DV\_Aggr\_Func | The database function used | AVG |  |
|  |  |  | to aggregate the checking | SUM |  |
|  |  |  | field | MAX |  |
|  |  |  |  | MIN |  |
|  |  | Min\_Margin | The min margin used for | 8,000 |  |
|  |  |  | fixed value rule checking |  |  |
|  |  | Max\_Margin | The max margin used for | 10,000 |  |
|  |  |  | fixed value rule checking |  |  |
|  |  | Past\_ | The number of times of | 7 |  |
|  |  | Variance\_Use\_Cou | available variance |  |  |
|  |  | nt | information used for |  |  |
|  |  |  | checking variance by | **设置了格式:** (中文) (不作校对), (其他) (不作校对),不检 | |
|  |  |  |  | 查拼写或语法 | |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | | | Page | |
| 14 of 46 | | |  |  |  |



ETL Technical Specification 15~~4~~ November 2005

|  |  |  |
| --- | --- | --- |
|  | percentage. |  |
| Margin\_Percentage | The positive and negative | 5 |
|  | margin percentage used to |  |
|  | validate the data file based |  |
|  | on past loading history. |  |
|  | 5 implies + - 5% |  |

**Table name: ETL\_FILE\_VALIDATION\_HIST**

This is the history table to store file validation statistics. The content will be used to calculate variance value.

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Description** | **Possible Value** |
| Filename\_Mask | This is foreign key to | Refer to foreign key |
|  | ETL\_FILE\_VALIDATION\_ | ETL\_FILE\_VALIDATION\_RULE. |
|  | RULE.Filename\_Mask | Filename\_Mask |
| Data\_Path | This is foreign key to | Refer to foreign key |
|  | ETL\_FILE\_VALIDATION\_ | ETL\_FILE\_VALIDATION\_RULE. |
|  | RULE.Data\_Path | Data\_Path |
| Check\_Type | This is foreign key to | Refer to foreign key |
|  | ETL\_FILE\_VALIDATION\_ | ETL\_FILE\_VALIDATION\_RULE. |
|  | RULE.Check\_Type | Check\_Type |
| Check\_Timestamp | The timestamp performing | 2005-09-21 14:00:00 |
|  | the checking |  |
| Table\_Name\_List | List of the final ADW | subr\_info\_hist, cust\_info\_hist |
|  | table(s) name where this |  |
|  | file goes to. It serves for |  |
|  | troubleshooting purpose. |  |
| Result\_Value | Result for the check type | 100 |
|  | on the data file at a time |  |

**2.2.3** **Data File Wildcard Matching**

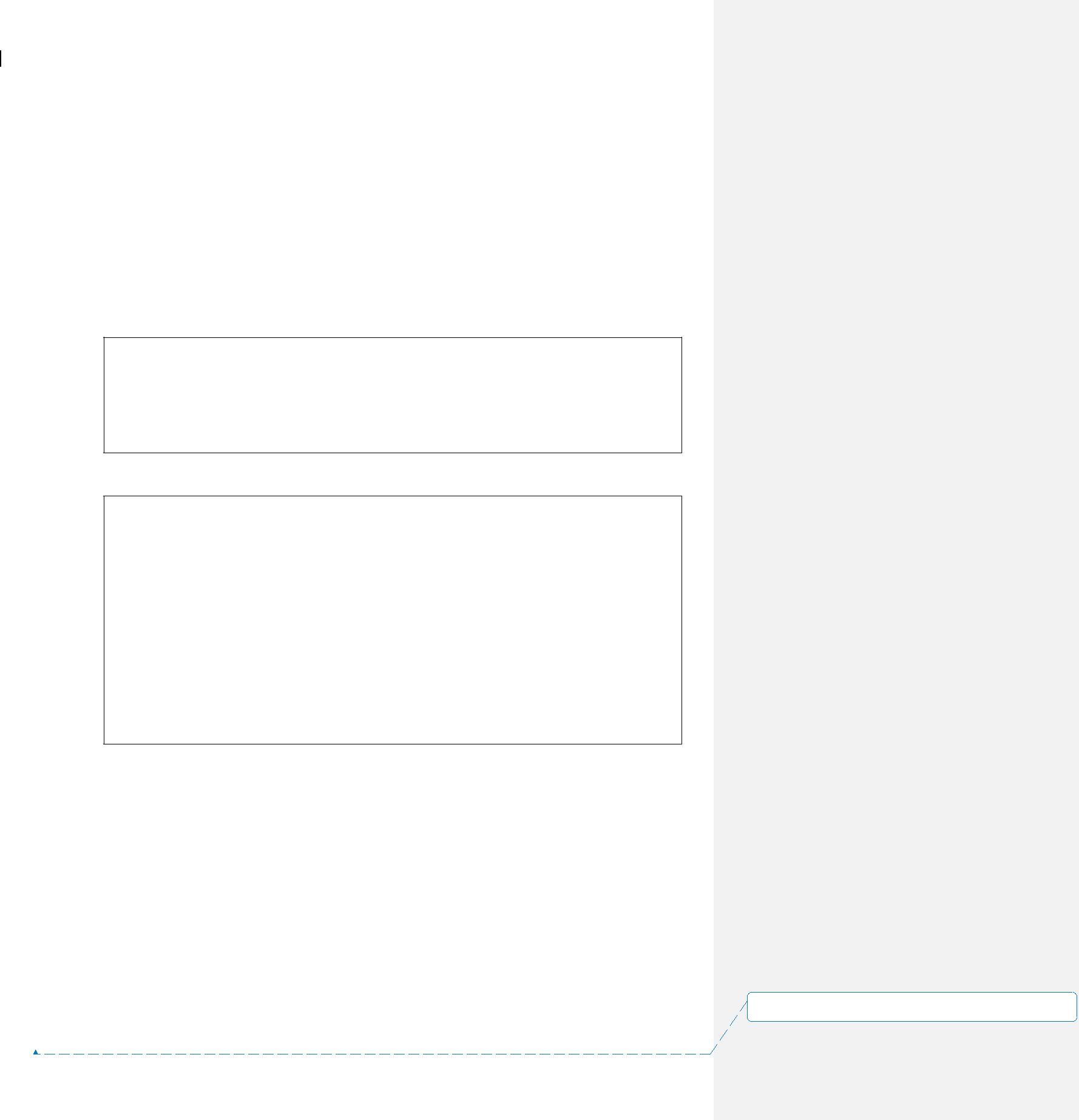
The following describes how to get candidate data files:

1. The system date is captured.
2. It is then adjusted by its offset value. The offset will have a default zero value. In case of running job to cater earlier data files, it is set to a negative value to adjust the file matching date.
3. The adjusted date is fitted into the date portion in the file name mask, according to its format specified in Filename\_Date\_Format. In short, [date] is replaced by the adjusted system date.
4. The filename mask is further processed by replacing every [n] by [0-9].
5. The final string is a regular expression for Unix “ls” and “grep” to match files.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

15 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

**2.2.4** **Control File and Data File Structure**

A control file indicates the completeness and readiness of its set of data files. It is a text file with standardized filename and file format generated by the pre-process daemon. It is used by ETL Automation for job triggering and dependency control. When a control file is tracked, ETL Automation will read its content and compare the physical file size of the data file. If file size is matching and no other dependency is pending, the job is executed.

The control filename is generated as the format below:

Format:

dir.<Data\_Source>\_<Job\_Name>\_<YYYYMMDD>

E.g.

dir.md\_loadgprs\_20051024

Each row in the control file follows the structure below:

Format:

<Job\_Name>\_<Orig\_Filename>\_<YYYYMMDD>

E.g.

Assume that:

Orig\_Filename is gprs\_20051024

Job\_Name is md\_loadgprs

System date is 20051024

Control file will contain:

md\_loadgprs\_gprs\_20051024.dat\_20051024

In which:

* <Data\_Source> is the 2-3 chars abbreviation code of data sources
* <Job\_Name> is job name for the loading job
* <Orig\_Filename> is the original filename during wildcard matching
* <YYYYMMDD> is derived from system date and adjusted by an offset.

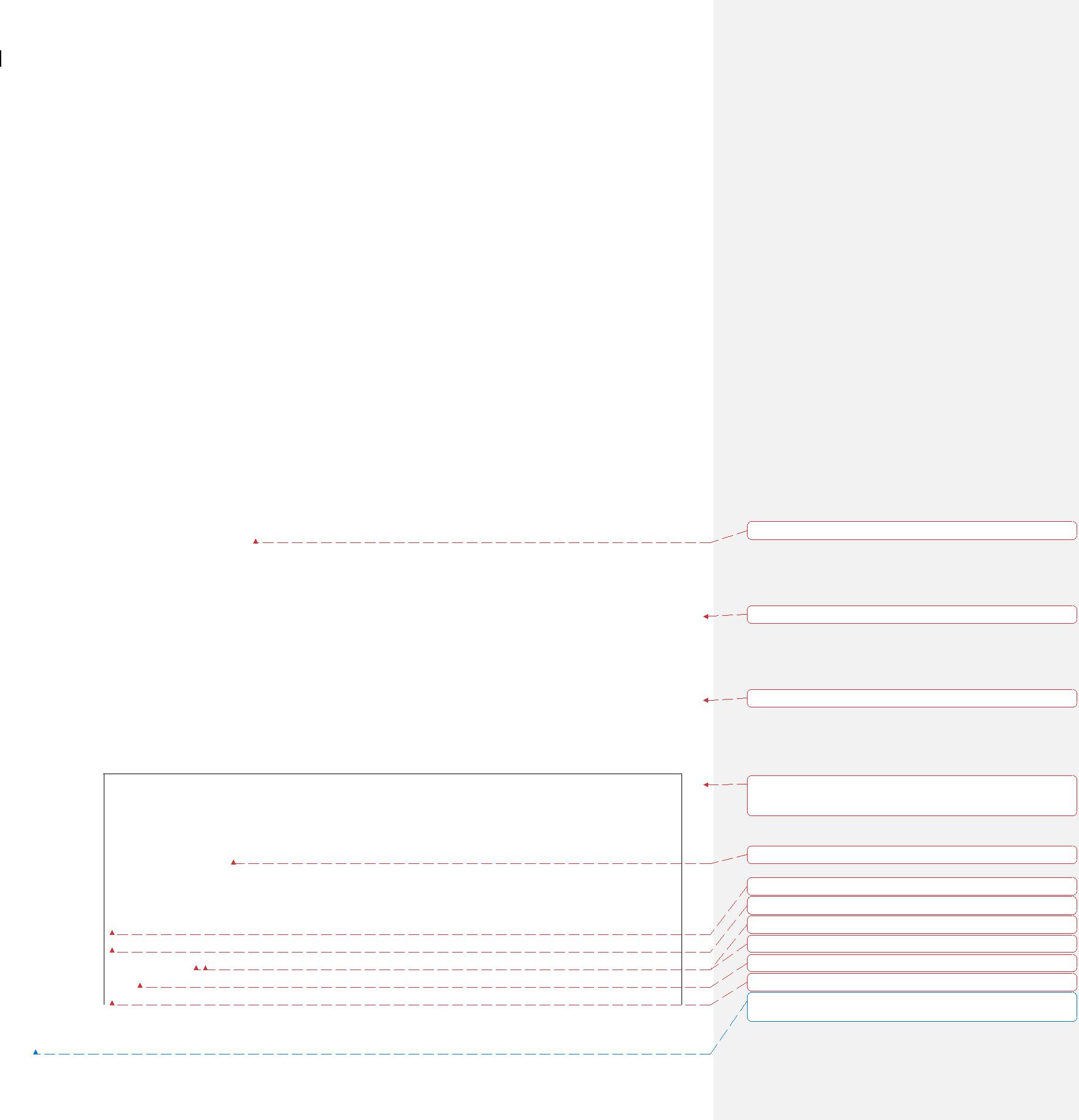
**2.2.5** **Using Special Handling Script**

The value in Special\_Handling\_Script column provides the name to call a shell script. This can solve specific problems and provide high level of flexibility.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

16 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

Currently, it is used in 2 areas.

**Perform Specific File Processing**

It involves transpose, decrypt, header trailer removal, complex unzip, and Excel file massage. Individual scripts are called to handle the tasks.

**Control File Generated from Data Source**

Billing system provides data once a day. A control is placed when all data files are ready. However, the data files may be of different masks and should be processed separately.

In order to handle this case, an extra row is added to the pre-process configuration table. This row contains the specific control file name as in the Billing system directory. When the control file for the date is found, it triggers a special script to copy all candidate files for today to different local sub-directories. The files will be copied firstly to a temp sub-directory and then renamed finally to make the process atomic. Once the directory is renamed, it will further trigger corresponding pre-processes. The Job\_Name attribute of this special row is set to null to avoid the generation of EA control file.

**Parameter Passing Schematic for Special Handling Script**

There are 2 arguments passed to the special handling script:

1. IN\_DATA\_PATH

This variable tells the script the full path location of input data files. The special handling script should treat all files there as candidates for its processing.

2. OUT\_DATA\_PATH

The special handling script should place its output files here.

Calling Example:

...

cd /opt/etl/MD/work/20051115.10901/IN

$SCRIPT\_PATH/special.ksh /opt/etl/MD/work/20051115.10901/IN /opt/etl/MD/work/20051115.10901/OUT

wait

rc=$?

if [ $rc –eq 0 ]; then

...

else

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 17 of 46 | |  |

**设置了格式:** 字体: 加粗

**带格式的:** 项目符号和编号

**带格式的:** 项目符号和编号

**带格式的:** 左,缩进: 悬挂缩进:0.5字符, 左 4.91 字符,

首行缩进: -0.5 字符, 边框:方框: (单实线, 自动设置, 0.5 磅 行宽)

**设置了格式:** 字体:Courier New

**设置了格式:** 字体:Courier New

**设置了格式:** 字体:Courier New

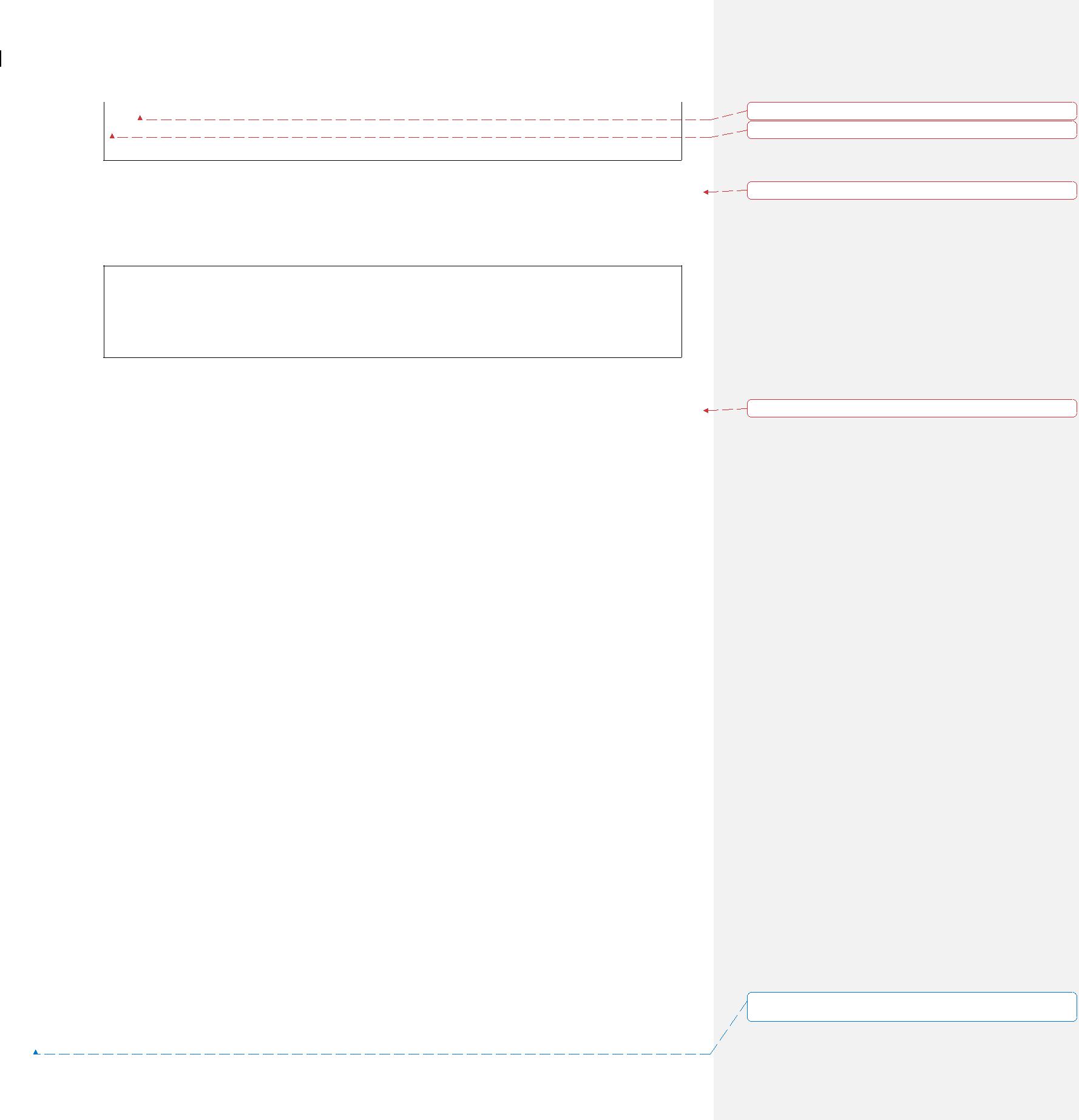
**设置了格式:** 字体:Courier New

**设置了格式:** 字体:Courier New

**设置了格式:** 字体:Courier New

**设置了格式:** 字体:Courier New

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

...

fi

The script should be placed at “~/preprocess/<Data\_source>/script “ directory.

At the beginning of the special handling script, it can include the following 2 lines to get the arguments:

IN\_DATA\_PATH=$1

OUT\_DATA\_PATH=$2

The special handling script should properly return its exit code.

|  |  |  |
| --- | --- | --- |
| ♦ | = 0 | Success |
| ♦ | <> 0 | Failure |

**2.2.6** **Restartable Pulling Process**

The pulling process of large files often takes a long period of time. The chance of aborted ftp, rcp, and copy cannot be ignored. Regarding this, the pulling mechanism is designed to be able to restart at its failure point.

A pulling will perform the following:

* A token file is touched at the beginning of pulling to indicate it is in-action.
* Then, pulling will run
* The token is finally cleaned up if the process completes successfully

If a pulling is aborted in the middle, the daemon will observe that an orphan file token remains there but have no corresponding process running. The daemon will restart the entire pulling process. The restarted pulling process will diff the source and target directories by file name and file size first to ensure completed files are not re-transmitted.

**2.2.7** **Loading Variance Factor – Checking Type**

The following measurement figures are required to check for loading variance:

* Data File Size – The size of data file in bytes is verified with the history size of the same file.
* Data File Count – The number of files in a set to load.
* Data File Record Count – The number of record in the file is verified with the historical record counts of the same file.
* Data Variance – The aggregated value (I.e. SUM, AVG, MAX and MIN) of

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

18 of 46

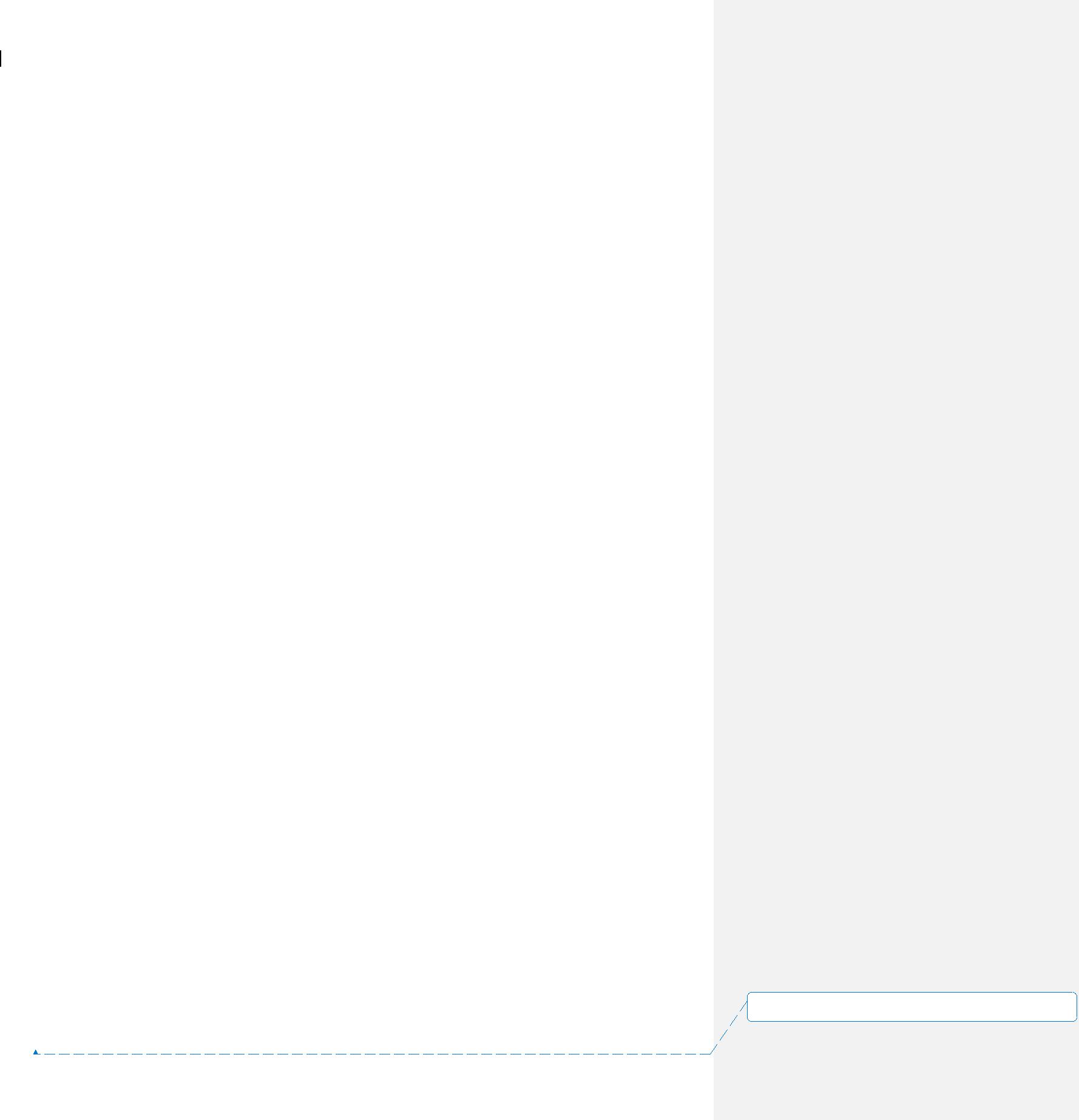
**设置了格式:** 字体:Courier New

**设置了格式:** 字体:Courier New

**带格式的:** 左

**带格式的:** 项目符号和编号

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

a chosen numerical field from the loaded table is verified against its previous history.

* Rejected Record Count – This means the total record count as found in error tables (I.e. E1 and E2 tables for FastLoad, ET and UV tables for MultiLoad) during loading.
* Output Record Count – The number of successfully loaded records.

Data file size, file count and file record count are checked during pre-process stage as the Input Variance Checking.

Data variance, rejected record count and output record count are checked during post-ETL process as the Output Variance Checking.

**2.2.8** **Loading Variance Checking Mechanism**

There are two checking mechanisms used:

* By a fixed margin value range
* By a margin percentage, calculated based on past n days of history

**Exception for Variance Checking**

If a data file is rejected due to variance exceeded, its measurement figures (I.e. File Size, Record Count and Numeric Field Aggregate) will NOT be inserted to the ETL\_FILE\_VALIDATION\_HIST table. Hence, it is NOT used for calculating the variance for the next time. This can avoid checking variance mechanism being affected by any bad or fluctuating figures.

If the information in the ETL\_FILE\_VALIDATION\_HIST table does not have enough history, the average value will still be calculated. For example, there is only three days of history but it is required to take average of seven days. In this case, average of the three days will be used as the variance checker.

**2.2.9** **Programming Languages**

The daemon will be written mainly using KornShell script. C language may be used whenever required to implement small modules.

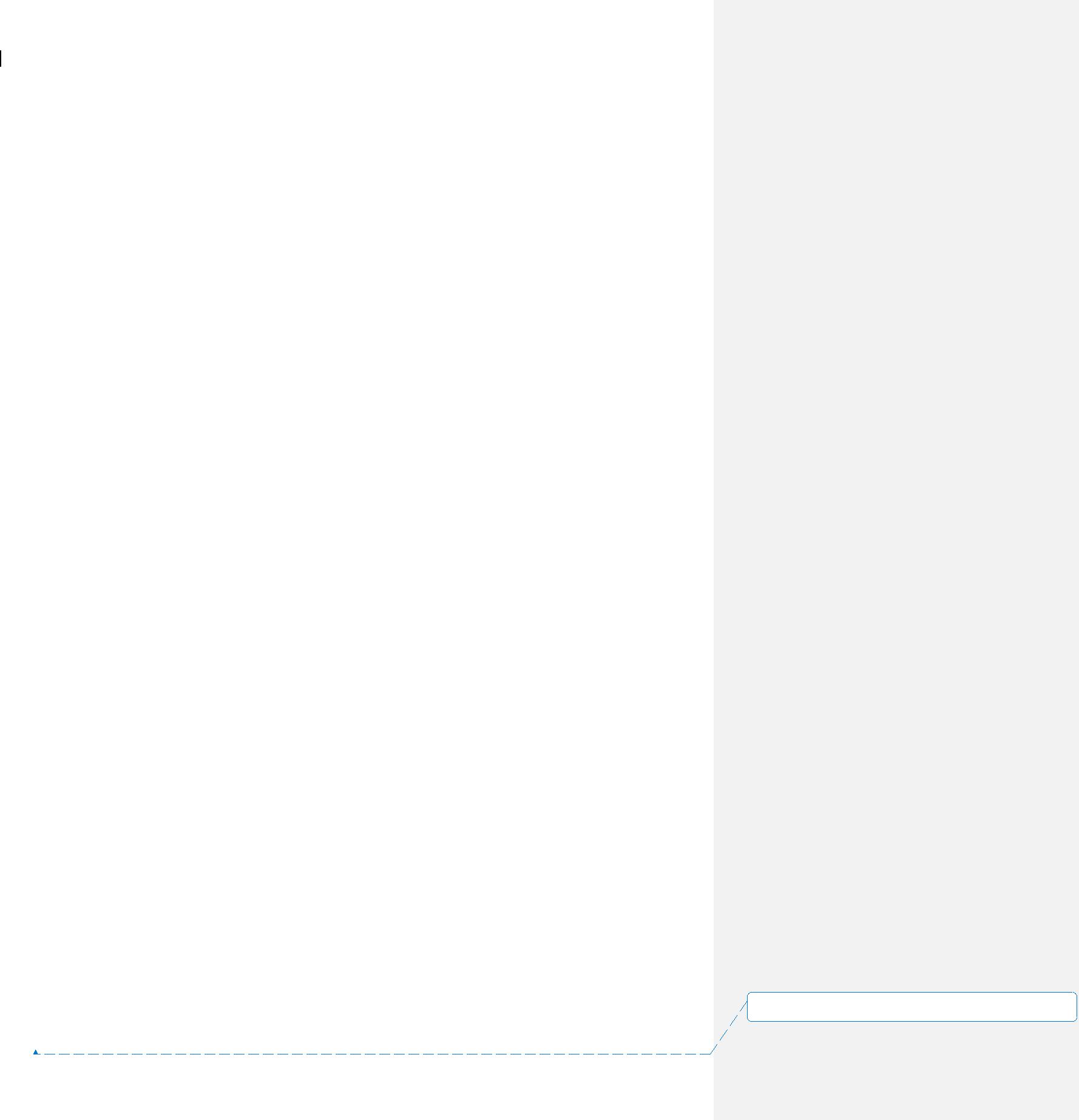
**2.2.10 Process Flow Design**

The pre-process mechanism is implemented as daemon, namely the Pre-process daemon.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

19 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

2.2.10.1 Pre-process Daemon

The Pre-process Loop Body

The daemon will firstly look for the exit file token (exit.token). If it is found, it terminates. Otherwise, it enters a loop body.

1. The file information from configuration table is captured.
2. It will firstly check the special rerun directories for file and call File Processor sub-process to handle it.
3. Then, it repeatedly forks sub-processes for the right candidates to perform pulling or extracting.
4. Then, it repeatedly checks for each monitoring directory. If any file found, it forks a File Processor sub-process to handle it.

After a sequence of these steps complete, it sleeps for a predefined number of minutes and repeats from the exit file token checking.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

20 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法

ETL Technical Specification

Sleep

E.g. 10 mins

No

No

Start

Exit file token

found?

No

Read

ETL\_Source\_File

table

Check rerun

directory

rerun directory

contains file?

No

Any more trigger

child to fork?

No

More directory to

monitor for pushed

files?

Yes

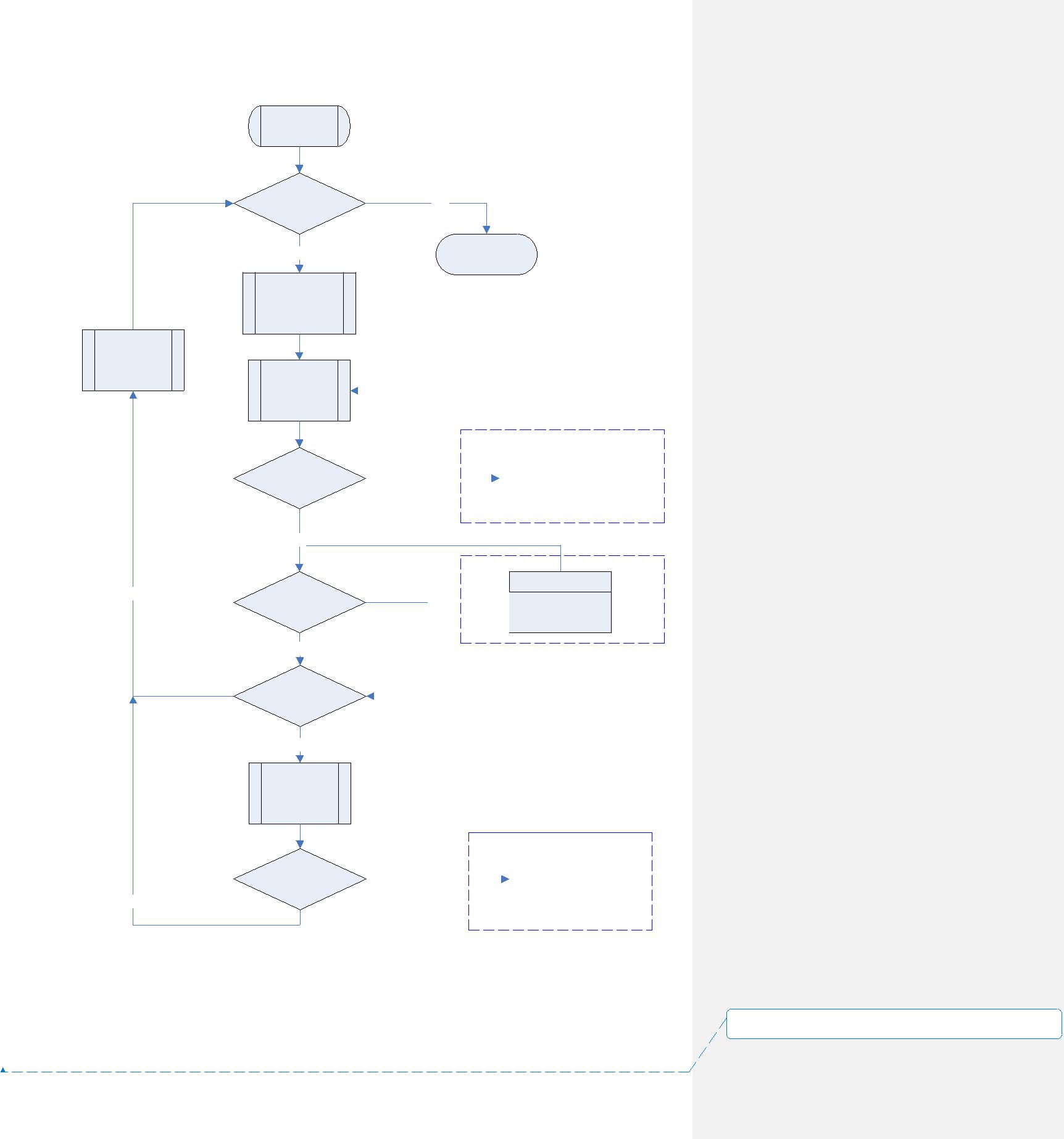
Check arrive

directory

arrive directory

contains file?

15~~4~~ November 2005



Yes

End

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  | Yes |  |  | File Processor | |  |
|  |  |  |  |
|  |  |  |  |  | Child process | |  |
|  |  |  |  |  |  |  |  |

Non-wait

Yes Copy / Extraction

Child process

Non-wait

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  | Yes |  |  | File Processor | |  |
|  |  |  |  |  |
|  |  |  |  |  | Child process | |  |
|  |  |  |  |  |  |  |  |

Non-wait

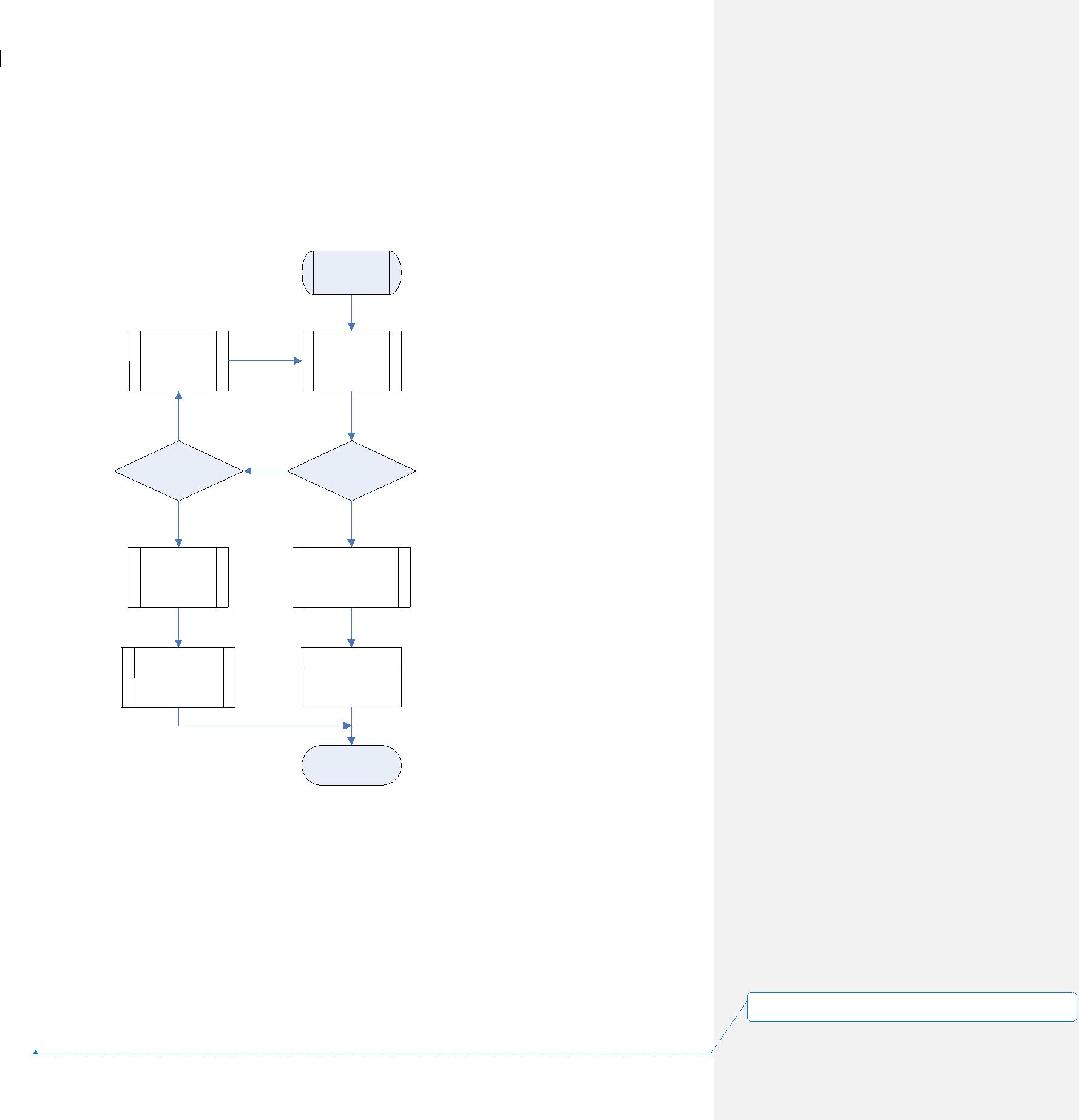
**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检

查拼写或语法

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~

21 of 46

Page



ETL Technical Specification 15~~4~~ November 2005

The Pull/Extract Child Process

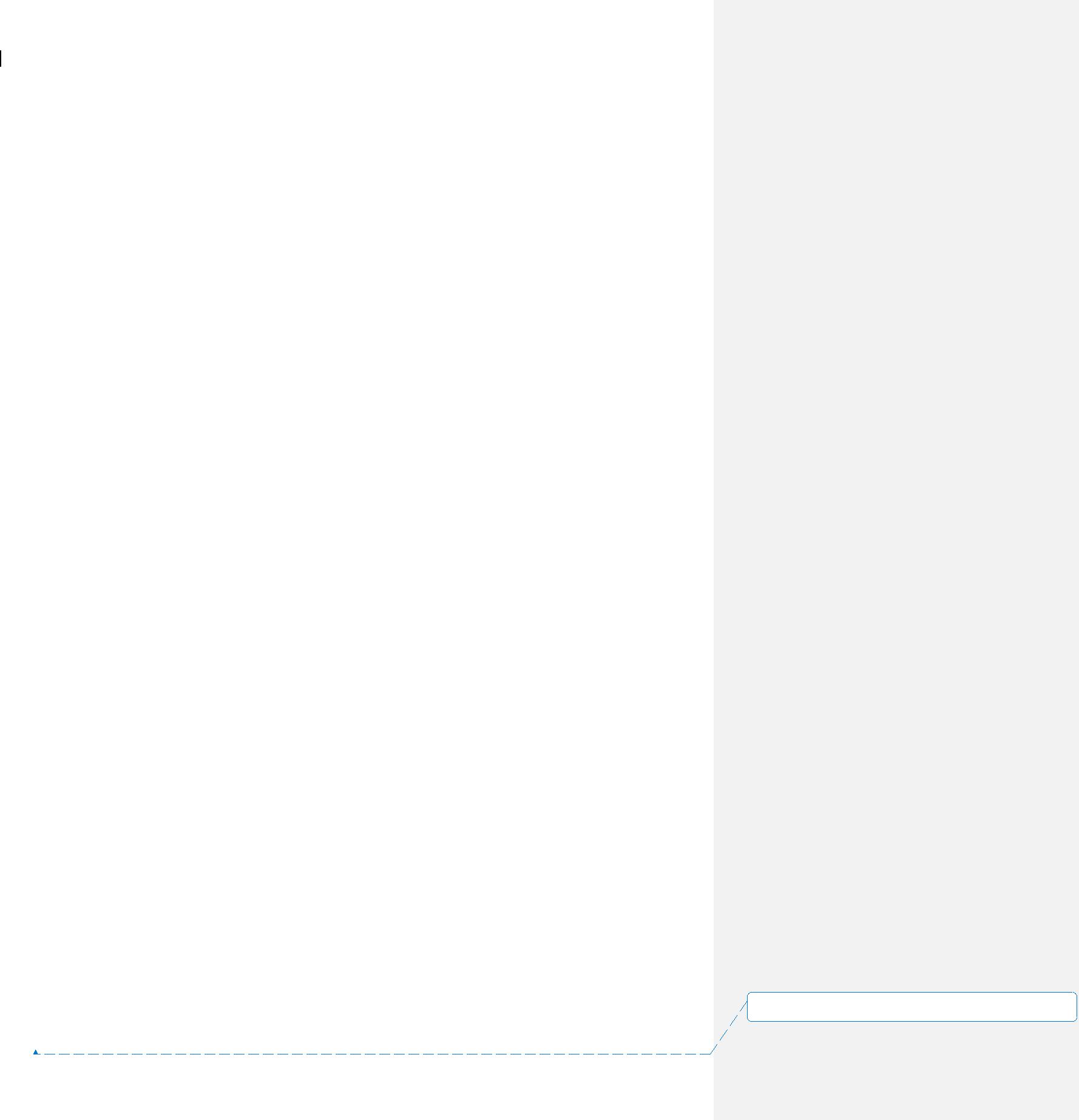
This is the child process forked by the Pre-process daemon. It receives the file information and performs pulling by either an ftp, rcp, copy or extraction script. Upon completion, it either places files to the Arrive directory or sends error message to HP Openview.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Start |  |
| Sleep for |  | Perform ftp/ |  |
|  | rcp/copy/ |  |
| Retry mins |  |  |
|  | extract |  |
|  |  |  |
| No |  |  |  |
| Retry count | No | Successful? |  |
| exceeded? |  |
|  |  |  |
| Yes |  | Yes |  |
| Notify |  | Update |  |
|  | ETL\_Source\_File |  |
| HP Openview |  |  |
|  | table |  |
|  |  |  |
| Update |  |  |  |
| ETL\_Source\_File |  | File Processor |  |
| table |  |  |
|  | Child process |  |
|  |  |  |
|  |  | End |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检

查拼写或语法

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 22 of 46 | |  |



ETL Technical Specification 15~~4~~ November 2005

File Processor Child Process

It is the child process called by either the pre-process daemon or the pull/extraction child process to handle a file set. It performs all required processing on the set based on the configuration table. Finally, it will either reject the files by sending message to HP Openview or create EA control file for loading.

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 23 of 46 | |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法

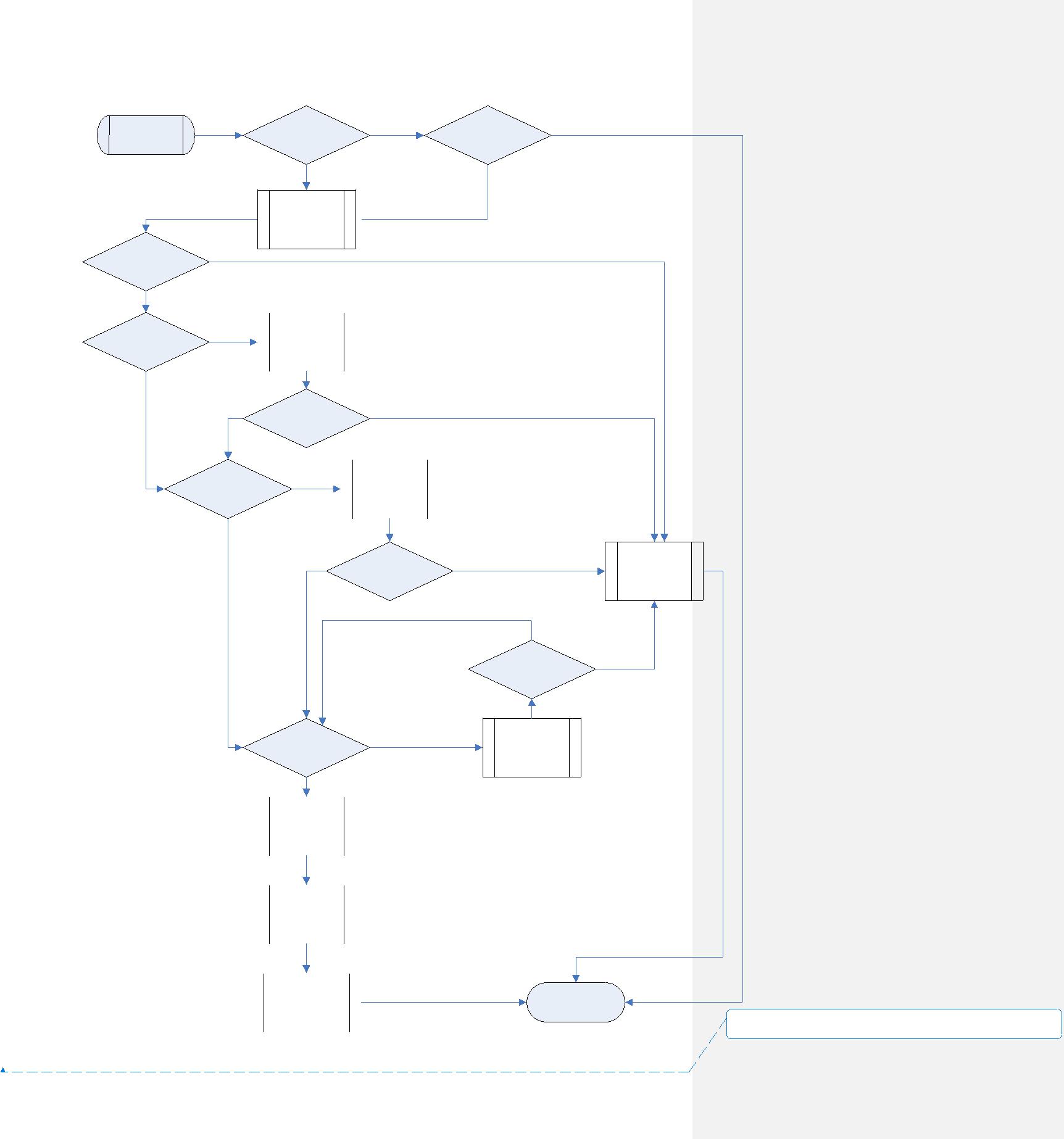
ETL Technical Specification

Start

Move successful?

Yes

15~~4~~ November 2005



|  |  |  |  |
| --- | --- | --- | --- |
| Need to check EA | Yes | Control file found |  |
| fail dir? | in EA fail dir? |  |
|  |  |

No

Move file set

to working dir  No

and backup dir

No

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| File set needs | Yes |  | Unzip file(s) |  |
| unzip? |  |  |
|  |  |  |  |
|  |  |  |  |  |

No

Unzip

Yes successful?

No

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Need to run special | Yes |  | Run special |  |
| script? |  | handling script |  |
|  |  |  |
|  |  |  |  |  |

Run script

successful?

Yes

No Yes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Check more |  | Yes |  |
|  | variance? |  |  |
|  |  |  |  |
|  | No |  |  |  |
|  |  |  |  |  |
|  | Move file set |  |  |  |
|  | to EA receive |  |  |  |
|  | dir |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | Generate |  |  |  |
|  | control file to |  |  |  |
|  | EA receive dir |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | Update |  |  |  |
|  | ETL\_Source\_File |  |  |  |
|  | table |  |  |  |
|  |  |  |  |  |

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~

24 of 46

No

Check variance

pass?

Check input

variances

End

Yes

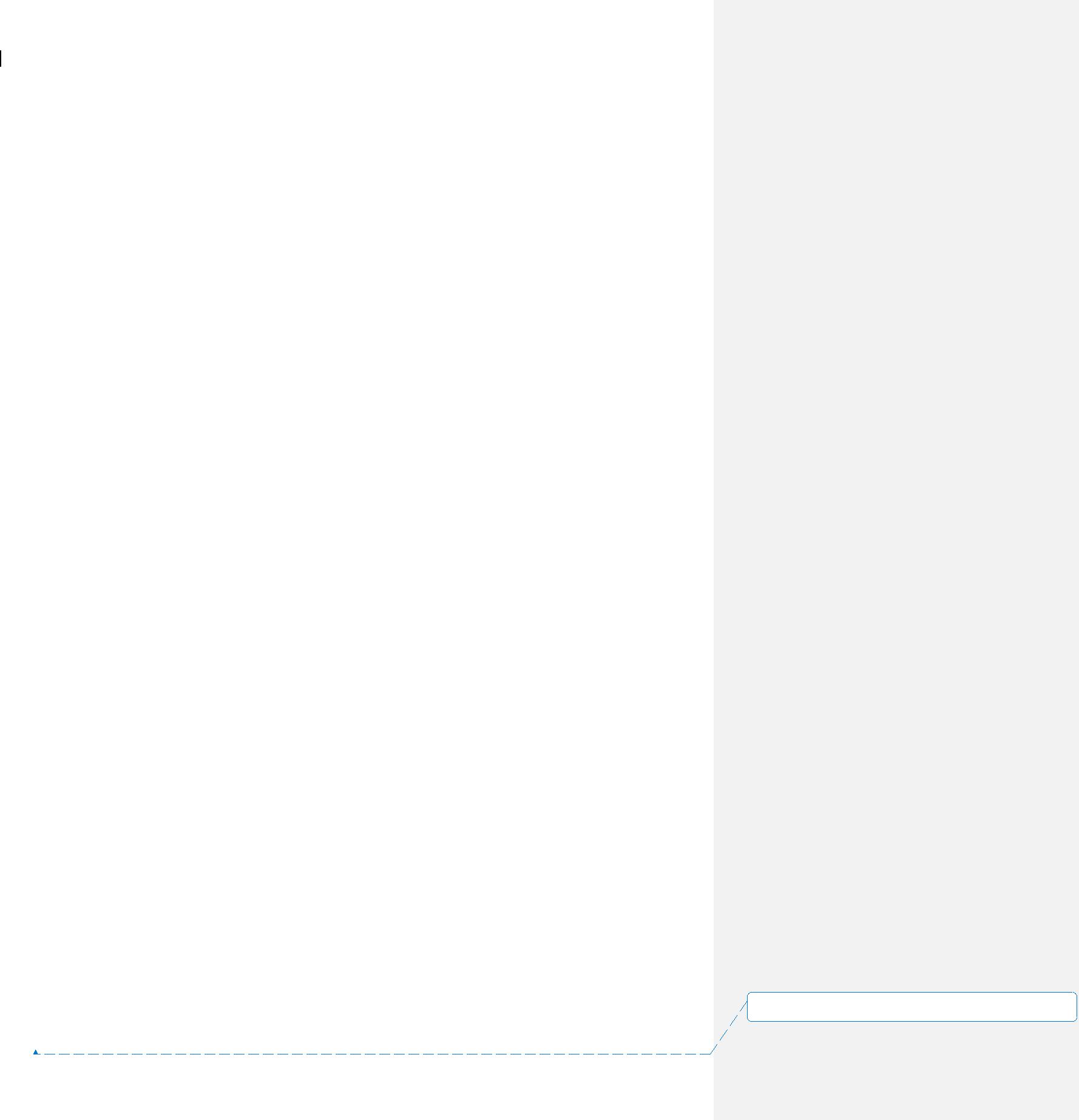
Notify

HP Openview

No

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼 写或语法

Page



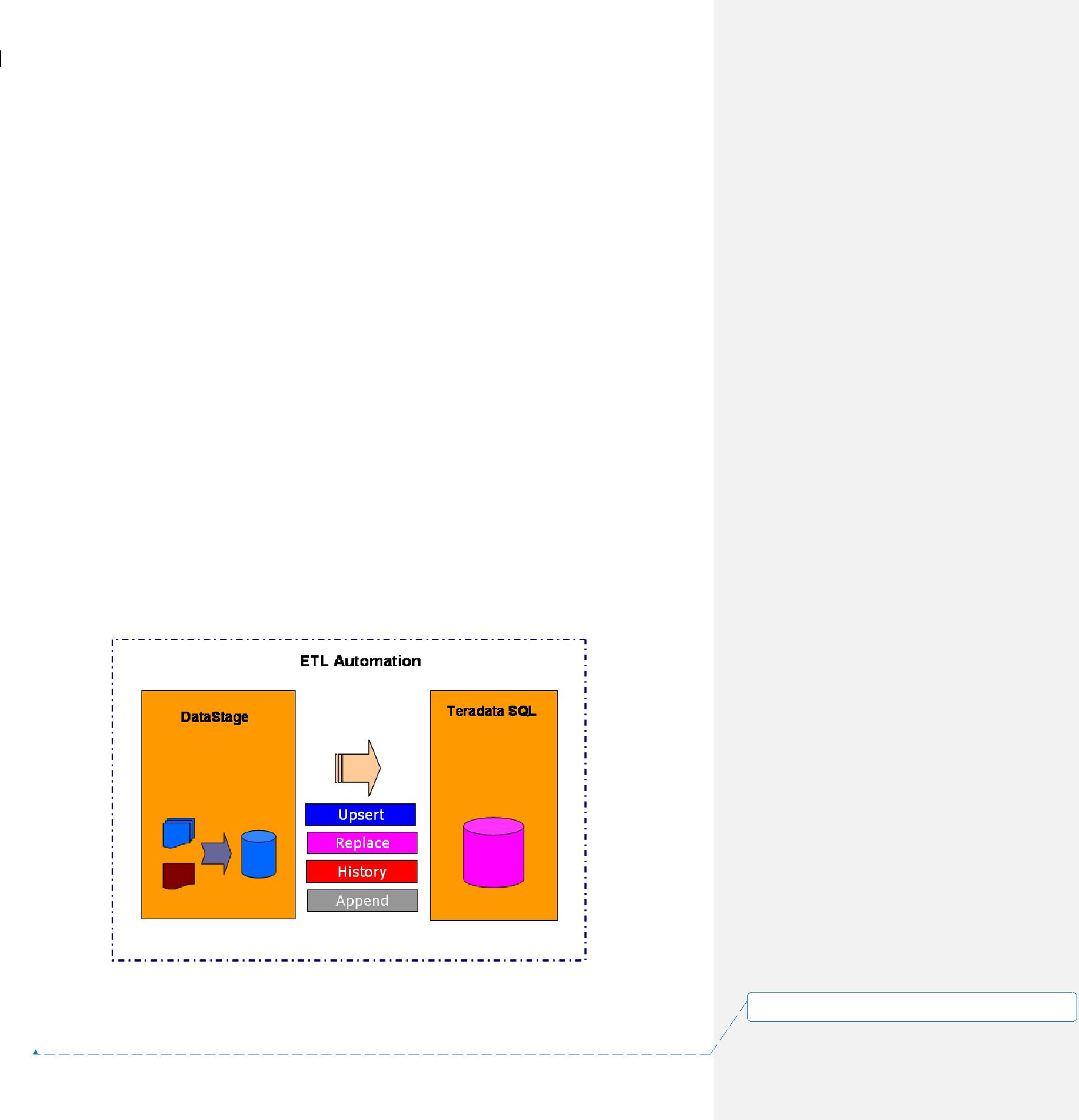
ETL Technical Specification 15~~4~~ November 2005

**2.2.11 Pre-process related Directory**

The following directories are used by the pre-process. Sub-directories of different data sources are indicated by <Data source> abbreviation code. The <Sys\_date> represents a sub-directory for the ease of file housekeeping.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Directory (bill02:/opt/etl)** | **Example File** | **Description** | |
|  |  | ~/preprocess/daemon/scri | \*.ksh | Scripts for the daemon | |
|  |  | pt |  |  |  |
|  |  | ~/preprocess/daemon/conf | adw\_preprocess.c | Configuration file includes: | |
|  |  | ig | fg | ♦ | EA Database Name |
|  |  |  |  | ♦ | Database User Id |
|  |  |  | exit.token | ♦ | Password |
|  |  |  |  | ♦ Data Refresh Interval in | |
|  |  |  |  |  | Minutes |
|  |  |  |  | ♦ | Log Housekeep Days |
|  |  |  |  | Dummy file to control daemon life | |
|  |  |  |  | The daemon will check itself and | |
|  |  |  |  | terminate if this exit file is found. | |
|  |  | ~/preprocess/daemon/log/ | adw\_preprocess.l | Log file directory for the | |
|  |  | <Sys\_date> | og | preprocess daemon. | |
|  |  | ~/openview | preprocess.log | Directory that contains log files | |
|  |  |  |  | monitored by HP Openview. | |
|  |  | ~/preprocess/<Data\_sourc | \*.log | Log file directory for various data | |
|  |  | e>/log/<Sys\_date> |  | source processing | |
|  |  | ~/preprocess/<Data\_sourc | \*.ksh | Directory for special handling | |
|  |  | e>script | \*.pl | scripts and DataStage extraction | |
|  |  |  |  | jobs. | |
|  |  | ~/preprocess/<Data\_sourc | \*.ctl | Data file directory at which pulled | |
|  |  | e>arrive | \*.dat | or extracted data files are located | |
|  |  |  | \*.gz |  |  |
|  |  | ~/preprocess/<Data\_sourc | \* | Working directory at which data | |
|  |  | e>working |  | files are under processing | |
|  |  | ~/preprocess/<Data\_sourc | \*.ctl | Data file directory at which backup | |
|  |  | e>/backup/<Sys\_date> | \*.dat | copy of pulled or extracted data | |
|  |  |  |  | files are located | |
|  |  | ~/preprocess/<Data\_sourc | \*.dat | Directory where data files are | |
|  |  | e>/rerun | \*.gz | placed for rerun case | |
|  |  | ~/preprocess/<Data\_sourc | \*.lock | Directory that contains lock | |
|  |  | e>/process |  | information to keep track of | |
|  |  |  |  | process running status | |
|  |  | ~/DATA/fail | \* | The directory at which the pre- | |
|  |  |  |  | process will check for previously | |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | | |  | Page |
| 25 of 46 | | |  |  |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

|  |  |  |
| --- | --- | --- |
|  |  | failed EA job |
| ~/DATA/receive | dir.<Filename> | Used by ETL automation |
|  | <Filename> | To place standardized control file |
|  |  | and data file to signify EA to start a |
|  |  | job |

**2.3** **DATA TRANSFORMATION AND LOADING APPROACH**

It should make suitable use of processing power of both DataStage server and Teradata server for transformation. In some cases, it may be simpler to implement transformation logic in DataStage. On the other hand, it may be more efficient to let Teradata to transform data by in-database table join.

Below are the general rules to determine where transformation will mainly take place:

|  |  |
| --- | --- |
| **Scenarios** | **Take place at** |
| Small table mapping | DataStage |
| Field level transformation logic | DataStage |
| Do not need to job with existing table |  |
| Process with huge volume of data | Teradata |
| Join with a table already exist in database | Teradata |

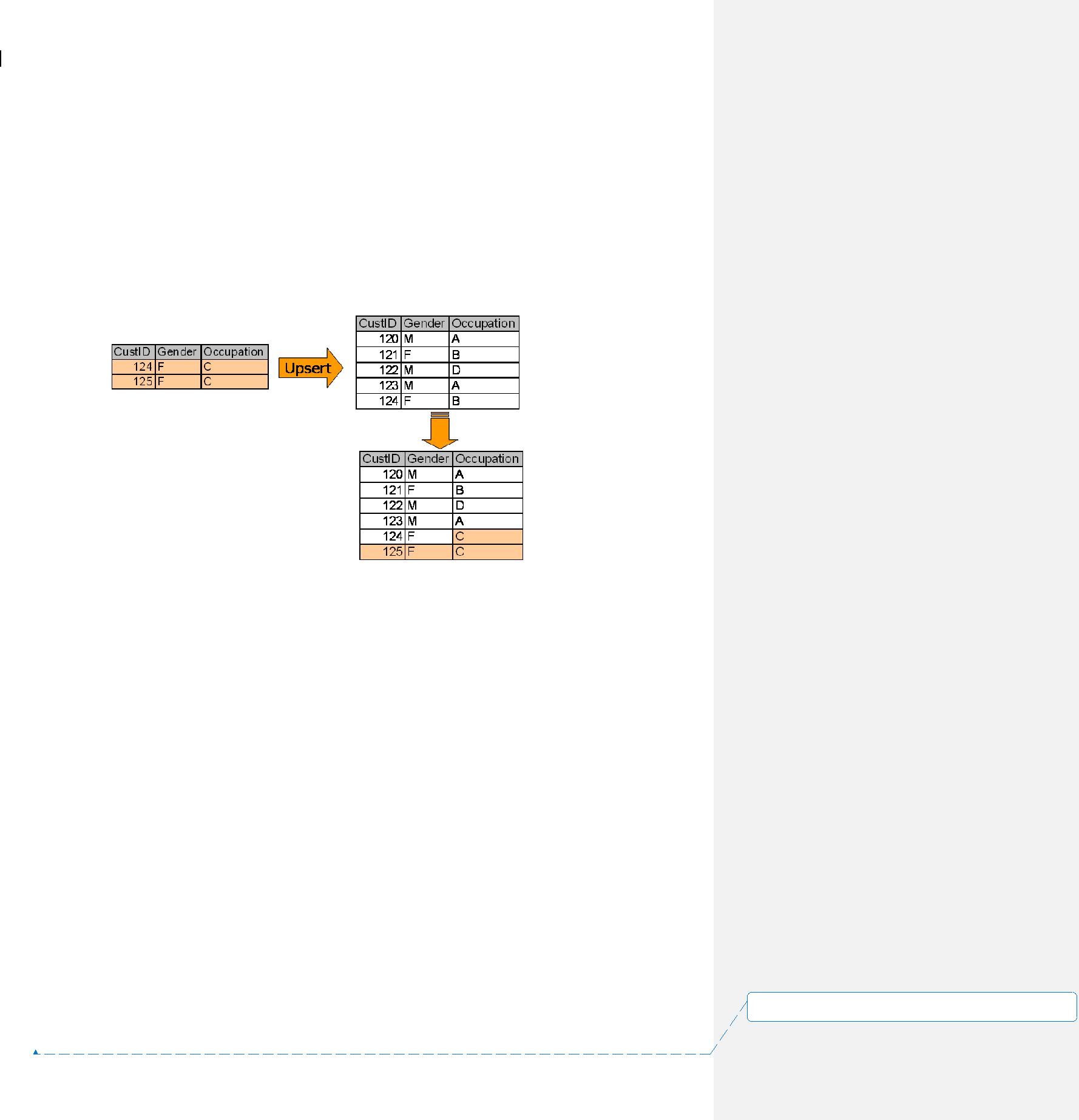
Four types of transformation approaches will be applied to different data according to their business needs.

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检

查拼写或语法

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

26 of 46



ETL Technical Specification 15~~4~~ November 2005

**2.3.1** **Upsert**

In an Upsert job, a new record is either inserted into the table, or replaces an existing record.

* It searches to see if a record exists in the target table, based on the primary key.
* If the record exists in the target table, the record is updated to the target table.
* If the record does not exist, the record is inserted to the target table.

**2.3.2** **Replace**

The logic is self-explained that the whole target table is just replaced by the source table.

**2.3.3** **History**

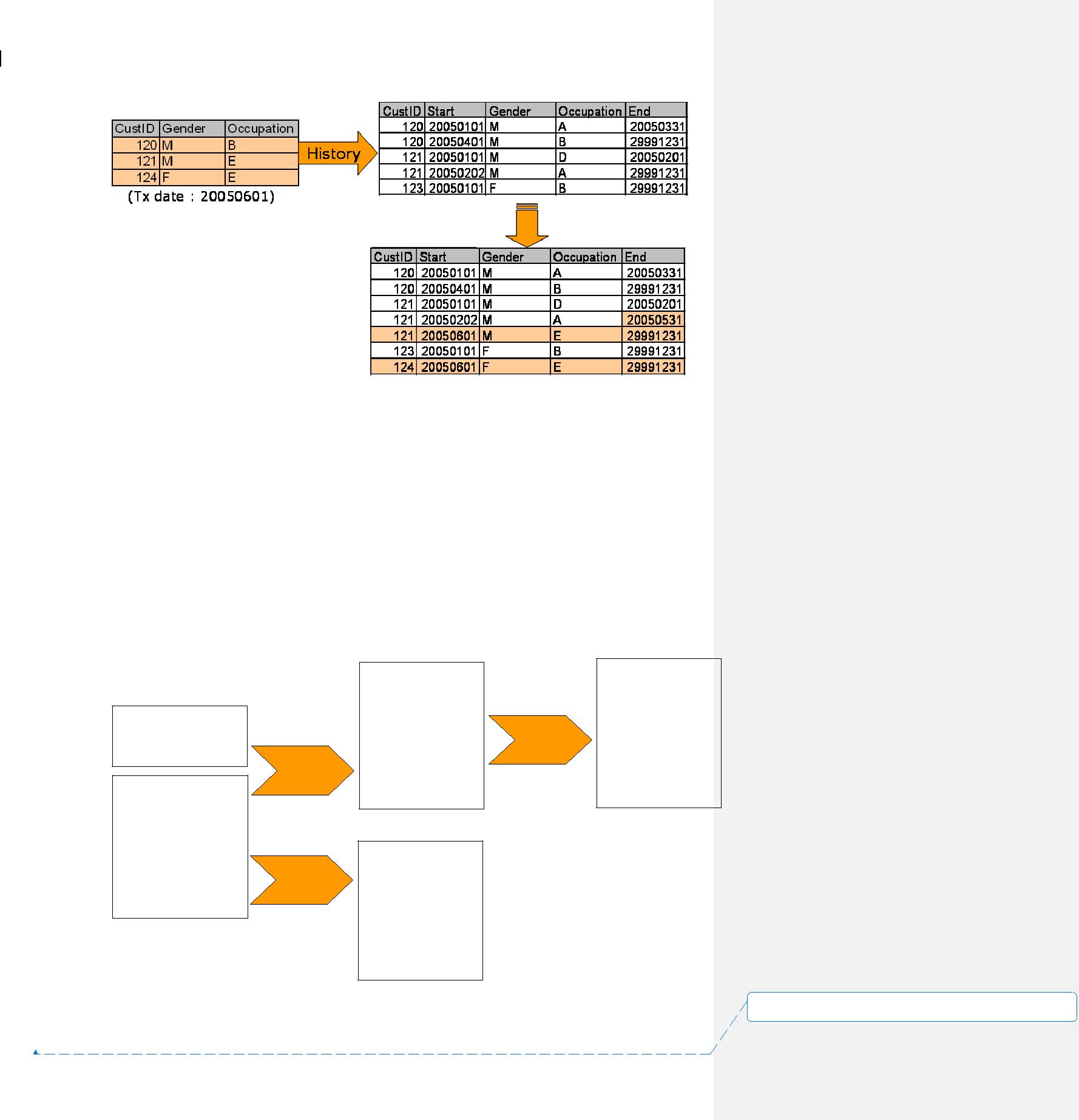
There will be an extra “Change\_Start\_Date” and “Change\_End\_Date” in the record to represent the life of the data elements.

* The Delta file/table is compared with the master table
* If a change of any column in the record found, the original record at the master table will be marked old, with the Change\_End\_Date = ‘20050531’.
* A new record with Change\_Start\_Date = ‘20050601’ and Change\_End\_Date = ‘29991231’ is created and inserted into the master table.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

27 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

**2.3.4** **Append**

The new arrival data is appended to the master table.

**2.3.5** **Table Image Switching Approach**

In order to minimize the period of database object locking and for the ease of re-run handling, table switching approach will be used in some tables. This approach may be applicable for table which involves Upsert, Replace, History logic. However, this will not be used for Append nature transformation and CDR table transformation due to database space and performance consideration.

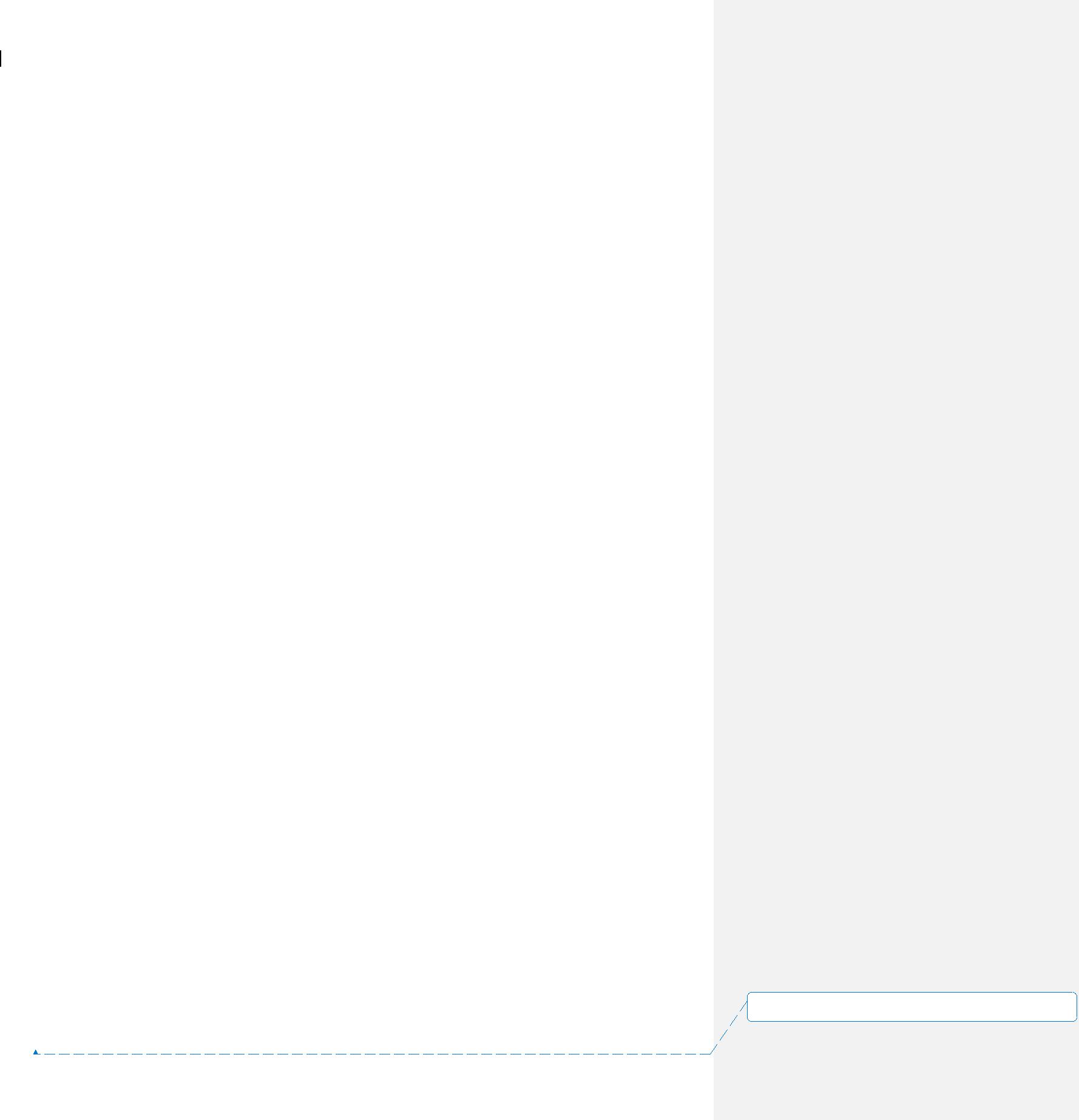
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | **Step 3** |  |  |
| **Customer\_Today** | **Step 1** | **Customer\_New** | **Rename** | **Customer** |  |
| **Table** |  | **Table** | **Table** |  |
|  | **Insert** |  |  |  |  |
| **Customer** | **Step 2** |  |  |  |  |
| **Table** |  |  |  |  |  |
|  | **Rename** |  |  |  |  |
|  |  | **Customer\_Old** |  |  |  |
|  |  | **Table** |  |  |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检

查拼写或语法

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

28 of 46



ETL Technical Specification 15~~4~~ November 2005

**Processing Steps:**

1. During the processing of ETL, the existing table will not be updated. Instead, a new table (<Table>\_new) will be inserted with the transformed data.
2. The original table is then renamed to old tables
3. Then, the new table is renamed to become the new master table for the date

**2.3.6** **CDR Table Implementation**

Teradata MultiLoad utility will be primarily used for upserting data to huge CDR tables. This is taken the following into considerations:

* MultiLoad does not cause large transient journal.
* It provides checkpoints and restart capability after a load failure.
* It allows Upsert logic for loading data.

For smaller CDR tables, the table image switching approach may be used.

**2.3.7** **Duplicate Row Handling**

There can be two ways to handle row duplication during loading. They may differ in some aspects:

**Duplicate rows are skipped**

Only one of the duplicated rows is loaded into table by FastLoad. Other duplicated rows are skipped. A post process will check the FastLoad log for the duplicate record count. If the count is non-zero, an additional script will run to capture these duplicate records into a data file for further follow up.

The capturing method involves a few steps:

* The original files in a set are concatenated into one file.
* The concatenated file is sorted.
* The concatenated file is sorted distinct to another file.
* The 2 files are diff’ed to produce the file storing duplicate rows.

Finally, HP Openview is alerted.

**Duplicate rows are loaded**

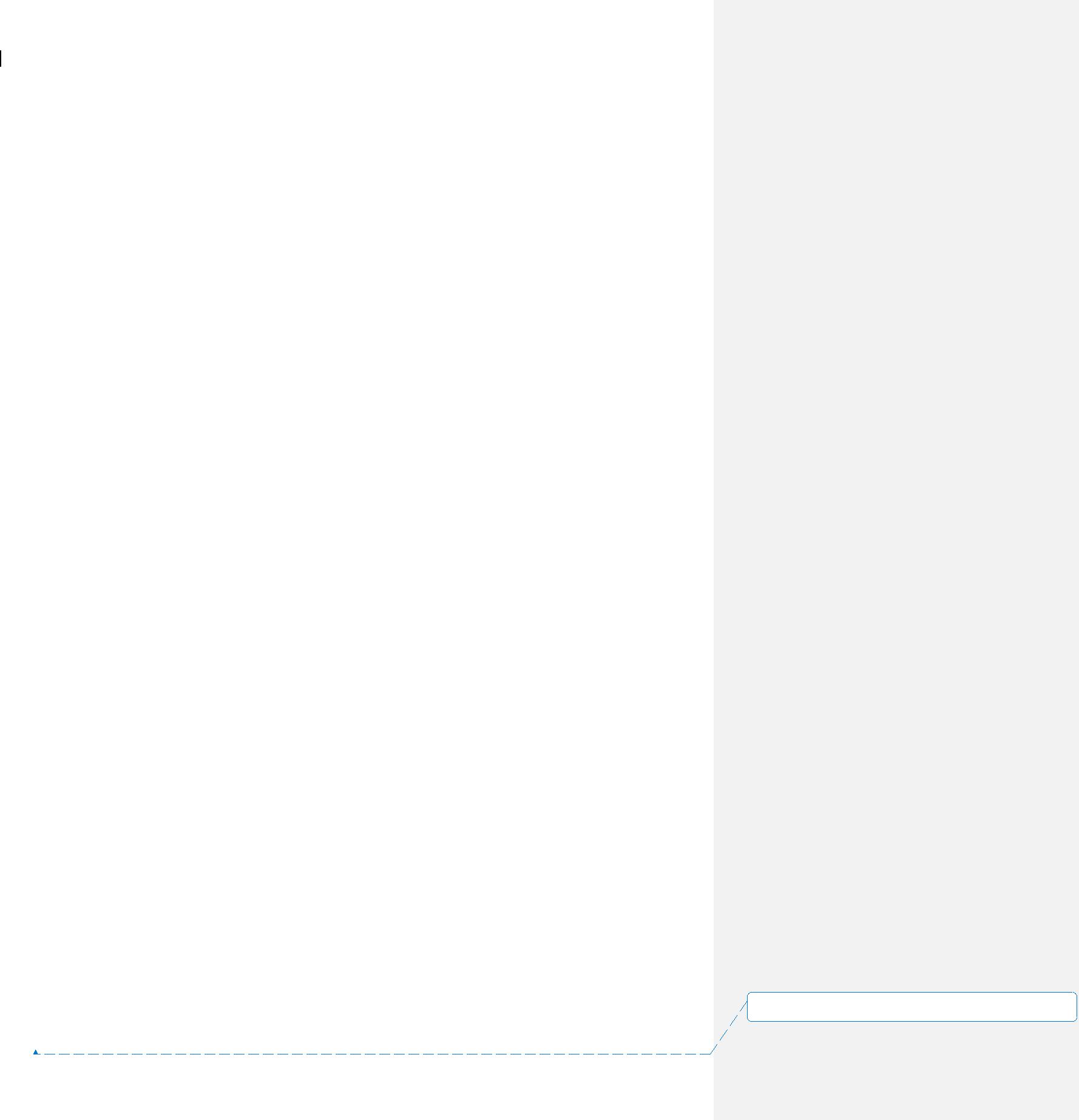
All duplicate rows are loaded into table. The table is defined as multi-set table with non-unique primary index. MultiLoad must be used for the loading. It returns successful also if duplicated rows are loaded.

The choice of handling method is subjected to individual job design.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

29 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

**2.4** **POST PROCESSING**

The post process is designed to capture some useful statistics from the job log files. These statistics are kept in database as historical data to serve the purpose of performance tuning as well as data quality control analysis.

**2.4.1** **Output Variance Checking**

Two kinds of checking will be carried out at post ETL stage.

1. Error Record Count

This includes rejected record count captured in error tables.

1. Data Variance

This is the table field chosen to impose checking on its aggregated value.

1. Output Record Count

The count of record loaded to table.

Any checking exceeding the margins will send message to HP Openview.

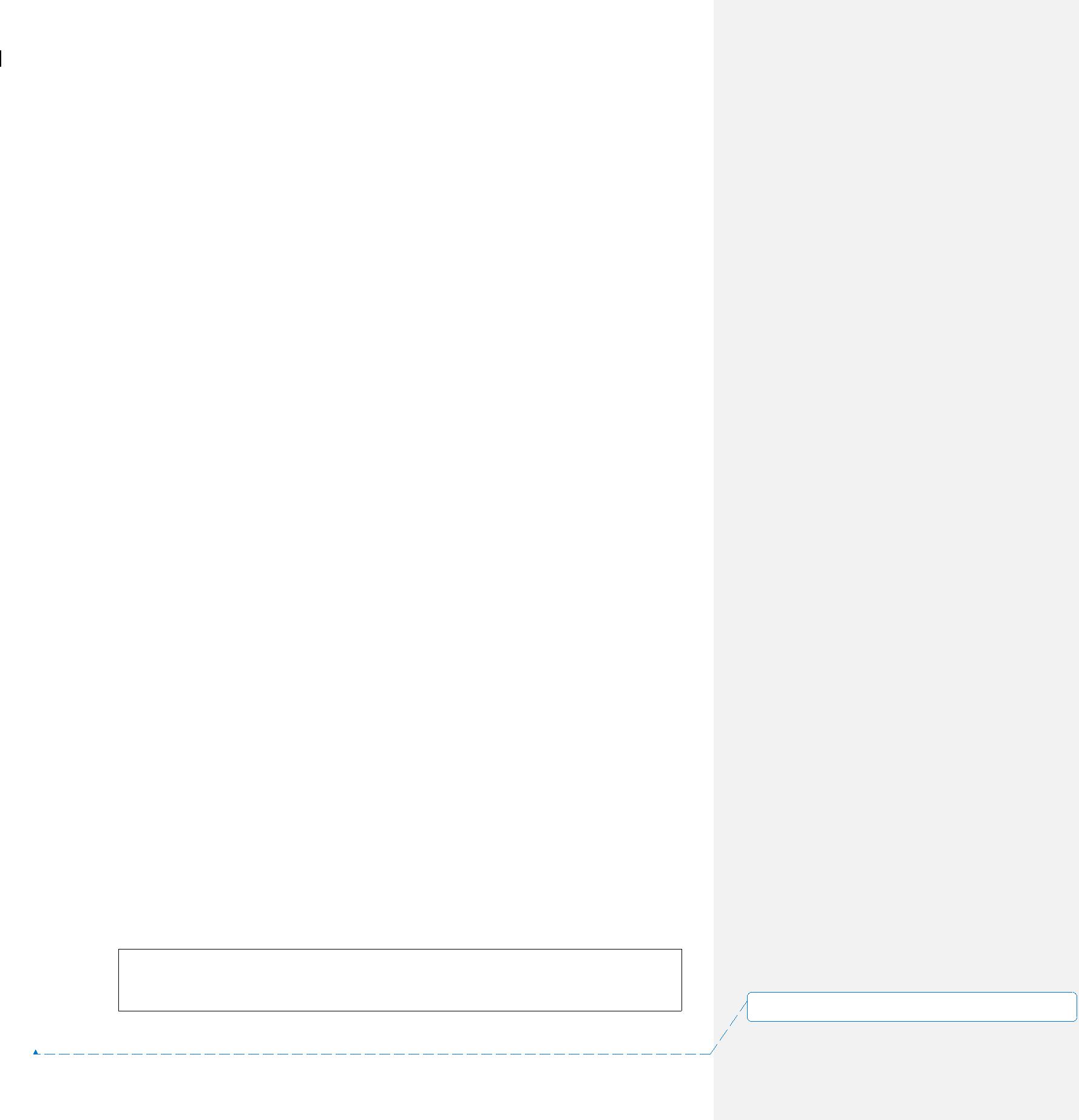
**2.4.2** **Table As\_of\_Date and Refresh\_date**

It requires a piece of information in the ADW to indicate the version of data image for the table. They will be updated whenever the table image changed. This is stored in the ETL\_Table\_Date table in the ADW.

The ETL\_Table\_Date contains the fields below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Column name** | **Description** |  |  |
|  |  | Table\_Name | Table name in the ADW |  |  |
|  |  | As\_Of\_Date\_Determine\_Method | T – From a column in loaded table |  |  |
|  |  |  | F – From file, the |  |  |
|  |  |  | ETL\_SOURCE\_FILE.Last\_File\_As\_Of\_Dat |  |  |
|  |  |  | e will be used |  |  |
|  |  | As\_Of\_Date\_Determine\_Column | The column name storing the As\_Of\_Date |  |  |
|  |  |  | information. |  |  |
|  |  |  | Only valid if As\_Of\_Date is determined |  |  |
|  |  |  | from loaded table |  |  |
|  |  | Filename\_Mask | Foreign key to |  |  |
|  |  |  | ETL\_Source\_File.Filename\_Mask |  |  |
|  |  | Data\_Path | Foreign key to ETL\_Source\_File.Data\_Path |  |  |
|  |  | Data\_Retention\_Period | A code representing the data retention |  |  |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | | | Page | |
| 30 of 46 | | |  |  |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

|  |  |
| --- | --- |
|  | period of the ADW table on monthly basis. |
|  | E.g. |
|  | 12+1 |
|  | 24+1 |
|  |  |
| As\_Of\_Date\_Offset | This is used to adjust the final table |
|  | As\_Of\_Date. |
|  | 0 |
| As\_Of\_Date | Image date for the table |
| Refresh\_Date | Last refresh date for the table |

**Determination of As\_Of\_Date and Refresh\_Date**

In general, there are several ways to determine the table image As\_Of\_Date, based on the availability of this date information following the priority below:

1. By a date or timestamp field inside the data record
2. By the YYYYMMDD portion of the data file name
3. By the timestamp of the data file

The As\_Of\_Date will be of ‘timestamp(0)’ data type in the ADW repository. In the case that the time portion cannot be determined, the time portion will be filled by ’00:00:00’.

The Refresh\_Date will be of ‘timestamp(0)’. It is the system timestamp captured at the end of job.

**2.4.3** **Table Collect Statistics**

Table statistic is the key for the Teradata optimizer to choose appropriate execution plan. They should be collected at a regular basis to keep the information up-to-date. However, collecting statistics for large CDR tables will be very resource consuming.

In general, smaller table will be collected daily while larger tables will be collected weekly or monthly.

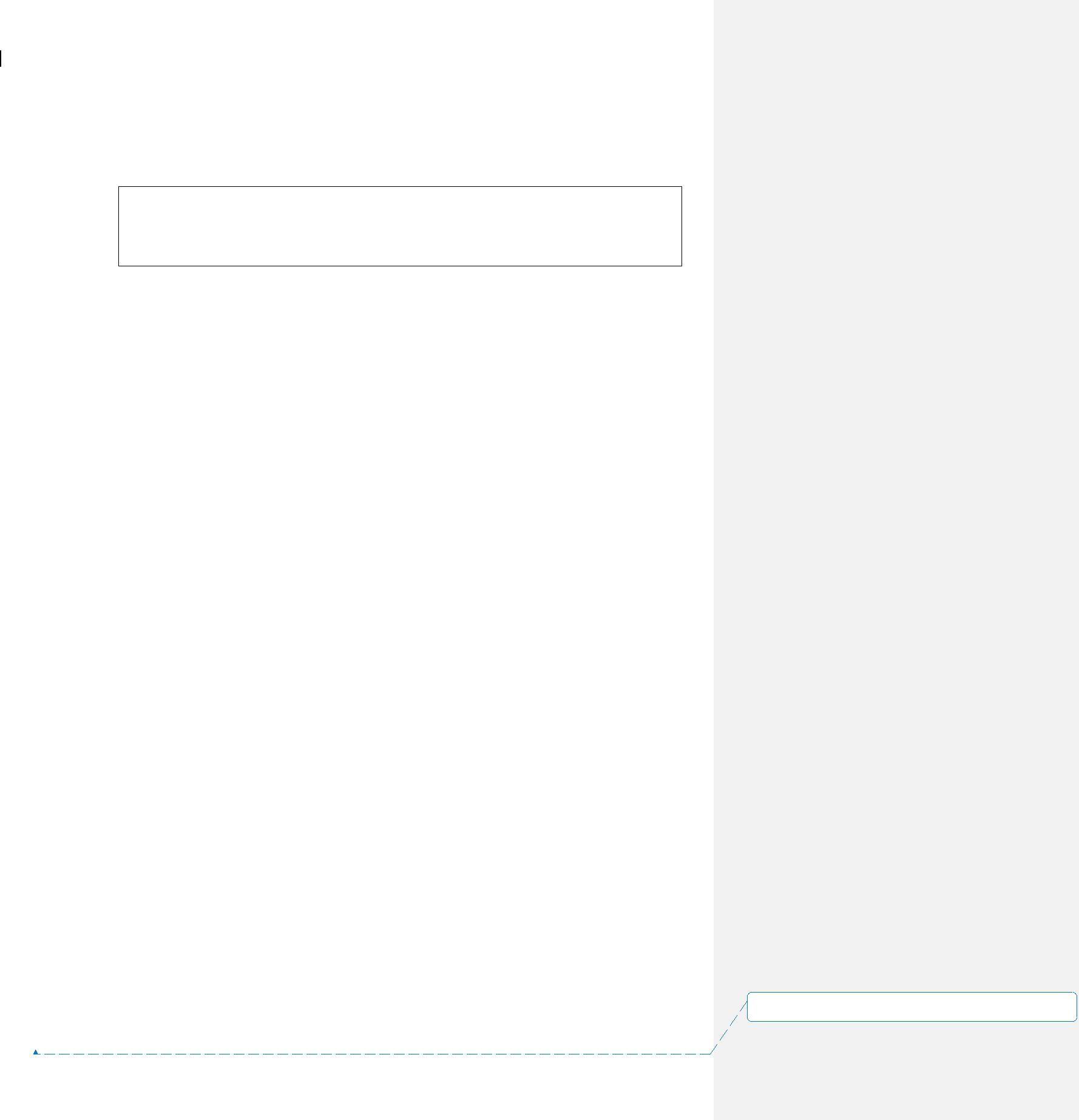
Syntax for collecting full statistics for a table:

* Collect statistics on <Table\_Name> index (<Index\_Columns>);
* Collect statistics on <Table\_Name> column (<Columns>);

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

31 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

For huge tables, only sample of the tables are collected.

Syntax for collecting sample statistics for a table:

* + Collect statistics using sample on <Table\_Name> index (<Index\_Columns>);
  + Collect statistics using sample on <Table\_Name> column (<Columns>);

1. **ETL AUTOMATION**

All loading jobs including DataStage jobs, FastLoad, MultiLoad and Bteq scripts must be defined in ETL Automation. Each job will have its own running schedule and dependency. Inter-job dependencies are also linked up there. ETL Automation acts as the central point for job maintenance and scheduling.

**3.1** **BATCH WINDOW**

There is no absolute time frame for pipelined extraction, loading, backup for the whole ADW. Each file or application specific file set will be processed on its own schedule. There generally two categories:

**3.1.1** **Night Batch Loading**

For heavy loading jobs, they will be arranged to run during night time, no matter the required data file already arrived or not. The loading characteristics of these files will be set to time triggered so that its running time can be controlled.

**3.1.2** **Immediate Loading**

The file will be loaded immediately upon data arrival. This is considered to be used for some small to medium size tables.

**3.1.3** **File Directory Flow**

The following diagram describes the file flow among different directories in pre-process and ETL Automation processing.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

32 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法

ETL Technical Specification

ull

cdr\_20051024

\_01.gz

Push

Extract

Normal data

source

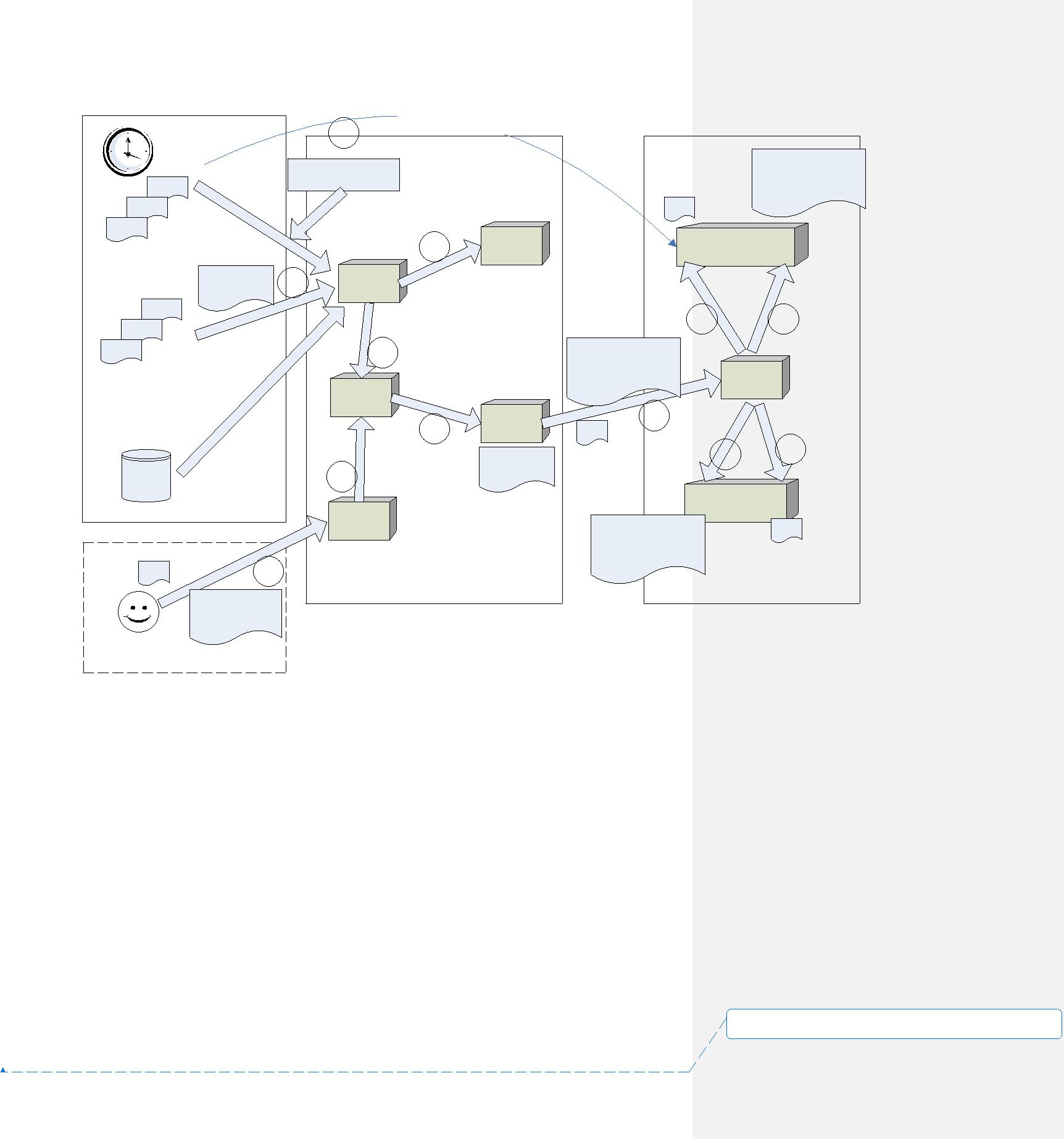
rerun.token 11

cdr\_20051024\_03

Re-run data

source

|  |  |  |
| --- | --- | --- |
|  | Check for control file in |  |
| 1 | EA Data/Fail directory |  |
|  |  |



File mask:

cdr\_yyyymmdd\_\*.gz

Match

backup

4 Directory

1. arrive

Directory

|  |  |  |  |
| --- | --- | --- | --- |
| Move |  |  |  |
| 3 | Unzip |  |  |
|  |  |  |
| working | Transpose |  |  |
| Decrypt |  |  |
| Directory |  |  |
|  | working |  |
|  |  |  |
|  | 5 | Sub- |  |
|  | directory |  |
|  |  |  |

cdr\_20051024

Move \_01

12

rerun

Directory

Pre-process

15~~4~~ November 2005

md\_loadcdr\_20051024

md\_loadcdr\_ \_01\_cdr\_20051024

20051024.dir

EA

DATA/fail

Directory

Fail

9 10

md\_loadcdr\_20051024

\_01\_cdr\_20051024 EA

receive

Directory

|  |  |  |  |
| --- | --- | --- | --- |
|  | 6 | Success & |  |
|  | Success |  |
|  | Fail |  |
| dir.md\_loadcdr |  |  |
| 7 | 8 |  |
| \_20051024 |  |  |
|  |  |  |

EA

DATA/complete

Directory

md\_loadcdr\_20051024

\_01\_cdr\_20051024

dir.md\_loadcdr

\_20051024

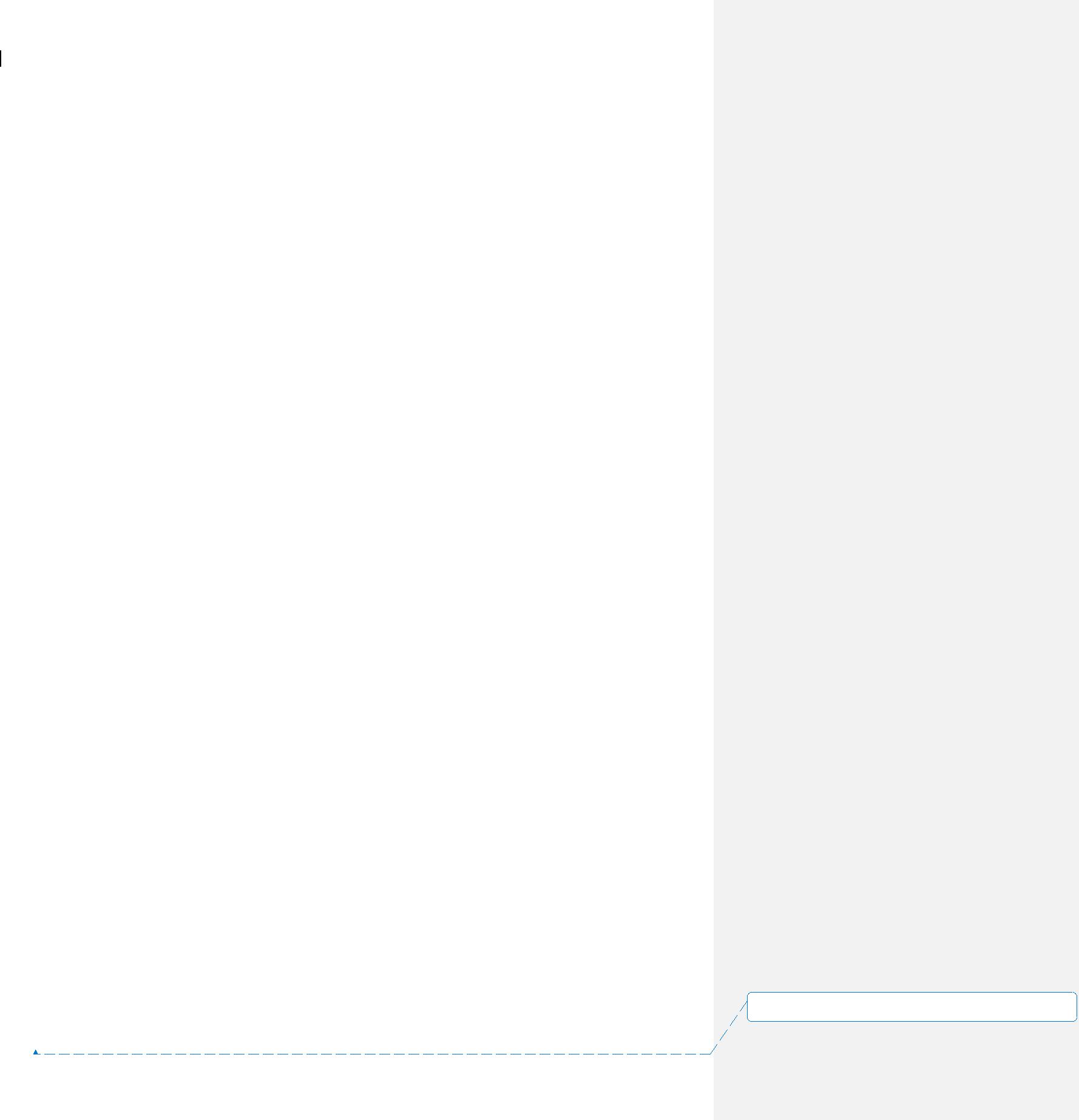
ETL Automation

1. If Check\_EA\_Fail\_Dir\_Flag is set, it firstly checks if file name mask has previous failed job from EA ~DATA/fail directory.
2. If no previous job failed, the data files are matched using the file name mask. Matched data files are moved to their own arrive directory.
3. The Monitor daemon will check the arrive directory by periodic time interval. To process a file set, the files are moved to a separate working directory. This can avoid ambiguity of more new files being pushed in.
4. A copy of the file set is also copied to backup directory. This copy is used for future troubleshooting and re-run.
5. The file set is further moved in a run-time determined sub-directory (By system date time and process id). At which, specific processing is carried out (E.g. Unzip, transpose, decrypt, Excel file massage). Validations will be imposed also.
6. After validation passed, data files are moved to EA arrive directory, followed by a control file.

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检

查拼写或语法

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 33 of 46 | |  |



ETL Technical Specification 15~~4~~ November 2005

1. If an EA job failed, the control file (Renamed from “dir.XXX\_yyyymmdd” to “XXX\_yyyymmdd.dir”) will be additionally moved to EA ~/DATA/fail directory. The original control file (without rename) will move to the ~/DATA/complete directory.
2. If the job complete successful, the control file will be moved to EA ~/DATA/complete directory only.
3. Data files are moved to ~/DATA/fail directory if job is failed.
4. Data files are moved to ~/DATA/complete directory if job is successful.
5. In case of re-run, new data files are prepared and placed at the rerun directory for its data source group. After all files for this re-run are placed, a special named token (I.e. rerun.token) is touched at the directory also to indicate the starting of re-run.
6. The monitor daemon will treat the data files for re-run and move to working directory for further process.

**3.1.4** **Running Jobs Multiple Times a Day in ETL Automation**

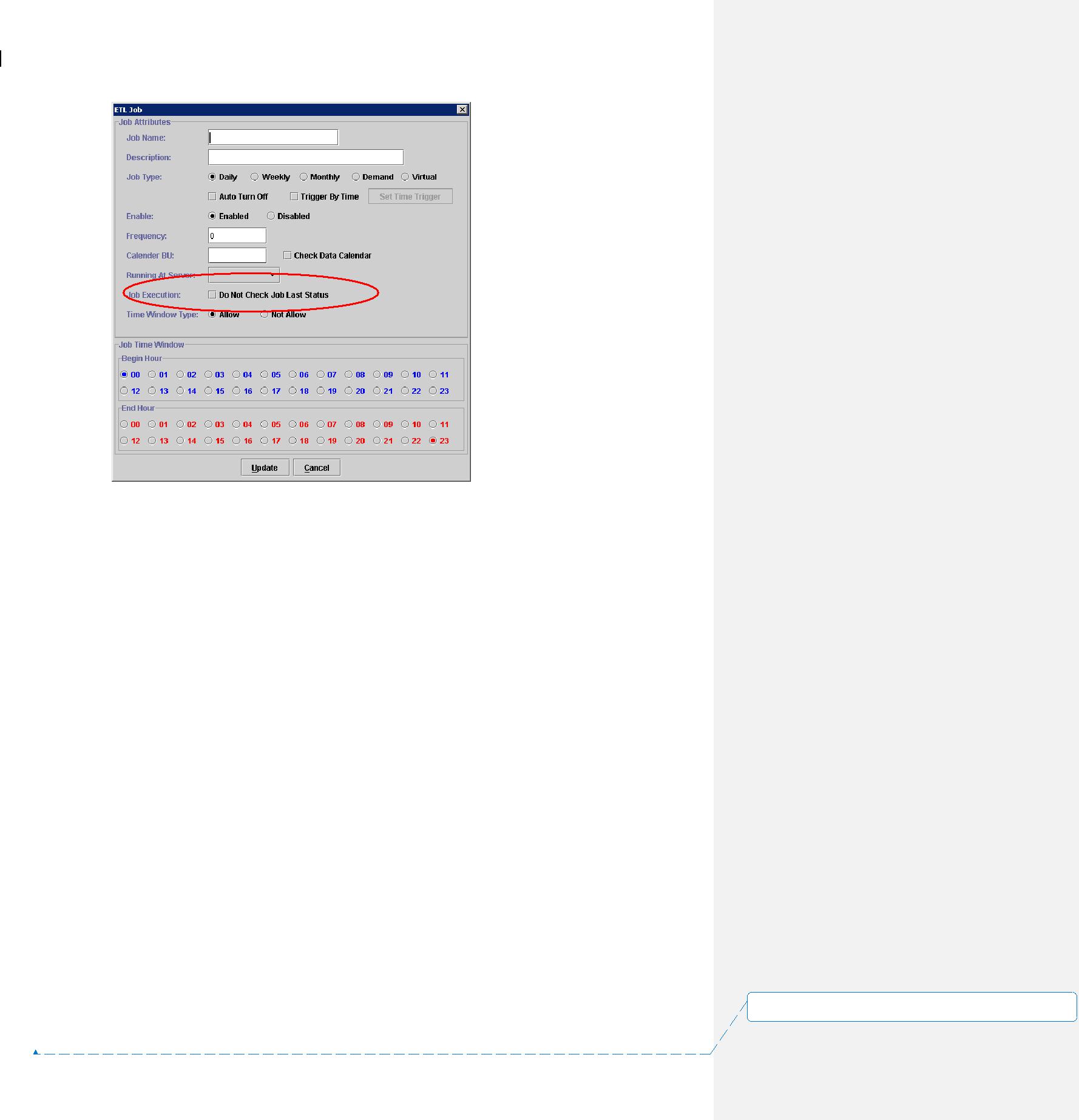
ETL Automation is originally designed to normally run a job once a day, based on its $TXDATE parameter. EA will prohibit further triggering of a job that has been run within a date. However, one can manually reset the job status to trigger the 2nd and subsequent runs.

In order to allow EA to trigger the same jobs multiple times a day automatically, it requires to have the “Do Not Check Job Last Status” flag checked. Hence, EA will bypass checking on the status of previous run, so that a job can run multiple times.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

34 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

**3.1.5** **Job Re-run**

In case of job failure, support staff will be alerted, and then appropriate fix will be applied. Sometime it involves script change, and sometime it involves data patching. After the data or related jobs are fixed, in general, there are two entry points for rerun.

**Restart from the Pre-process**

New data files are prepared to the rerun directory. A special named file token is touched to the rerun directory. The pre-process daemon will capture the token and move the files for re-run.

**Restart Jobs via ETL Automation**

The ETL Automation provides a client GUI for job scheduling and setting. Job statuses in the EA repository are reset from this client interface. Then, the job will be restarted after checking further dependencies. After successful completion, dependent jobs and down stream jobs will be started automatically.

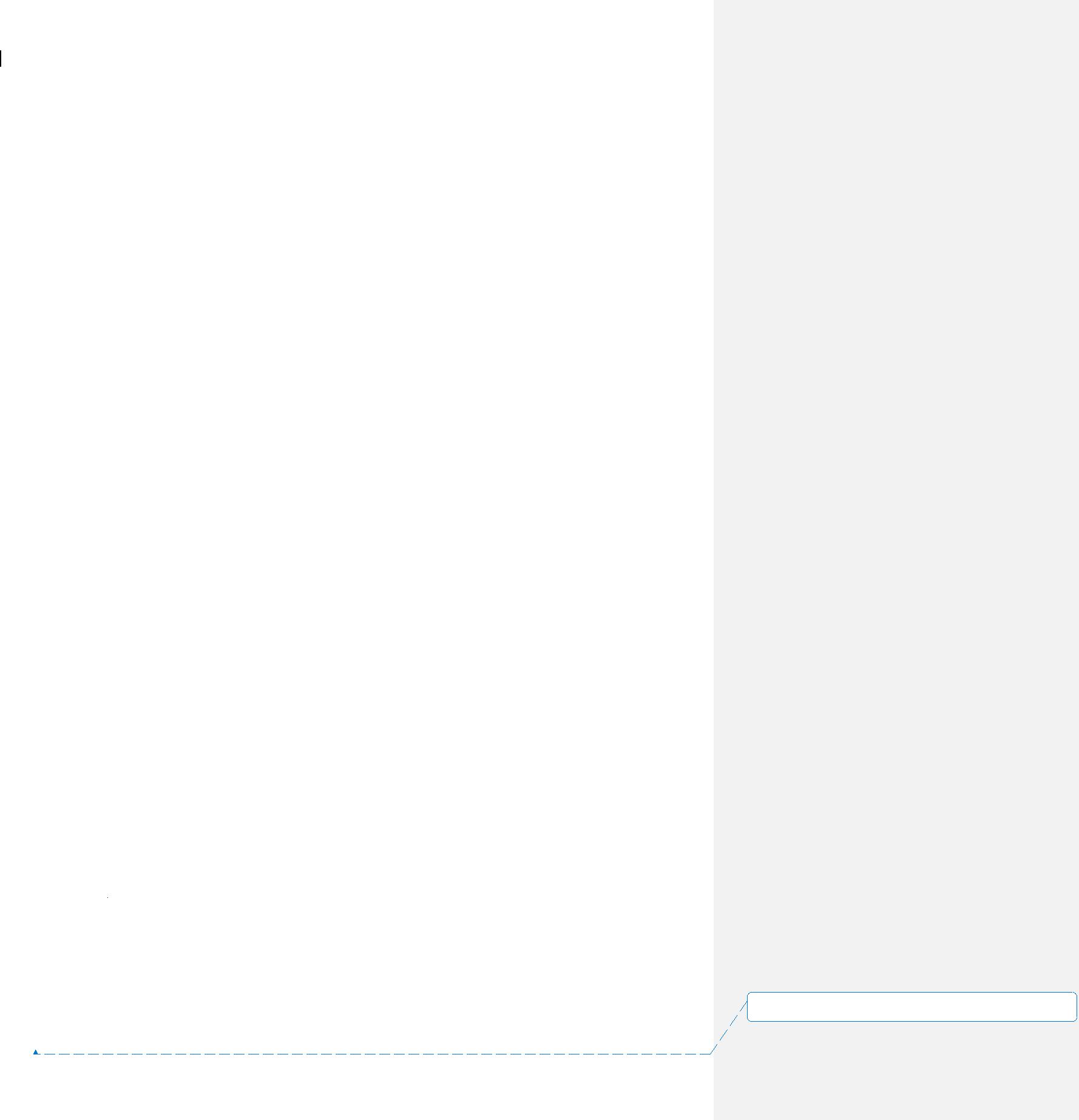
**3.1.6** **ETL Automation Directory**

All the job scripts are located at the Sun Solaris server (bill02). The three major components (Pre-process daemon, ETL automation service and DataStage jobs)

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

35 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

must follow this directory structures. The following table describes the directories and their purposes.

|  |  |
| --- | --- |
| **Directory (bill02: /opt)** | **Description** |
| ~/etl/APP/bin | For storing ETL job scripts |
| ~/etl/DATA | For storing EA data files and control files |
| ~/etl/LOG | For storing EA job logs |
| ~/etl/bin | For storing EA program jobs |
| ~/etl/axm | For storing loading access modules shared |
|  | objects |
| ~/etl/etc | For storing miscellaneous files |
| ~/etl/lock | For storing locking file for EA internal |
|  | processing |
| ~/etl/tmp | Temp directory for EA |

**3.2** **PASSWORD ENCRYPTION**

In order to allow batch loading jobs to access Teradata database, we keep the encrypted database logon string in a protected file. This is achieved by using the password encryption module ‘EncodeLogon.pl’ provided by ETL Automation. Whenever a password is needed, job script will call a decryption module ‘IceCode’ to decrypt it and store it as local variable within the job.

1. **HP OPENVIEW**

HP Openview will be the primary console for monitoring the whole ETL process.

Several components will send message for critical errors:

* The pre-process daemon will send error message if invalid file is encountered.
* ETL will also report job loading error as well as post-ETL checking failure.
* The usage of disk for the staging area should also be monitored by HP Openview.

In order to achieve these message communications, several directories and logging files are agreed to achieve monitoring purpose.

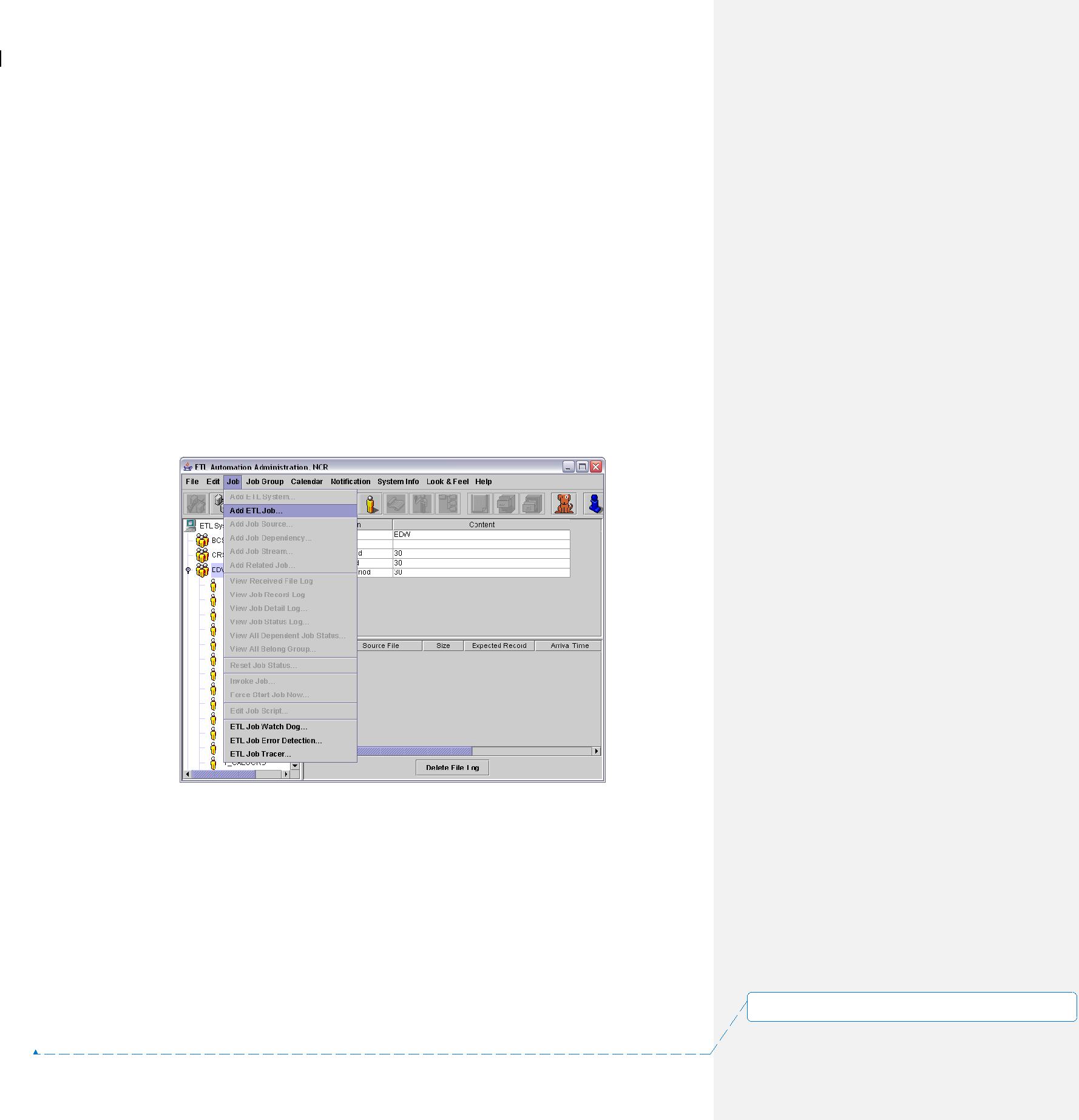
|  |  |  |
| --- | --- | --- |
| **Directory (bill02: /opt/etl )** | **Log Files** | **Reported from** |
| ~/openview | ea.log | ETL Automation |
| ~/openview | preprocess.log | Pre-process daemon |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检

查拼写或语法

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

36 of 46



ETL Technical Specification 15~~4~~ November 2005

1. **APPENDIX**

**5.1** **ETL AUTOMATION OPERATION INTRODUCTION**

**5.1.1** **Overview**

ETL Automation (EA) acts as job scheduler of the data loading and transformation in ETL operation. The elementary operation unit of EA is JOB. EA manages job by application systems and keep it in different directories. Each job will serve its own purpose like data loading, transforming.

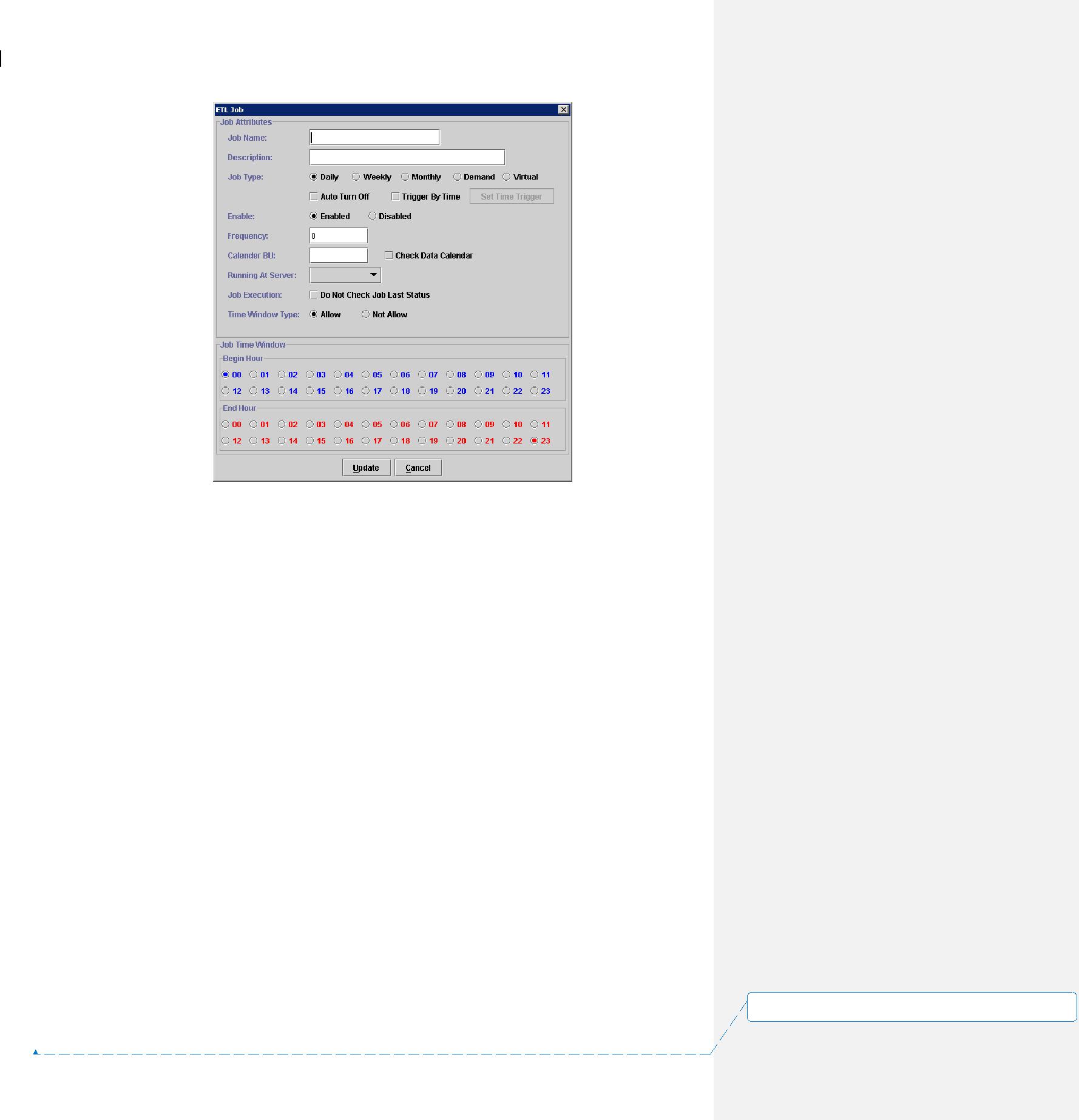
To create a new ETL System, choose a predefined ETL System under the ***“ETL*** ***System”*** tree item in the System Panel. Then select ***Job*** > ***Add ETL Job*** from themenu bar or click the add ETL Job icon from the tool bars.

Then ETL job dialog box will be shown up as below:

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

37 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

To create an ETL Job, you must fill in the following details:

**Job Name:** A unique name for the job with a limit of 50 characters.

**Job Description:** A short description for the job with a limit of 50characters.

**Job Type:** Daily – indicate the job is running everyday.

Weekly – indicate the job is running weekly.

Monthly – indicate the job is running monthly.

Demand – indicate the job only runs on demand.

Virtual – indicate a virtual job.

**Auto Turn Off:** If user chooses this option, the job will be disabled afterthe job is complete.

**Trigger By Time:** If user choose this option, the job will start automationby defining the running period of the job. **Enabled/Disabled:** Choose to enable or disable the ETL Job.

**Frequency:** By assigning a value to the Frequency field, user can control

the running period of the job. Example: by inputting “1, 10” in this field, the job will only process on 1st and 10th. EA will check the transaction of the control first before continuous to process. The valid values for this field are:

0 – everyday

-1 – end of the month

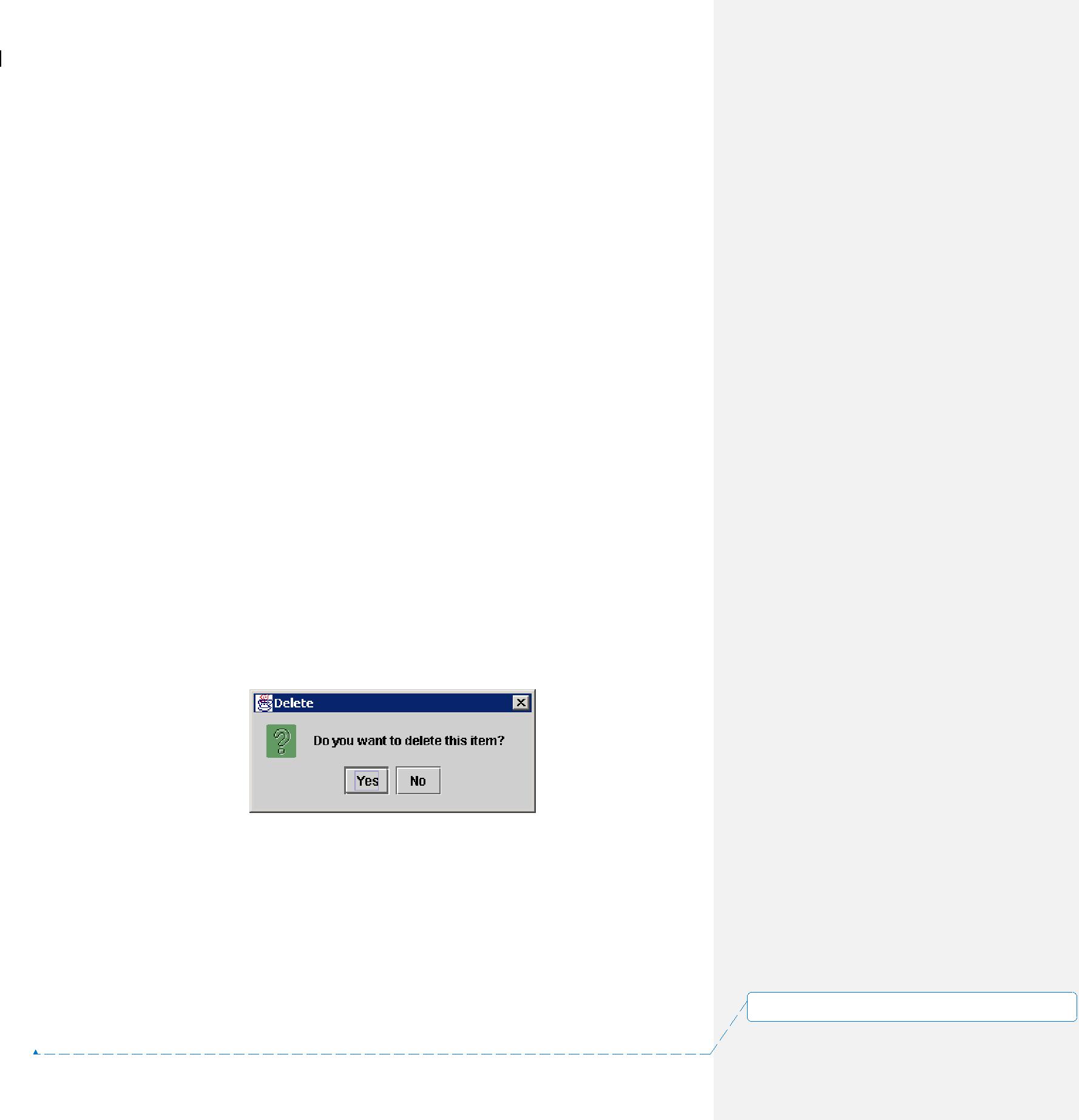
1 to 31 – 1st, 2nd…31st

41 to 47 – Monday, Tuesday…Sunday

**Calendar BU:** The name of the business unit.

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 38 of 46 | |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

**Check Data Calendar:** By choosing this option, EA will check thetransaction date of the control file to make sure it follows the sequence in the data calendar.

**Running At Server:** The server where the job is going to be run.

**Do not check job last status:** By disable this option, job can only be runwhen the last job status is complete.

**Time Window Type:**

Allow – By choosing this option, the time specify in Job Time Window will become the time that allow the job to run.

Not Allow – By choosing this option, the time specify in Job Time Window will become the time that do not allow the job to run.

**Begin Hour/End Hour:** By choosing the different option from the TimeWindow Type, the time period specify here will be the allow/not allow time period for the job.

After you have input all the details of the job, you can press ***“Update”*** to add the new job or ***“Cancel”*** to deny any changing.

To modify the setting of an ETL Job, you only need to click on the ***name of an*** ***existing ETL Job*** from the ***“System Panel”***. Then on the menu bar select ***Edit*** > ***Property*** or you can click the property icon showing on the window.

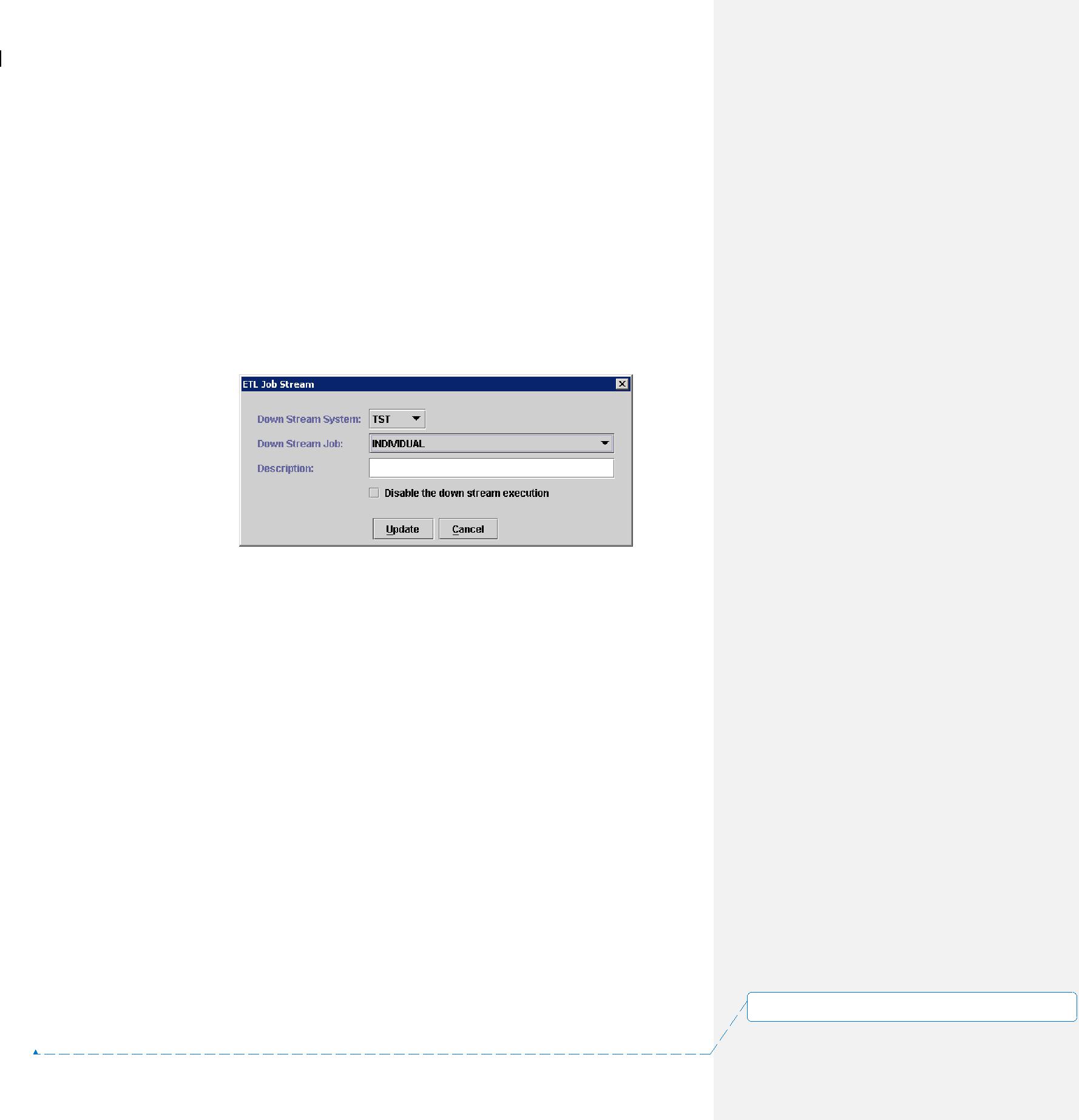
To delete an existing ETL Job, first you need to click on the name of an existing ETL Job form the ***“System Panel”***. Then on the menu bar select **Edit** > **Delete** or you can click the delete icon showing on the window. Next a dialog box will show up and ask to confirm deleting the selected item. Choose ***“YES”*** to delete the ETL Job or ***“NO”*** to cancel the action.

Different job relation can be setup among jobs. They are Job Stream and Job Dependency:

* Job Stream is the mechanism to kick-off downstream job when upstream job is completed. If job B is a stream job of job A, then after job A (upstream job) is finish, job B (downstream job) will start to run automatically if no other Job Dependency is outstanding.

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 39 of 46 | |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

* Job Dependency is the mechanism in designing job flow to specify a pre-requisite condition for a specific job. This pre-requisite condition can be a job completion or an arrival of a file. Dependency not like a Job Stream that a job it can be run automatically when dependency condition is satisfied. E.g. If job A is a dependency job of job C, then job C must wait until job A is completed.

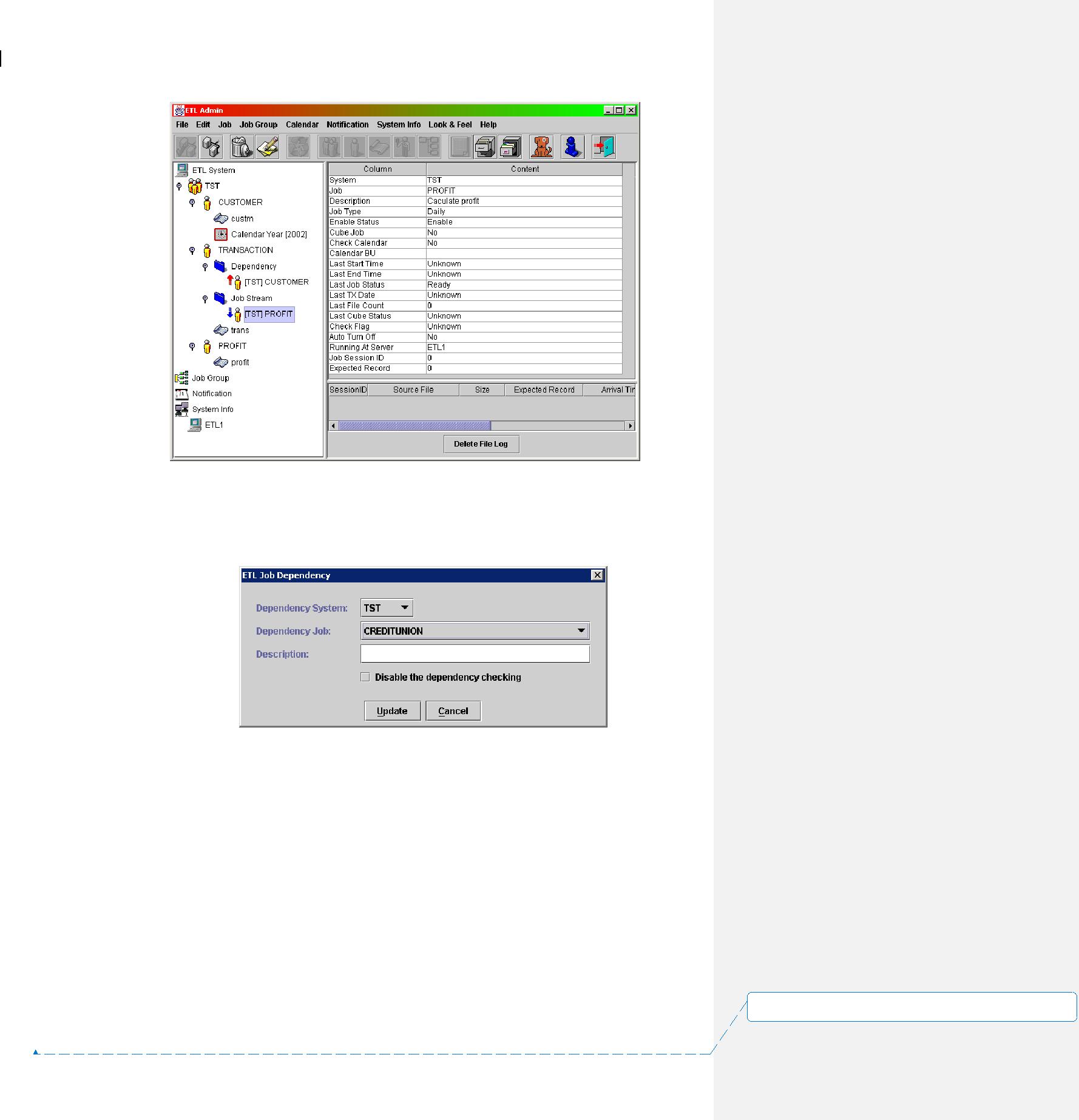
In EA, Job Stream and Job Dependency can be easily setup from ETL Automation Administration by the following step:

Job Stream Setting:

1. Choose the “Add Job Stream” icon in the ETL Administrator program and the “ETL Job Stream” window will be opened.
2. Choose system name from Down Stream System pull-down.
3. Choose the name of the down stream job from the selection list and enter a short description about the job.
4. Job Stream can be disabled by clicking “disable the down stream execution” option. It can be enabled at any time later by editing the Job Stream’s property.

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 40 of 46 | |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

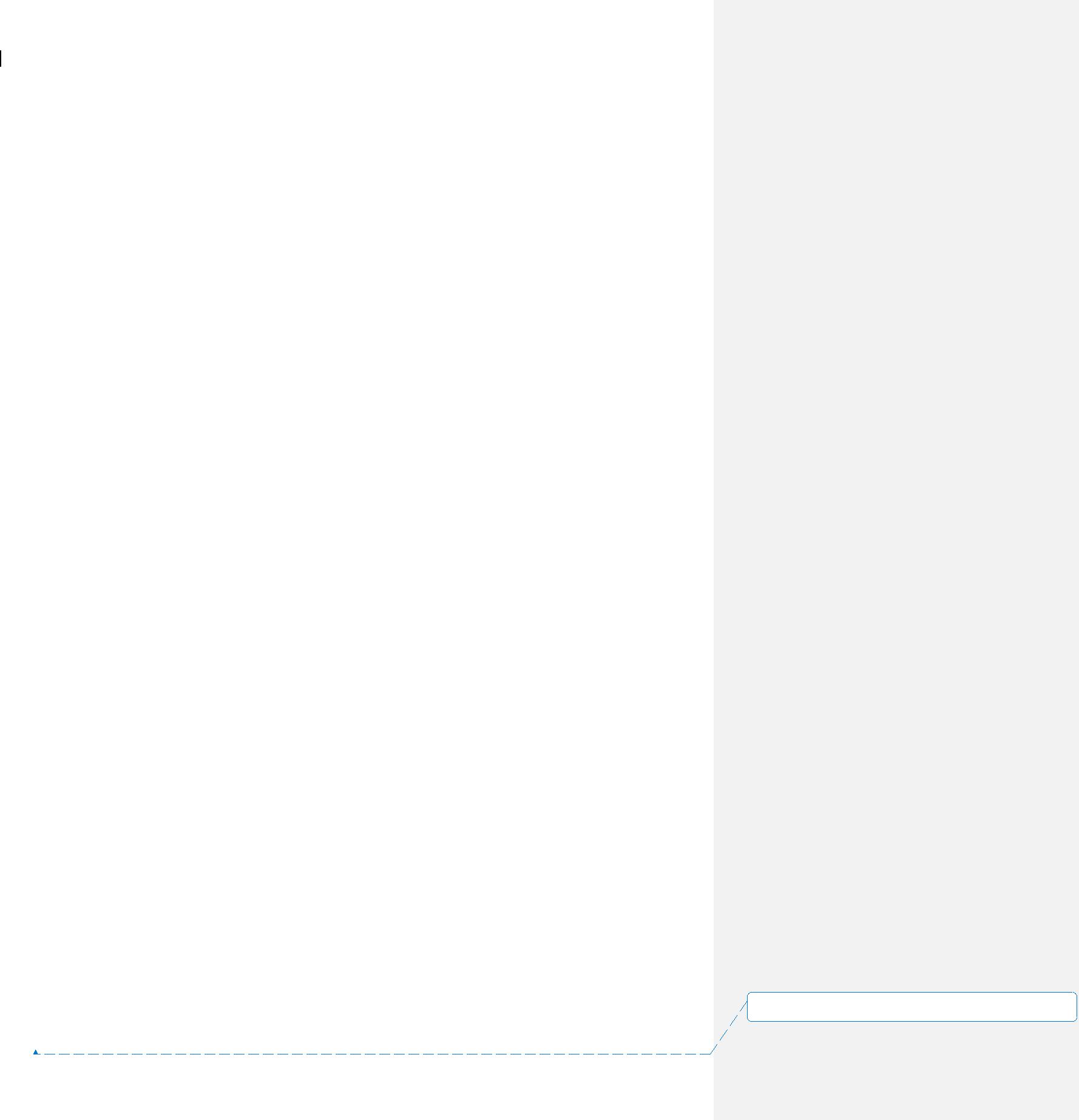
Job Dependency setting:

1. Choose the “Add Job Dependency” icon in the ETL Administrator program and the “ETL Job Dependency” window be opened.
2. Choose system name from Dependency System pull-down.
3. Choose the name of the dependency job from the selection list and input a short description about the dependency job.
4. Job Dependency can be disabled by clicking “disable the dependency checking” option. It can be enabled at any time later by editing the Job Dependency’s property.

Each EA job is invoked by a control file. A daemon program called etlrcv will monitor the RECEIVE directory (/opt/ETL/DATA/RECEIVE) to check if any control file is coming. If a control file is found, it will fork a process to invoke etlslave.pl in order to execute the ETL Job if the following criteria are fulfilled for the job:

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 41 of 46 | |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

1. Transaction date of the control file must be greater than or equal to transaction date of the job
2. Data file’s size and filename must be the same as the content in the control file.
3. It’s dependent jobs must been completed
4. Job script in Perl must exist in the directory as specified in job setting.
5. The job must meet it’s time window to be run. (I.e. we can set the job to be run between 0:00a.m. to 1:00a.m., so if the control file comes at 2:00a.m., the job is not allowed to run.)

ETL Automation acts as centralized job scheduling and controlling center for the whole ETL process. For details of operation and configuration, please refer to ***ETL Automation User Manual***.

**5.1.2** **Daily and Monthly Job Setting**

EA jobs are either invoked by supplying a control file or triggered by an existing job. However, the control file may arrive at a earlier or later time. It affects the actual job execution time.

We can configure job trigging method and running frequency by editing the job’s attributes in ETL Automation.

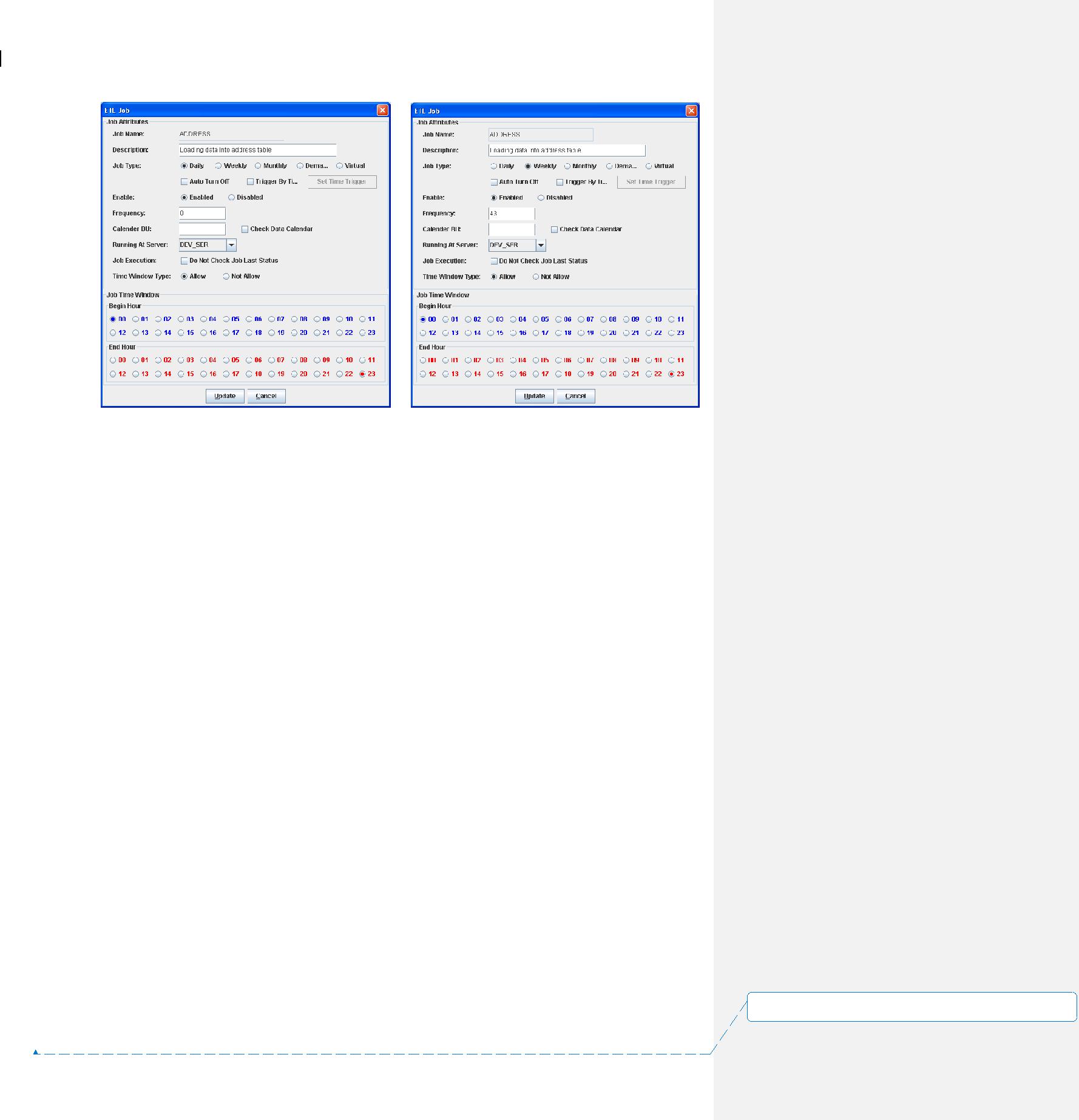
***Trigger by Control File***

By choosing different job type and assigning different value to the frequency field, we can control job to run if the transaction date of the control file match with the frequency setting.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

42 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

Different frequency code as follow:

0 means every day, -1 means the last day of month, 1 to 31 means the day number in a month, 41 to 47 means the day number in a week.

For example:

Suppose we only want the job running at every Wednesday, we can do it by choosing the weekly option and assign 43 to the frequency field. Also if we want to change the job to run at the end of each month, we can choose the monthly option for the job type and assign -1 to the frequency field.

Then when the control file arrived, EA will check the transaction date of the data file and design whether to run the corresponding job.

**5.1.3** **Environmental Parameter File**

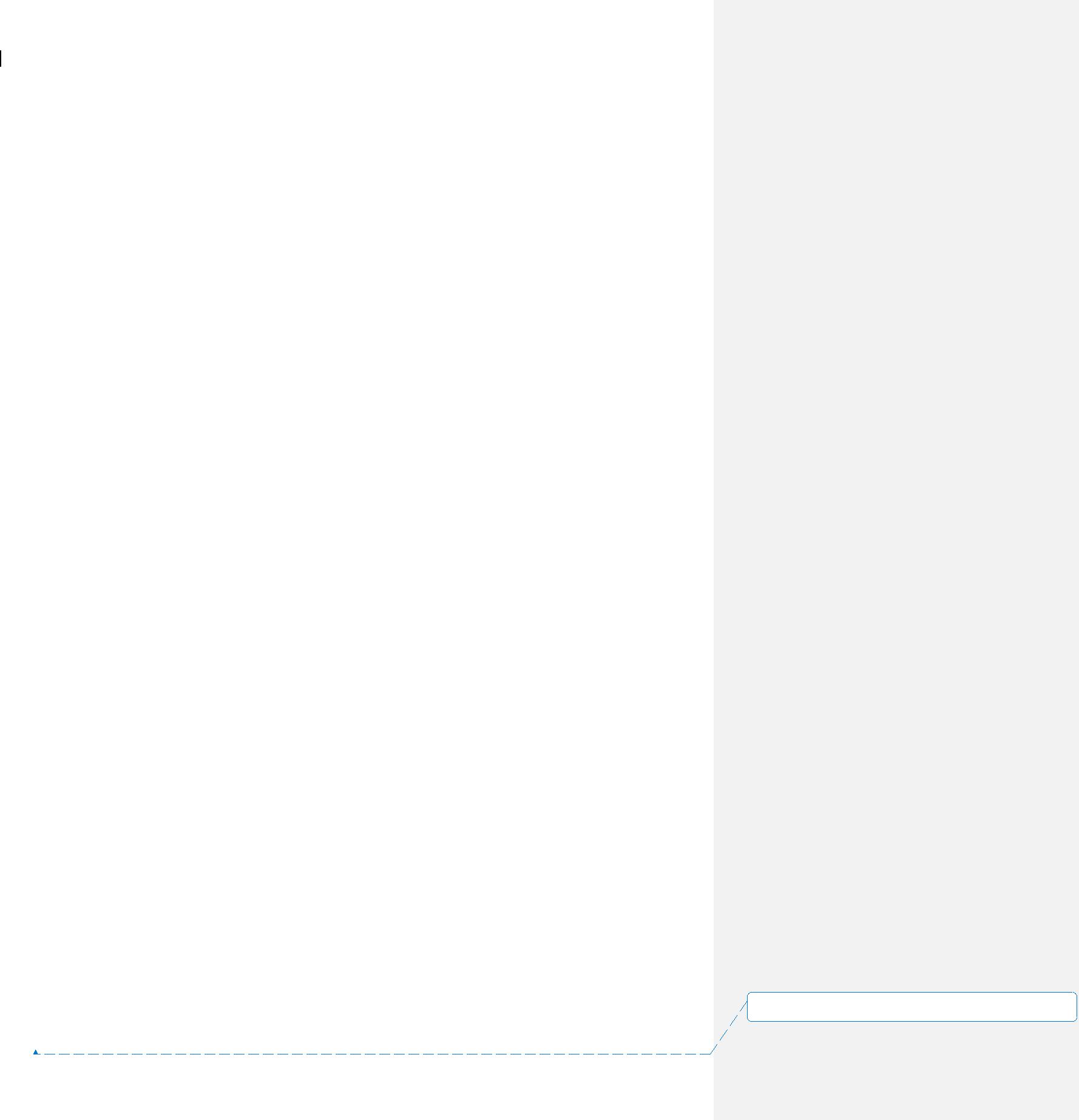
In ETL job coding, an environment parameter file ‘etlvar.pl’ will be used to control either development, UAT or production are being processed. It also helps to parameterize our job design to allow same piece of codes to work in different environments and enhances the script usability. In ETL Automation, the parameters in the parameter file will be extracted when kick-off the job and passing the parameter job’s internal variable for job use.

**5.2** **DATASTAGE OPERATION INTRODUCTION**

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

43 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

In ETL process, we will use DataStage to perform some of the data loading job as data reformatting task. DataStage GUI environment allows user to use drag and drop approach to develop job and generate FastLoad scripts for the data loading to Teradata automatically. This feature can help to speed up the development time and let developer more focusing on the job design.

DataStage can be divided into two parts – server side and client side. DataStage Server is the server application which responsible for executing job. There are four DataStage client applications used in ETL for building, controlling and monitoring job in DataStage. The four applications and their functions are listed as below:

1. DataStage Administrator – To perform administration tasks in DataStage. These include creating and moving projects, setting up DataStage users and setting up purging criteria.
2. DataStage Designer – A design interface used to create DataStage Jobs. Each job specifies the data sources (i.e. database, sequential file…etc), the transformation required, and the destination of the data which is the Teradata database in our case. Finally, jobs are compiled to create executables that can be run by the server.
3. DataStage Director – A user interface used to validate, schedule, run, and monitor DataStage server jobs and parallel jobs.
4. DataStage Manager – A user interface used to edit and view the contents of the DataStage repository.

To start DataStage client, we need to create a project from DataStage Administrator first. Project is used for storing DataStage jobs and user-defined components. After a project is created, we can create jobs in DataStage Designer and manage any DS job components in the DataStage Manager.

DS job requires to be compiled before execution, and then we can develop an EA job to kick-off by DS command line prompt ‘dsjob’.

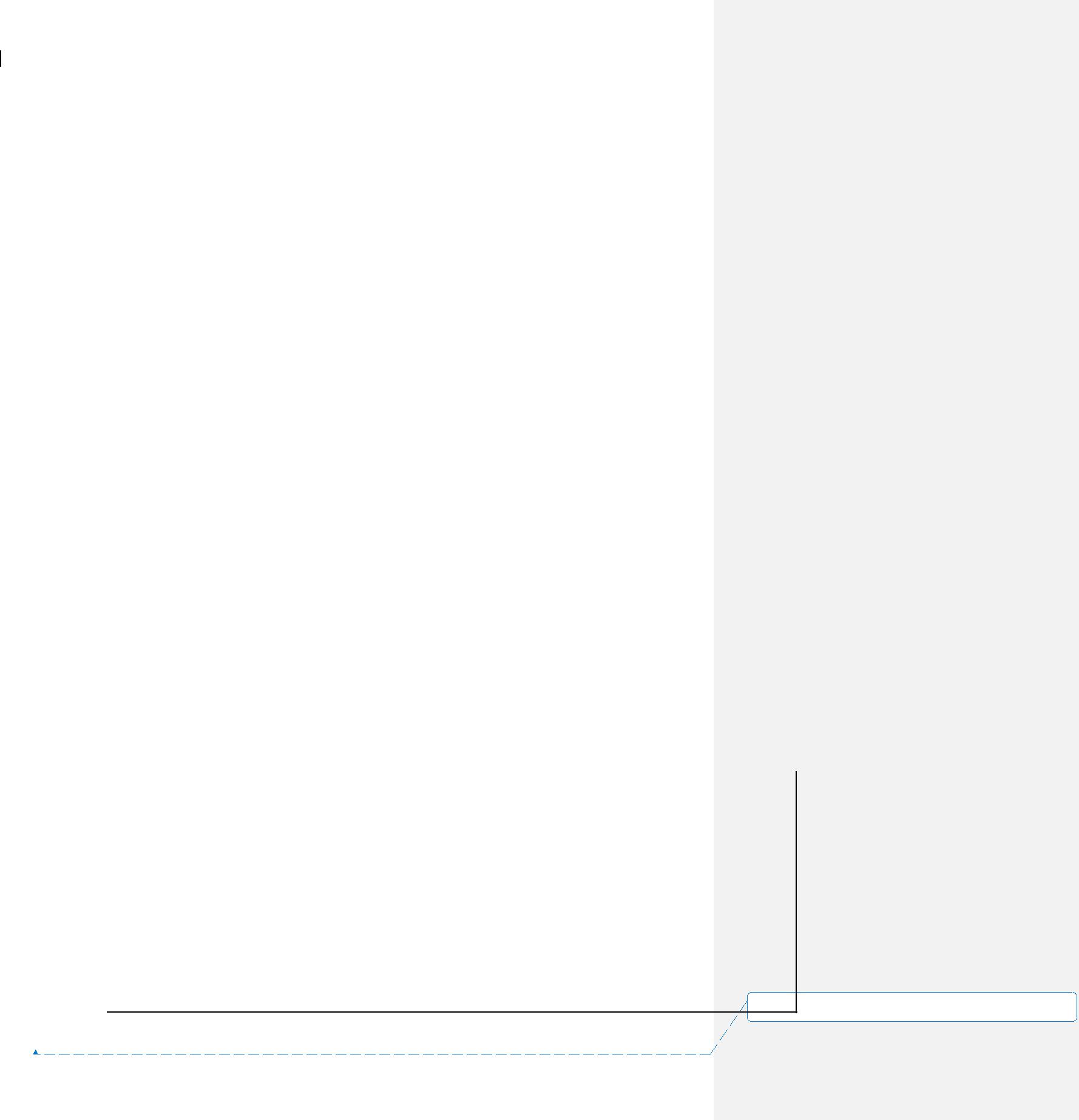
**5.3** **ETL PERFORMANCE TUNING**

There are several features that may help to improve ETL performance. The choices of implementing these features are subject to actual job performance and workload. There are two major Teradata features helping this.

0\_1.doc~~SMC-V ADW Project - ETL Technical Specification.doc~~ Page

44 of 46

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检查拼写或语法



ETL Technical Specification 15~~4~~ November 2005

**5.3.1** **Partitioned Primary Index**

For some large table like CDR tables, as well as medium size tables, Partitioned Primary Index (PPI) is applied to improve query performance.

Partition Columns should be determined during PDM design. For the re-definition of PPI on existing table, its structure needs to be modified by recreating its DDL. Then, the data is repopulated. Data rows inside the PPI table are ordered by Partition Number, and then by Row Hash (actually Row ID) sequence within the Partition. This design can help optimization for range queries to eliminate non-candidate partitions.

All history tables and transaction tables should have Partition Primary Index defined. History tables are generally partitioned by end date while transaction tables are partitioned by transaction date.

For the details on the choice of partitioned columns, please refer to the PDM design documents.

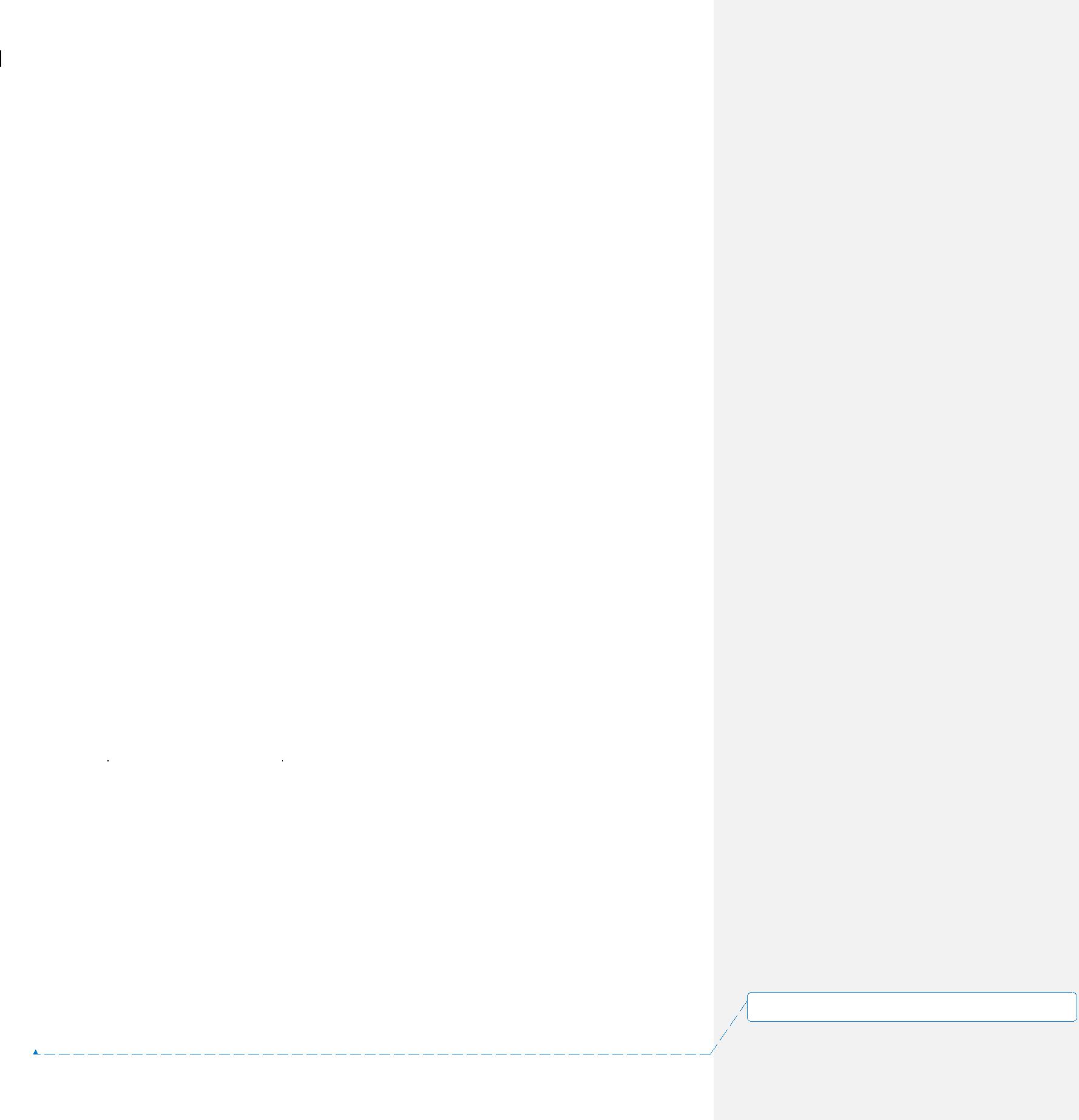
**5.3.2** **Secondary Index**

Secondary Index can be built on specific tables to give query alternatives on the data access path. This may or may not gain improvement on query performance. Therefore, this should be evaluated and determined case by case.

**5.4** **IDENTIFIABLE ERROR MESSAGES**

The following are messages that may send to HP Openview

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Check** | **Error** | **Severity** | | **Checking** | **Rule** |  | **Action** |  |  |
|  |  | **Phase** | **Condition** |  |  | **Target** |  |  |  |  |  |
|  |  | Pre-ETL | File not | Information | | File | Interface file is expected to be |  | Wait for an |  |  |
|  |  | Check | found at |  |  |  | pulled from a dir location, but not |  | interval and retry |  |  |
|  |  |  | expected |  |  |  | found yet |  |  |  |  |
|  |  |  | time |  |  |  |  |  |  |  |  |
|  |  | Pre-ETL | File not | Warning | | File | Interface file is expected to be |  | Send message to |  |  |
|  |  | Check | found after a |  |  |  | pulled from a dir location, and |  | HP Openview |  |  |
|  |  |  | number of |  |  |  | retry failed for N times |  |  |  |  |
|  |  |  | retry |  |  |  |  |  |  |  |  |
|  |  |  |  |  | |  |  |  |  |  |  |
|  |  | Pre-ETL | File set | Information | | File | Set of interface files is expected |  | Wait for an |  |  |
|  |  | Check | incomplete |  |  |  | to exist at a dir location, but not |  | interval and retry |  |  |
|  |  |  | at expected |  |  |  | found yet. |  | **设置了格式:** (中文) (不作校对), (其他) (不作校对),不检 | |  |
|  |  |  |  |  |  |  |  |  | 查拼写或语法 | |  |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | | | |  |  | Page | |  |
|  |  |  |  |  |
| 45 of 46 | | |  |  |  |  |  |  |  |  |  |



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ETL Technical Specification | |  |  |  |  | 15 | ~~4~~ | November 2005 | |
|  |  |  |  |  |  |  |  |  |  |
|  |  | time |  |  |  |  |  |  |  |
|  |  |  |  |  |  | | | |  |
|  | Pre-ETL | File set still | Warning | File | Set of interface files is expected | | | | Send message to |
|  | Check | incomplete |  |  | to exist at a dir location, but not | | | | HP Openview |
|  |  | after a |  |  | found and retry failed for N | | | |  |
|  |  | number of |  |  | times. | | | |  |
|  |  | retry |  |  |  |  |  |  |  |
|  |  |  |  |  |  | | | |  |
|  | Pre-ETL | Machine | Critical | File | The Ip address of the file | | | | Send message to |
|  | Check | cannot be |  |  | location is not reachable | | | | HP Openview |
|  |  | reached |  |  |  |  |  |  |  |
|  | Pre-ETL | Machine | Critical | File | Cannot login the machine to get | | | | Send message to |
|  | Check | login failure |  |  | the required file | | | | HP Openview |
|  |  |  |  |  |  | | | |  |
|  |  |  |  |  |  | | | |  |
|  | Pre-ETL | Failure in | Error | MD File | Validate head/trailer based on | | | | Send message to |
|  | Check | head trailer |  |  | file | | | | HP Openview |
|  |  | validation |  |  |  |  |  |  |  |
|  |  |  |  |  |  | | | |  |
|  | Pre-ETL | Failure in | Error | File | Produce an error when a rule is | | | | Send message to |
|  | Check | input |  |  | failed to pass in pre-process | | | | HP Openview |
|  |  | variance rule |  |  | checking | | | |  |
|  |  | checking |  |  |  |  |  |  |  |
|  |  |  |  |  |  | | | |  |
|  | Post-ETL | Failure in | Error | Table | Produce an error when a rule is | | | | Send message to |
|  | Check | output |  |  | failed to pass in post-ETL | | | | HP Openview |
|  |  | variance rule |  |  | checking | | | |  |
|  |  | checking |  |  |  |  |  |  |  |
|  |  |  |  |  |  | | | |  |
|  | ETL | Job script | Error | File | Failure in a job run | | | | Send message to |
|  | Check | failure |  |  |  |  |  |  | HP Openview |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

**设置了格式:** (中文) (不作校对), (其他) (不作校对),不检

查拼写或语法

|  |  |  |
| --- | --- | --- |
| 0\_1.doc | ~~SMC-V ADW Project - ETL Technical Specification.doc~~ | Page |
| 46 of 46 | |  |