eService Online 5月22日发布事故总结和改进工作计划

2019年5月22日，我们发布了eService Online和eService相应的新的版本。发布后，用户反映eService Express中不能正确显示出Job List。5月23日晚间，7个现场作业组接连发生同样的问题。5月24日对eService 发布进行了紧急回滚，人工干预所有现场作业计算机的回滚安装和数据目录的清理。其后又发现Unit Number不能正确提取，导致监控不能正确使用。针对程序中存在的关键bug，制定了Dispatch在eService 操作中的附加操作和Supervisor在现场的附加操作，以保证整个系统的正常工作。

在完成应急处理工作后，开始对eService Online, Sanjel Eservice和Eservice Express进行全面的问题排查。

在排查过程中，Dispatch不断报告eService Online的功能缺陷和反常的行为。开始了对问题的修复工作。

6月17日，确认新版本eService R6.6.1和eService Express 1.2.0.38通过现场试运行和验收测试。

6月18日，主要问题基本理清修复，发布了新版本的eServiceOnline 6.6.1。但有些问题出现在eService的代码中，因为目前无法更新eService的WCF服务，只好通过在数据库中加入trigger，强制修复数据错误。

此项修复工作，需一直持续到下一个版本的eService发布。

事故原因总结：

在这个新版本的发布中，我们对整个EService 系统进行了较大规模的重构，引入MDD生成的代码。在重构过程中，由于代码的移动和调整，出现了关键逻辑代码的误改和丢失。从而导致重要逻辑错误和数据缺失。

在同时开发的新功能中，新实体和原有实体之间的关系理解不清，实现上的不足，导致功能不够健壮。

整个系统规模过大，在重构过程中，系统边界不清，相互依赖不容易理解，是客观上的重要原因。

开发员工作不够细致，大量的低级错误导致严重后果，测试环节不足，这些是主观上的主要原因。

重要错误：

1. 在重构过程中引入了DrillingCompany这一实体，将原Company实体中的混淆的概念进行理顺。通过接口协议别名等方式，避免对原eService代码的改变。这其中导致了以下问题。
   1. 测试时重点侧重eServiceOnline和eService的测试，忽略了eService Express。导致eService Express 不能正确解析eService生成的XML文件。
   2. 违反了面向开发的SOLID原则之一的OPEN-CLOSE原则，OPEN for extension, Close for modification.
2. 在RigJob相关代码迁移重构的过程中，在eService中Job Service中，第一次重构将JobLifeStatus转为实体时，Completed状态是8，却写成了9。在第二次重构改回到Enum的时候，把状态按9写成了Deleted。原来的代码竟然就注释在这个错误的代码上方。
3. 在Blend Breakdown计算时，对一部分代码进行了拷贝，但不知何故其中将一个条件中的Prehydrated改成了preblend。导致后面全部错误。

改进计划：

1. 加强测试工作。
   1. Sanjel改变了发布流程，增加了现场试运行环节。在eService/eService Express的的发布过程中，先推送到一个作业团队进行试运行。试运行成功后，推送到三个作业团队进行验收测试。验收测试通过后，全面推送。
   2. 开发团队要恢复单元测试和系统测试的自动化，逐渐转变为测试驱动开发。
2. 完善架构设计，明确开发规范。
   1. 对当前架构设计进行复查，并进行重构。
   2. 详细定义架构的层次和组件职责，制定开发规范
   3. 强化系统边界，避免交叉依赖。
   4. 强化子系统的独立性，明确系统交互定义。
3. 提高开发人员的技术水平，强化专业素养
   1. 针对出现的问题进行自我反省。
   2. 强化开发原则的培训和习惯养成。
   3. 加强技术培训和学习
4. 改善开发过程
   1. 复查需求和设计沟通流程
   2. 构建合理的文档系统，强化文档的相关性和可溯性
   3. 严格执行开发过程管理。