|  |
| --- |
|  |
| Sprint Memo |
| From 2019-06-24 |
|  |
| ©MetaShare Inc. |
| **6/25/2019** |

|  |
| --- |
| 本文记录Sprint的每一天早会的备忘录 |

目录

[1. 引言 1](#_Toc12342247)

[2. 演进式开发过程 1](#_Toc12342248)

[2.1 2019-06-24 1](#_Toc12342249)

[2.1.1 工作安排 1](#_Toc12342250)

[2.1.2 早会记录 1](#_Toc12342251)

[2.1.3 下班前工作汇总 1](#_Toc12342252)

[2.2 2019-06-25 1](#_Toc12342253)

[2.2.1 工作安排 1](#_Toc12342254)

[2.2.2 早会记录 1](#_Toc12342255)

[2.2.3 下班前工作汇总 1](#_Toc12342256)

# 引言

Eservice项目从2019-06-24开始对开发过程进行优化，本文包括每天早会后对早会上讨论事情的汇总，包括早会讨论了什么事情，每天做什么改进，接下来要安排什么会议讨论什么事情等等，以供大家查看。

# 演进式开发过程

## 2019-06-24

### 工作安排-Adam

对Bin的需求进行整理，把Bin的管理做为一个独立系统。所有其他系统对Bin的关系或操作都以交互的方式进行。  
很多功能已经实现了，通过整理我们可以发现原来的实现的问题。所以不要被现有的实现给牵着走。  
第二部分，根据分析整理的过程，思考应该怎样记录和描述这些需求和分析。演进式开发过程管理系统应该有哪些实体？  
两个项目同步走，记住第二个项目顺势而为，不需要想太多。

### 早会记录

### 下班前工作汇总-Joe

Hi Adam，

我更新了今天的任务文档，今天时间比较不是很充足，会有些粗糙。

不过我目前的思路是以Task为基准对改动进行追溯，每一次的改动都先在Tesk上体现，因Tesk的范围较小，可测试性和控制性强。

在创建文档时，Requirement和Backlog作为同一个文档存在。

TastCase和Task和改动将存在于同一个文档，文档顶部会标注出TastCase所属的BackLog的文件名。

目前只是初步想法，还请您多指教。

SVN地址：<https://117.34.118.114:8443/svn/SanjelDocuments/trunk/Requirements/Evolutionary> development process

## 2019-06-25

### 工作安排-Amam

使用5W2H分析的时候， 应该是一事一分析。而不是把一个系统或是域的所有5W2H混在一起。而且你的出发点是实体本身，而不是Actor.

例如，大一点的需求

Bin Management.

Who – Dispatch

What – Track Bin movement.

When – A job needs blend to be shipped to wellsite (Rig is mandatory for a wellsite) beforehand, a bin(s) is needed to assign to a rig, so blend can be hauled to the bin. After the jobs are done with the rig, we need to haul the bin back to yard from the rig.

Why – Dispatch will have the visibility of bin’s availability, so the job needed blend can be prepared in advance.

Where – Bin may be moved from one Service Point to another Service Point, from Service Point to rig, from rig to home Service Point or another Service Point.

How – Dispatch select a bin and select rig/ Service Point to complete the assignment.

How Much – L (Here can be the high level estimate for the effort needed on this level, normally we use t-shirt size to mark it,  XL, L, M, S, XS.  If it is L, or XL, normally we have to breakdown further to make things more clear.)

因为问题比较大，不太可能一下描述清楚。因而需要分解成小的事件，进一步描述清楚。在分解之前，需要根据已知需求，对识别出的实体进行准确定义。

* Bin
* Rig
* Service Point
* Job
* Blend

Breakdowns

1. Assign A Bin to A Rig.
2. Release A Bin from A Rig.
3. View Bins

……

划分后的小事件进行进一步信息澄清和描述， 进一步识别出实体和实体状态

Assign A Bin to A Rig

Who – Dispatch

What – Assign a bin to a rig.

When – Bin is available, rig is active (? – if dispatch can assign a bin to a down/inactive rig for planning purpose, need to confirm)

Why – Will schedule product haul for a bin for job preparation.

Where – Bin may have assigned to another rig, need to release and assign to this rig. Bin may be in yard and ready to assign. (Is there other place the bin located?)

How – Dispatch select a bin and select rig to complete the assignment.

How Much – M

* Bin
  + Location
    - Rig Location
    - In Yard
  + Status
    - Assigned.
    - Available
* Rig
  + Status
    - Active
    - Down
    - Inactive
* Product Haul

以上只是一个大概分析流程的示例,  随着后面的信息不断获取，可以不断补充。主要是识别实体，实体间的关系及其约束，实体的状态及转换的条件。

所以在你提交的文档中，实际上是把反向后的需求逻列出来又按5W1H进行了排列。所以是不对的。现在需要你假装从需求获取，到分析，到设计的过程依次进行迭代。

敏捷过程的获取用户故事的一个简单的标准模板是：

As .., I would like to .., So that….

如果套用5W2H的话， 是这样的

As [Who], I would like to [What, Where, When, How], so that [Why , Result].

比如用户说一句，我要能给Rig分配一个Bin，你刨根问底，应总结出类似的一条需求。

做为一个Dispatch, 我要根据一个Job的需要，在Job开始前把一个可用的Bin分配给Job所在的Rig, 所以可以把Blend提前运到Bin中备用。

也就是说，每一个问题都要用5W1H去提问题。How Much可以是用户对重要性的评估，也可以是你对问题的大小评估， 提问题时不是必须的。

你再重新对需求整理一下。这中间你会产生新问题，你可以问我，进一步获取信息。

继续努力。

### 早会记录

### 下班前工作汇总-Joe

今日的记录已经更新至Evolutionary development process文件夹，我在里面新创建了两个文件夹，分别存放Bin的需求和EDP的记录。

今天我领会了一下您的意思，通过OnLine了解了Bin的基础需求之后，完全将Bin作为一个独立的系统进行了分析，并进行了记录。

对于EDP的记录，我把在分析Bin过程中的想法也进行了记录，请您查看。

## 2019-06-26

### 工作安排-Adam

你不是被传统流程框住了。你是被5W2H给框住了。谁说分析要用5W2H了？解释方法论了吗？

我建议你回归传统方法，也别用我的例子去套。用你认为掌握的最好的方法去分析。我们慢慢长出来，而不是框出来。

### 早会记录

### 下班前工作汇总-Joe

Hi Adam,

因为年会的缘故，我今天的时间基本都投入到了ERP的文档整理上，今天只更新了BinRequirement-2019.06.26.txt

主要使用了我最常用的需求分析方式，即获取Requirement，通过沟通获取业务参与人员都有哪些，业务流程是怎么样的。

按照我接触最多的分析流程，获取到这3类信息后，就直接开始对系统进行设计了。

根据我在公司学到的UML相关知识，这样可能不太严谨。我想先放空自己，用我最熟悉的方式来整理需求，之后发现问题在回溯。

注：如果是一个全新的东西我做起来应该会更熟悉，现在Bin已经实现的部分放在我脑子里，经常不自觉的进行套用。

## 2019-06-27

### 工作安排-Adam

因为之前的开发中信息很零乱，虽然开发任务中得到了整理，但是可溯性差，现在找不回去了。现在要你做的是相当于一个全新的东西，把涉及到Bin的需求都找到，放到一个文档里，我们称其为原始需求(Raw Requirement)，这一步叫做收集需求（Requirement Collection）。

现在可以告诉你，目前Bin的实现是有问题的，所以不要管它。

下一步才是需求分析(Requirement Analysis)，涉及到昨天讨论的方法。对需求进行整理，分割、联结，发现缺失的信息和矛盾的信息，返回第一步向用户提问题，来补齐需求信息。也许需要反复澄清很多次。直到所有的功能需求都有充分的信息支持。

这一步的产出，应当是较高层次的用例。粒度比较大。用例图、活动图是主要工具，因为目前没有合手的工具，你不要用VISIO去画，太浪费时间，可以在纸上画画就行。

我看了一下文档，Business Process中的断言并没有信息支撑。 比如：

“3.1 OSR根据Job的情况申请使用Bin”，不知道你从哪儿获得的这个信息。

“ 3.2 Dispatch为该Job下的Rig分配Bin”  是不准确的， 因为需求中说了，Bin是分配给Rig的，会跟着Rig走，分配时没Job什么事儿，和Job产生关系是后面的事儿。“分配”这个词是不是不准确？英文是Assign, 中文找个什么词更准确？

其他三个也有问题，你需要用户提供的信息支持，不清楚就要问。

下一步，你把之前所有文档中，用户需求信息都收到这个文档中，有些信息是你自己反推的，要特别标明。

然后再分析整理。

### 早会记录

### 下班前工作汇总-Joe

Hi Adam,

我今天整理了项目文件里目前能找到的关于Bin的需求，并进行了记录。

分析的部分只开始了一点点，但是在整理Bin的原始需求后，我感觉思路清晰起来了。

## 2019-06-28（年会中）-2019-07-01

### 工作安排-Adam

为了摆脱旧的文档框架的影响，我在根目录下有一个EDP文件夹，把新文档都移到这里。随着想法一点点成形，新的目录结构会逐步形成。

我们讨论的如果写原始需求，这些都是演化开发方法的系统需求。你原来写的那些太多了，我们还没到那儿呢。你先就原始需求获取这一步，去写它的原始需求。

在BinRequirements 2019.06.27中， 我加了些注释，你看一下。另外，不用每天起个新的文件，在原来的上面改，然后提交就可以了。能过SVN版本比较，很容易看出来改了什么。

版本间的变化，是EDP中很重要的思想，所有的后续的开发过程都是这个变化来驱动的。这一点你要慢慢体会。

获取原始需求后，你开始分析，你的Requirement Analysis部分跳跃太大，还是从现有系统来，没有假装从头演化。分析的开始，是形成概念模型，这是一个实体初步识别的过程，开始发现（Domain）域，（Entity）实体，实体间的（Relation）关系，有关系才有实体间的属性（Property）。 实体的（Status）状态，状态的改变，状态改变了，才有了动作（Action）。有了改变，才能识别出操作者（Actor）。

把动作（Action）描述出来，才有了(Activity)活动。然后是状态改变的条件。有条件才有了逻辑（Logic），也叫工作流(Work Flow)。

至此还都是系统无关的描述。

你继续做以上的事情。

### 早会记录

### 中途沟通

Q：

Hi Adam,

我有一些问题：

1、现在对于Bin的管理是怎么做的，Bin是否属于消耗品？

2、新的Bin被采购回来后，我们会在系统上对其进行添加，之后通过OnLine进行调用吗？

3、Bin是否牵扯到维修、检查等异常状态？如果是，那是否需要显示出来？

4、关于Bin的管理，这次分析需要考虑到吗？

5、Bin的Ownership是用什么信息来表示的，所属公司，所属地区，还是别的什么？

目前的问题就是以上5条，需要您的解答。

A：

1. 现在对于Bin的管理是怎么做的，Bin是否属于消耗品？

Bin不是消耗品，它是很大的贮藏罐，可以理解成是临时仓库。

1. 新的Bin被采购回来后，我们会在系统上对其进行添加，之后通过OnLine进行调用吗？

新的Bin是由固定资产管理部分在ERP中填加，对于我们的系统来说它是主数据，Online没有管理这个数据的权限。一个BIN被加于到ERP，eservice online或eService 就可以看得到。

1. Bin是否牵扯到维修、检查等异常状态？如果是，那是否需要显示出来？

会，有这些状态，但目前需求中没有。

1. 关于Bin的管理，这次分析需要考虑到吗？

我们只考虑Bin的动态管理，在哪儿？装了什么？装了多少。

1. Bin的Ownership是用什么信息来表示的，所属公司，所属地区，还是别的什么？

所属地区

Q：

Hi Adam,

前提：

关于Bin的管理，这次分析需要考虑到吗？

我们只考虑Bin的动态管理，在哪儿？装了什么？装了多少。

根据您的描述，我们需要考虑的是Bin在哪？装了什么？装了多少？

目前Bin在哪是比较容易实现的，关于装了什么和装了多少的问题，可以实现，但是会增加很多客户的操作。

我想了解一下Bin和Blend的关系：

1. 一个Bin可以装载多个类型的Blend吗？
2. 在客户的工作流程下，一个Job需要Bin时，会规定Blend必须怎么装载在Bin上吗？还是说Blend装载在Bin的相关规则是人为控制的，也就是由装载人员自己根据经验进行。

我在OnLine上看到Blend有很多类型，但是Bin只有两种类型，所以这个部分有点疑惑。

如果不用控制Bin里Blend的类型，那么我将会将一个Job下所有Bin的装载物都直接获取Job所需的Blend，不对单独的Bin进行区分。

如果需要控制Bin里的Blend，那么我将会在Dispatch为一个Job下的Rig分配Bin时，获取该Job的所有Blend，需要让Dispatch选择Bin需要装载哪些Blend。这样就可以解决用户想要看到Bin装了什么和装了多少的问题。

我在网络上没有搜到相关信息。所以请您讲一下，我就对Bin和Blend有概念了。

### 下班前工作汇总（2019-07-01）-Joe

Hi Adam,

今天的需求整理更新在项目文件里了，不是很多，请查看。

现在我的思路感觉慢慢清晰起来了，但是还有很多需要问的东西，主要在于业务流程部分。

## 2019-07-02

### 工作安排-Adam

1. 开发任务

我们七一国庆放假，  
  
你们应该也从ERP项目回归了。  
  
接下来两周，你们要完成rigboard 的bin和tools的开发，需求Joe还在整理中。  
  
今明两天，如果有时间投入，做以下事情：  
  
给eService solution 加一个新项目，eServiceR6Api, 然后反向查找eService online 中，对eService 的操作，在Api中进行实现。目的是将eService 和online 对实体库的依赖彻底分开。  
先分析设计，方案经我审核后，才可以动代码。  
  
为了分离，需要双方共同依赖的集成实体库，但要记住它只存在亍集成接口层面，不可以过度依赖。

1. 需求任务

我知道你们最近在赶年会的进度，所以很多事情你无人可问。接下来，你可以多问问毕欣欣有关业务方面的知识。另外有一些文档中有相关的知识，你要去把它们找一下，读一下。如在SanjelDocuments\Requirements\Phase 5\第五期需求整理  的文档中第六页的图片中，中间一些竖着的罐子就是Bin的一种。

你还有一个任务就是把这些文档按EDP的过程重新整理。

你先把背景知识补一下，有问题再沟通。

### 早会记录

### 下班前工作汇总

1 开发

Hi Adam,

       今天工作内容汇总：

1. 将eServiceOnline项目的trunk路径代码更新到了最新版本。
2. 根据通话内容先尝试重构了从online创建BinSection的功能，代码提交在trunk路径上，请您Review下实现思路是否正确。

2 需求

Hi Adam,

今天Bella给我讲了一下Blend和Bin的关系，根据目前了解到的业务背景，我更新了关于Bin的需求，已经上传了。

请您查看。

谢谢！

## 2019-07-03

### 工作安排

1. 开发任务

Hi Bright,

从你的描述上看，你的思路是错的。

昨天通话时，我主要说了两件事情：

1. 识别eServiceOnline对eService的调用，这部分应当是一个文档。
2. 给eService solution 加一个新项目，eServiceR6Api, 然后反向查找eService online 中，对eService 的操作，在Api中进行实现。

而且强调，我们现在只动eService，消除它对MDD生成的实体库的依赖。

我看了一下代码，存在以下错误：

1. 你的动手方向完全是反的，你并没有去理解eService Online对eService的访问需求。
2. 我说了集成实体需要不依赖于任何实体，它只是定义了集成的接口，而且集成的方式是用Web API。也就是说，是通过HTTP通信来进行耦合。这个实体实际上可以不要，我们完全可以通过JSON对象进行交互。这个实体存在的唯一意义就是强类型。所以它的构建首先是空的，然后根据识别的出的必要的字段加进去，它应当尽量是一个平的结构。我还给你举了Inthinc API作为第三方API的例子，它只是给你API文档，其中有JSON对象的结构。

反观你在eService Online中定义的实体项目，它是给谁用的？如果是给Online用的，它应当是eService来定义实体。想象一下，Online是要调用eService 的API，必须得按eService 的要求去调用，不能自说自话。

1. 这个集成实体的项目是不能和Online项目混在一起，也不能和eService 混在一起，否则就会和Solution绑定，部署的时候就会出问题。
2. 集成实体是要求是支持JSON的，你把WCF contract的一些属性引了进来，加大了依赖。
3. 你重复了所有对象的层次结构，BinSection里有Bin，Bin还有BinType。这不是最小集。BinSection中的RootId也不是集成的交互语言，RootId是系统内置属性，相当于是private，对于两个系统间的交互界面是不会暴露出来的。

BinSection不是一个好的选择，因为JOE在分析Bin的需求，我们要对这部分进行重构。用BlendSection来做。它目前是只读的。

按以上要求，完成以下工作：

1. 昨天的代码回滚重做。
2. 分析当前eServiceOnline对eService 的直接调用，查找出所有调用接口。这些接口基本上都在eservice的MicroService中定义过了。对每个调用接口，分析最小调用的参数和最小返回值。形成文档。
3. 创建一个新的SVN项目，叫SanjelIntegration Library。下面以后用于保存所有用于集成的库文件，每个项目的库文件单独创建一个Solution。如eServiceIntegrationLibrary。在下面加一个项目叫做WebAPILibrary。它不得有对任何其他Sanjel库文件的依赖。以后其他项目的集成库文件项目按这个模式统一执行。生成的Assembly文件统一输出到SanjelCommonLibrary\Lib\IntegrationLibraries下面。这么做的原因是为了让你们不那么容易地产生错误依赖。记住
4. 给eService solution 加一个新项目，eServiceR6Api。这个项目依赖于SanjelCommonLibrary\Lib\IntegrationLibraries\eServiceIntegrationLibrary\WebApiLibrary.dll。实现BlendSection的相关接口。命名空间就用Sesi.eServiceIntegrationLibrary.WebApiLibrary
5. 今天不许动eServiceOnline与此相关的代码。明天审核后再继续向下做。

今天Bright完成以上工作。完成后如果有时间协助以下工作。

Colin今天在eServiceOnline上做下工作。

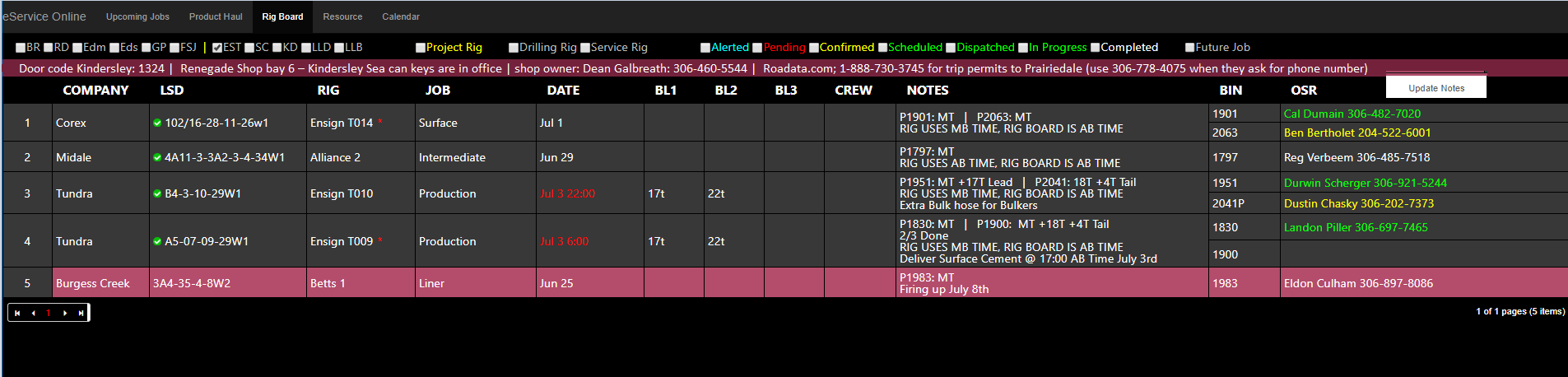
任务一：

加一个独立页面District Job Board，它是RigBoard的只读版，要求如下：

1. 在访问URL中加入Service Point Id做为参数
2. 每次访问只显示一个Service Point的RigBoard页面，相当于只勾选了相应一个Service Point，另外Future Job也默认勾选。
3. 不显示上面的过滤条件，除了Job Status的那一组，它只是保留下来做为颜色说明的。
4. 所有的Context Menu都不需要。
5. 可以在RigBoardController代码中实现，增加一个带参数的index方法，但不要改动任现有RigBoard 部分的代码。目前可以拷贝GetPagedRigJobViewModels方法生成一个新方法，把一些参数改成固定参数，在RigJobViewModel的列表组装完成后，再遍历一下，把所有的ContextMenu全部去掉。这样可能是最简单的。切记，不可对现有代码进行重构和改动。

任务二：RigBoard中增加District Notes功能

1. 显示位置：过滤条下方的隔离条。如图所示：



1. 点击左键，弹出编辑菜单，”Update Notes”。点击菜单，弹出编辑页面。使用我们Notes的统一实现模式。
2. 只有选择一个Service Point的时候显示出来，多于一个Service Point或是没有勾选任何service point都不显示
3. 任务一中页面不显示这个Notes。

有不清楚的地方请及时沟通。

谢谢，

Adam

1. 需求任务

Hi Joe,

在业务分析和系统分析过程中，最重要的一点就是要建立正确的概念(Correct Concept)，而且第一个要解决的问题是无歧义语言(ubiquitous language) 。

关于ubiquitous language这个词，在中文翻译的书中往往称为“通用语言”，这是不准确的，让我们来看看Martin Fowlor大师是怎么说的。

Ubiquitous Language is the term Eric Evans uses in [Domain Driven Design](https://www.amazon.com/gp/product/0321125215?ie=UTF8&tag=martinfowlerc-20&linkCode=as2&camp=1789&creative=9325&creativeASIN=0321125215)https://www.assoc-amazon.com/e/ir?t=martinfowlerc-20&l=as2&o=1&a=0321601912 for the practice of building up a common, rigorous language between developers and users. This language should be based on the [Domain Model](https://martinfowler.com/eaaCatalog/domainModel.html) used in the software - hence the need for it to be rigorous, since software doesn't cope well with ambiguity.

Evans makes clear that using the ubiquitous language in conversations with domain experts is an important part of testing it, and hence the domain model. He also stresses that the language (and model) should evolve as the team's understanding of the domain grows.

ubiquitous language是Eric Evans在领域驱动设计中使用的术语，用于在开发人员和用户之间构建通用，严格的语言。 这种语言应该基于软件中使用的域模型 - 因此需要严格，因为软件不能很好地应对模糊性。

埃文斯明确指出，在与领域专家的对话中使用无歧义语言是测试它的重要部分，因此也是域模型。 他还强调，随着团队对领域的理解的深入，语言（和模型）应该演化。

所以我觉得无歧义语言更为合适，所谓演化，就是当前无歧义，但是随着时境(Context)的变化，也会产生歧义。这时就需要改进定义，消除歧义。

关于时境(Context)这个翻译我们下回接着说。

以上很重要。今天说这个事情，是关于文档存放位置的问题。你把新的文档都放在了Requirements的文件夹下面，这在概念上本身就是不对的。而且Requirements这个文件夹是旧的开发过程的一部分，也是文档可追溯性差的原因的一个重要部分。我在之前的邮件中提过让你把文档移到EDP文件夹下面有一个干净的开始。而且在这个目录下，它的结构也会随着我们构建开发过程去不断演化。

消除歧义不只在需求定义上，也需要贯彻在开发过程中的每一个细节中。

请根据以上反馈调整文档的存放位置。

### 早会记录

### 下班前工作汇总

Hi Adam,

       以下是今天的任务完成情况：

1. 新创建的svn项目已提交，项目路径 <https://117.34.118.114:8443/svn/SanjelIntegrationLibrary/>
2. 在EService中新加了eServiceR6Api，实现了GetBlendSectionsByCallSheetId接口
3. 整理了部分接口文档，提交在 <https://117.34.118.114:8443/svn/SanjelDocuments/trunk/Projects/I-Implementation/eServiceIntegrationApi.docx>
4. 新增加District Job Board功能已完成，代码已提交
5. District Notes功能完成了部分，更新了eServiceMDM项目、库依赖文件、数据库升级脚本。

谢谢，

Bright

## 2019-07-04

### 工作安排

接口文档反馈已经提交。

Colin今天继续完成 Online的功能，然后做以下几个任务：

1. Resource board/Crew Board – Add one menu item in context menu on Worker column – “Remove all workers”. This is for easy removal of all workers instead of removing them one by one.
2. Rig board – Open a larger window when entering notes. NOTES输入框初始尺寸太小了，虽然可调，但用户体验极差。将其调为横向三倍，纵向两倍的尺寸。

IntegrationLibrary项目输出路径用的是绝对路径，我编译也不好用。Assembly命名不合规范。Bright修正以上问题，这个工作暂停。让我再想一想。BlendSection可能有点难，做第一个迁移示例不太合适。它应该是有结构的。你定义的实体把它下面的BlendAdditiveSections给丢了，也就是说，如果你切换了，马上就会出错。主要问题是你没有对接口需求进行认真分析。eService提供的接口是为online服务的，所以online中对这个实体的使用是需求。这后面还涉及到整个计算部分的重构。

Bright今天继续Joe的分析，业务需求基本上差不多了，你可以对照己经有的实现，开始详细定义Backlog（Test Case 形式）。

谢谢，

Adam

### 早会记录

### 下班前工作汇总

    今天工作内容汇总：

1. Colin完成了online上邮件中的任务，代码已提交。
2. IntegrationLibrary项目输出路径和Assembly命名已修改，相关项目的引用已替换。
3. 更新了bin的需求文档，需要您看一下。

## 2019-07-09

### 工作安排

Hi Bright,

Bin的需求文档已经反馈和更新。

Test Case 可以，请继续。

目前需要加入系统分析的部分，加到同一个文档中。包括：

1. 实体的识别和关系识别， 类图。但要用文字来描述， 用MDD字段的方式。实体的定义，实际上是之前讲过的无歧义语言。识别的，需要使用它与用户进行准确的交流。
2. 过程的识别。活动图。但要用文字来描述。相对于业务场景（用例， USE　CASE）。所有实体要严格使用上述定义。
3. UI　Prototype设计。注意积累UI中实体的定义，并结合实体定义进行准确说明，如Service Point下拉框。

这三部分在分析过程中应当是与业务需求分析同步进行的，但Joe进行分析时，这部分是缺失的。在需求获取和分析过程中，UI Prototype是一个很好的工具，使用户得到直观的感受，并易于得到反馈。这是一个统一的迭代过程。

相应可以开始MDD更新和代码开发工作，但一定是Test Case驱动的任务。

这部分开发过程上的工作比较多，请多用心，Joe回来后，将规范和心得传授给他，并总结到开发过程的文档中。

谢谢，

Adam

### 早会记录

### 下班前工作汇总

1. 今天修改了SanjelData的所有dll输出位置，更换了SanjelCommonLibrary、SanjelEService、eServiceOnline的引用问题。
2. 在bin的需求文档中加入了实体的识别文档，需要等您提交完昨天的反馈之后再继续往下进行。
3. 今天剩下的时间和Colin一起做MDD支持SQLite的任务了。

## 2019-07-10

### 工作安排

Hi Joe,

这次让你重新分析Bin的需求，我有两个目的：

1. 让你尽可能快的熟悉相关的业务背景。这个主要信息来源是已经实现的功能和向团队成员的请教。
2. 通过你学习过程，我们识别出演进式开发的过程需求，从而定义一个更适合我们情况的开发过程。

今天的提问很好，它暴露了以下的一些问题：

1. 不擅于从周边的团队成员获得支持。有很多问题随便问一下，就能得到答案。如果觉着不方便经常打扰他们工作，可以把问题积累到一起问一下。
2. 获取信息联想能力不足，Product Haul 的界面上就有Bin的选择，没有考虑Schedule Product Haul是缺失了一个主要的交互元素。 这也是目前文档可溯性的缺陷，但是这个缺陷不是很快能弥补的，如何让文档系统的设计能支持更好的可溯性，是我们在演进式开发过程中要提出来的。
3. 我早就提出来，业务流程已经识别出来了，你应该画出活动图。很多情况如果不通过流程来串起来，孤立地去思考是不可能形成一个闭环的。你考虑的两种方案，与你需求文档中的user story是冲突的。
4. 面向对象的基本原则的概念要建立起来，关注点分离，(Seperation of Concerns)。这次分析的关注点是Bin，所有其他关联操作都是以Bin为核心的。既然是以Bin为核心，就是要识别所有通过关联的 操作，来关注对Bin的状态的改变。站在Bin的角度，Blend的存量，只是一个简单进出库的问题。站在Product Haul的角度，它只是一个运送目的地的问题。二者是如何衔接的，你应该可以想明白。从Job 的执行角度，它是用了多少Blend的问题，这些Blend的是从哪些Bin中取的？Job开始前是否有足够的Blend？这些都是很好的问题。但是这些可以在过程上描述清楚，不等于我们在系统设计实现上要进一步深入，否则会无休止地追问下去，项目边界无法确定。在项目边界内无法自动化的活动，必然会有相应的手工流程来填补，这样才可以形成信息的闭环。在User Story中3.1.7， 3.1.8中，实际上我们已经描述了手工去修改Bin中 的Blend及其存量的功能。就是针对你提出的问题中目前可能存在约束信息不充沛而导致的准确性的问题。

在演进过程中，有的时候关注点的识别是会随着项目范围扩大而变化的，比如Bin。 在之前的开发过程中，我们的关注点是Rig、Product Haul，Bin是它们的属性而已，我们并不很多地关注Bin本身，现在需求扩展，我们更多关注Bin本身。不是说之前所识别的实体及关系是错的，但是随着信息的增加，和关注角度的不同，有些关系可能会进化。这一点你体会一下。

下面以红色文字提出了一些反馈。你总结今天所有的反馈，更新开发过程文档。

### 早会记录

### 下班前工作汇总

今天根据您的回复更新了Bin的需求文档，但是目前对Bin的Blend跟踪有些问题，即查询过程太复杂。

根据开会讨论结果给您发送了我们的问题详情，请您查看。

今天在开会过程中我清晰的认识到了业务流程记录的必要性，所以增加了业务流程的记录，在演进式开发过程管理系统中也进行更新。

今天Colin先完成了Assign a Bin功能的校检功能，即已经被分配给Rig的Bin不显示在Assign a Bin的页面中。

Bin Board已经添加，但是Blend和Quantity的数据还有问题，需要等待您的反馈才能继续进行。

## 2019-07-11

### 工作安排

终于等到你们自己发现问题了， 而到了这一步，刚好是业务分析到业务分析的转折点。这时要有设计的工作了。首先要通过对业务的分析和抽象，识别出实体间的关系和约束，识别出实体的状态的改变和触发改变的消息。而且为了实现这些关系、约束和状态，我们需要进行系统的设计。

Bin的Blend的跟踪过程复杂，是因为你们把各个不同方向的需求都搅在了一起，从而把层次、逻辑都变得复杂。需要把复杂的事情，分割成容易解决的简单问题进行处理。有空找一下关于“傅里叶变换”的文章读一下，学习点思考方法。

这次让你只对Bin的需求进行分析，其实已经给你定了一个边界，对你来讲，只有边界内和边界处两种情况。当你关注所有对Bin的改变都是发生在边界上的时候，你还觉得Blend的跟踪会变得复杂吗？

业务分析把这个问题搞清楚，后面就是系统分析和设计要做的事情了。至于是否增加新实体，还是继续使用并重构原来的实体，只要保证与业务分析模型的等价和对业务需求的实现，都是可选的方案。具体反馈，我在另一个邮件中以标注方式进行回答。

### 沟通记录

Hi Adam,

我们尽早对于您的反馈进行了讨论，目前对于Bin的状态分解有了一个思路，请您Review。

因为您快要到休息时间，我还没有完成全部的分析，所以请您先Review一下思路，如果没有问题，今天内我们会细化这个思路，完成关于Bin状态的需求分析。

5.1 Bin Board

在OnLine下的Resource Board中增加Bin Board，需要展示如下属性：

5.1.1 Bin Number

5.1.2 Location（状态，这部分比较复杂，解释如下：）

当Bin被分配给Rig，Location为Assign，分为两种情况：

Rig有Job时：Bin的状态为工作中，此时Bin无法被回收，自然也无法分配给其他Rig。

Rig无Job时，Bin的状态为未工作，此时Bin可以被回收，可以分配给其他Rig。

当一个RigJob的Product Haul被创建时，如果Load处于Scheduled状态，则Bin的状态为待装载Blend。

当对应的Load进行了On Location的操作，代表Blend运输完成，则Bin的状态为已装载Blend，已装载Blend分为两种情况：

由于在一个Rig Job中，一个Bin可能对应多个Load。

当一个Bin对应的多个Load都进行了On Location的操作，则Bin的状态为装载Blend完成。

当一个Bin对应的多个Load中存在Scheduled状态的Load时，Bin依然为已装载Blend状态。

注：关于装载完成和已装载的状态，主要查询当前时间下是否有对应的Load处于Scheduled状态。

是：则Bin为装载中，即Blend的装载还未完成。

否：则为装载完成，即Blend已经装载完成。

通过这种方式，我们可以在RigBoard中对Bin的状态进行展示，Dispatch便可以通过Rig Board知道RigJob当前需要的Blend是否已经到位。

由此得出，Bin的状态主要有以下几种：

Assign（Not Job/In Progress）

Is loading（正在装载）

Have the loading（已装载）

Completion of loading（装载完成）

我们是否要考虑Bin中的Blend是不是有剩余的问题？

如：RigJob没有完成，被停止了。可能Bin中的Blend并没有用完，没有用完的Blend怎么处理？

如果需要考虑Bin中Blend是否剩余的问题，那么Bin的状态还要增加。

Delivered（已经交付Blend）

Non-Delivery（未完全交付，有剩余）

如果RigJob完成，我们就默认Bin中的Blend全部用完了。

如果RigJob未完成且中断了，我们就要考虑Bin中Blend的剩余情况，这主要在于Sanjal对于剩余的Blend怎么处理。

谢谢

Joe

Adam回复

分析过度了。没有那么多状态。你所说的那些状态没有需求支持。我们并不关心是否正在装载。我们只关心这个Bin在哪儿，里面有什么，有多少。

目前只有两个状态，AssignedToRig, InYard

记住，所有分析必须要有需求支持。需要想象力，但不是天马行空的空想。

Bin是一个静态的容器，它的状态与是否有job在进行没有关系。

对于Bin中Blend的存量，只有两个动作，进和出。product haul会把blend运到Bin中，这是进。Job 会从Bin中取Blend用，这是出，出怎么做，还沒有需求，所以目前提供在BinBoard修改Bin的Blend和Quantity 的功能。

### 下班前工作汇总

Hi Adam,

1. 根据您今天的反馈，我更新了Bin的需求文档，目前主要针对的功能点如下：
   1. Assign a Bin和Schedule Product Haul时对Bin进行过滤
   2. 在Schedule Product Haul进行Save操作时，如果Load中的Blend、正在运送中的Blend、已经装载于Bin中的Blend总量超过Bin的Capatity，则给出提示且Save失败。
   3. 确认了Bin和Rig的关系，确认了Bin Board的属性和数据获取方式。
   4. 对于Bin Board的Quantity属性，经过讨论确认了计算规则。
   5. 修改Bin装载的Blend和Quantity的功能。
2. 今天也更新了演进式开发过程管理系统的需求文档，更新的较少，正在整理业务流程。
3. Colin更新了Bin需求的设计图，请您Review。SVN地址为：<https://117.34.118.114:8443/svn/SanjelDocuments/trunk/Projects/D-Design/Online/BinDesignModel.vsdx>     （bin  design model2）

谢谢

## 2019-07-12

### 工作安排

Coli, 在概念基本没有问题，表述上不太对的地方。

1. 只是BinInformation下为什么有一个BlendSection，而且还是多对一的关系。应该是一对0到1的关系，而且应当是BlendChemical
2. Bin到BinInformation 也不应当是Composite，应该反过来是个关联。因为你不能去修改Bin实体本身。

但做为设计模型就会导致很多问题，前天你们刚提出过Bin中的Blend查找很麻烦，我也有了最小化原则和最少层次原则的反馈。Colin是不是没有看？

作为设计模型，只要与概念模型等价，也许会产生额外的约束，但是总体实现代价会降低很多。

我修改了一下设计模型，你们参考一下。

另外，你在后续一些底层设计没有很清楚的情况下，根据概念模型和对交互模型的理解清楚的部分，实现了BinBoard的UI部分这点做得很好。给予表扬。这是很大的进步。这样可以更好地帮助我们得到用户的反馈，校正对需求的理解。但还有以下几处需要改进的：

1. 界面设计的细节。在实现用户界面的时候，要运用自己的经验和直觉，尽可能提交一个合理的布局。如BinBoard各列的宽度，一眼望去，你会立即发现哪些列过宽，哪些列拥挤不堪。举手之劳，会给业务分析人员在给用户展示时节省很多时间。最终用户，在反馈时往往先挑这些明显的小事儿开始，纠缠于细节，浪费宝贵的时间。
2. 在后续的实现中，因为对数据源或计算逻辑不确定，应该把它封闭起来，定义一个接口，以后只需要修改接口外面的逻辑就可以了。这样就做到了关注点分离，把复杂的事情给分解了。如BinViewModel中ProductHaulLoadModel的引入是完全没有必要的，它会带入非常重的依赖，而且在最终结果的地方，你还要加上注释当前的实现是不对的。花了大量的时间写代码，后面还要花大量时间重构。

在最初阶段最简单的办法也许是在BinViewModel中直接定义Blend和Quantity这两个属性，在BinBoardController中，调用WebConetxt的接口，最终把逻辑封装在BinBusinessProcess中。

但是，现在我们对更多的需求更清楚了，我们现在需要一个BinInformationModel来组织前台数据。根据今天的设计反馈，它是把RigBinSection实体改名为BinInformation，然后进行相应扩展。RigBinSectionModel同步改名为BinInformationModel进行相应的扩展。

Joe, 对于需要版本的控制的文档，内容只能增改，不能删除。我们通过加注释的方式对所做的修改进行解释。这样我们可以保留全部历史。这是我们所说的可溯性。你昨天把5.3删掉了，这是不对的。Update Bin Working Service Point是个真实的需求，从模型驱动分析的角度它是客观存在的，必须识别出来。而且在概念模型上它是非常重要的。不能因为它是隐含的，就把它删除了。

我把它放回来，序号改为5.7了。

我指出存在的问题，如果没修改的话，你应当自己修改。如笔误，把None打成了Note.

### 早会记录

### 下班前工作汇总

## 2019-07-13

### 工作安排

Hi Team,

今天重点还是在完善Bin的功能上，有些架构上的问题，等我去了以后再讨论重构。我周一出发之前，要发布一个测试版本。

1. 任务应当是由分析驱动的，而不是从技术上总结一下给我看，我们是在细致地观察开发过程。我期望的是你补一下团队在识别任务过程中的过程。比如说”Assign a Bin“需要重构，是哪个Test Case或是Backlog要求重构，重构中又包括哪些任务？如”重构RigBinSection, 更名为BinInformation，修改MDD中的定义“，下面会有详细地实体定义，而这个任务又可以追溯到分析中的实体定义，及用户需求等。然后是”重构eServiceOnline, 完成BinInformation的替换。Assign a bin， Remove the bin等只是受影响 的用例，它们应该可以通过我们建立的回溯体系识别出来，加入到测试中。
2. 从任务的划分上看，分析和开发上对Bin和Blend的关系的理解上还是有问题。
   1. 分析上的问题，没有把Bin本身看成是一个独立的系统，从系统的角度上看是如何与其他信息交换信息的。这个等我去了先给你们讲一下，记得提醒我。
   2. 程序设计上，一直没发生我期待的封装，这和我们之前讲的架构中各层的职责有关，不同层中相互的接口的定义和消除不必要的依赖。等我去了之后，我们再讨论这个问题，然后再进行重构。

测试中存在的问题：

1. Update Blend右击没有菜单
2. Update Quantity右击有菜单，但是Disabled.

Code Review中的疑问。

1. 在Revision 1380对1379比对中，eserviceWebContext
   1. 460-463行，为什么会加入这个逻辑？BinInformation是从RigBinSection重构来的，它包含了提供Rig和Bin关联的信息，所以在这个重构中并不会增加任何逻辑。但这里加入的创建逻辑显然是有问题的，这改变了原来系统的行为，可能会导致需要大量的回归测试。

OPEN-CLOSE原则需要再学一下。

* 1. 1000-1008行，实体关系理解错误，导致写出错误的代码逻辑。之前我曾经说过不要定义CreateOrUpdate这种接口，靠近用户界面的定义要讲人话，用接近自然语言的描述，这些接口都是分析过程中提出来的。到了后面系统接口，才会用Create, Update这种技术术语，而且定义接口语义要明确。

读一下EDP/ Asp .Net Core MVC framework based archictecture description这个文档

2. 在BinBoardProcess中，如下代码

            List<BinInformation> binInformations = eServiceOnlineGateway.Instance.GetBinInformationsByBinId(binId);

            if (binInformations.Count!=0)

           {

                BinInformation binInformation = binInformations[0];

说明对BinInformation与Bin的实体关系理解上有问题。我们说过BinInformation是Bin的扩展，是一对一的关系，这是一个强约束，不不可能出现一个BinId有多个BinInformation的记录的。如果出现了多个记录，也就是说底层出了问题。但是这种逻辑，会掩盖底层问题或是CIM定义错误。

### 早会记录

### 下班前工作汇总

Hi Adam,

今天我们完成了以下工作，请您Review。

1. 需求部分
   1. 根据您的反馈更新了Bin的需求，修改了Test Case和Backlog，增加了Ticket的部分。
   2. 对Equipment的需求进行了更新，完善了Backlog和Test Case。
2. 开发

2.1  今天我们发布了Bin的测试版本，目前分析中需要的所有功能都已经实现，等待您的反馈。

谢谢

Joe

## 2019-07-16

### 工作安排

### 早会记录

### 下班前工作汇总

Hi Adam,

今日工作情况如下：

1. 需求

根据您的反馈更新了Bin的需求文档，此部分只有微小的更新。

更新了Equipment的需求文档，主要完善了Backlog和Test Case。

2.开发

Colin今天对Equipment的需求画了设计图。

根据您的反馈对设计图进行了修改，并完成了MDM的创建。

明日进入功能开发。

谢谢

Joe

## 2019-07-17

### 工作安排

### 早会记录

### 下班前工作汇总

Hi Adam,

今日工作情况如下：

1. 完成了：
   1. Display Plug Loading Head Board
   2. Display manifold Board
   3. Display Top Drive Adaptor Board
   4. Display Plug Loading Head Sub Board
   5. 由于Notes结构的修改，今日也完成了4个Board中Update Notes的功能。
2. 未完成的工作
   1. Assign Plug Loading Head（这个部分是本次需求的重点，逻辑较多，是明日主要工作目标）
   2. Return Equipments

谢谢

Joe

## 2019-07-18

### 工作安排

### 早会记录

### 下班前工作汇总

Hi Adam,

今日工作情况如下：

1. 早上对本次工作进行了总结，讨论了关于演进式开发过程管理系统。
2. Colin今日完成了Assign Plug Loading Head的前台工作。

谢谢

Joe

## 2019-07-19

### 工作安排

### 早会记录

### 下班前工作汇总

Hi Adam,

今日工作情况如下：

1. 早上对本次工作进行了总结，讨论了关于演进式开发过程管理系统。
2. Colin今日完成了Assign Plug Loading Head的前台工作。

谢谢

Joe

## 2019-07-19

### 工作安排

1. Adam讲解关于模型的定义和开发过程。
2. Adam讲解Sanjel的完整业务流程。

### 早会记录

### 下班前工作汇总

今日工作主要为参加会议，具体如下：

1. 熟悉新的开发流程。
2. 熟悉Sanjel的业务流程和场景。
3. 熟悉eService Express的需求。
4. 进行演进式开发过程管理系统的前期分析。

## 2019-07-20

### 工作安排

1. Adam讲解认知科学与业务分析的应用场景。
2. 完成无法将评论/泵计划添加到为客户打印的图表中。的需求分析并实现。
3. 完善Sanjel team的开发过程，制定规则。

### 早会记录

### 下班前工作汇总

今日工作情况：

1. 完成Comments的需求分析。
2. 完成在CurrentJob页面中增加PostJob列，PostJob页面可添加Job的Comments功能。
3. 完成在图表上打印Comments的功能。
4. 确认Sanjel team的开发过程，规则定制完成。

## 2019-07-21

### 工作安排

1. Joe：完成eService Express的所有需求分析。
2. Bright：修改JobMonitor中实时监测表格的算法，使时间节点为整数。
3. Bright：隐藏JobMontior中的“时间段”和“间隔时间”设置
4. Colin：在打印时增加时间段配置弹窗，配置需要打印的时间段，递增时间后生成对应的PDF表供打印。

### 早会记录

### 下班前工作汇总

1. eService Express的所有需求分析完成。
2. 修改JobMonitor中实时监测表格的算法，使时间节点为整数。（完成）
3. Bright：隐藏JobMontior中的“时间段”和“间隔时间”设置（完成）
4. Colin：在打印时增加时间段配置弹窗，配置需要打印的时间段，递增时间后生成对应的PDF表供打印。（完成）

## 2019-07-22

### 工作安排

1. Joe：根据反馈继续完善eService Express的需求，整理Burndown，绘制Ui原型。
2. Bright：修复在eSe中确认主管选择后，你还必须确认WITS设置的BUG、使Job Monitor中的所有趋势图垂直对齐Y轴和辅助Y轴
3. Colin：增加在程序启动时清空JobChartPrintOut文件夹temp folder、将Density的趋势图放到到Job Monitor中其他趋势图的上方，打印时同理。

### 早会记录

### 下班前工作汇总

1. eService Express需求完成，Burndown完成，Ui原型绘制完成，等待反馈。
2. 增加在程序启动时清空JobChartPrintOut文件夹temp folder（完成）
3. 将Density的趋势图放到到Job Monitor中其他趋势图的上方，打印时同理。（完成）
4. 修复在eSe中确认主管选择后，你还必须确认WITS设置的BUG（完成）
5. 使Job Monitor中的所有趋势图垂直对齐Y轴和辅助Y轴（未完成，无法控制插件，调整前台样式的方式不合理，不采用。）

## 2019-07-23

### 工作安排

1. Joe：完善Burndown、熟悉eService需求，根据反馈修改UI原型。
2. Bright：使PDF中每个趋势图的标题只显示项目/计算单位，例如速率(m3/min)、生成的PDF表单中的kg/m3变更为kg/m³、在打印PDF表单时根据卡车的数量控制泵送表单的Y轴最大值
3. Colin：在PDF上，将温度/密度放在同一个趋势图内、打印时生成图表的颜色为标准的蓝/红/绿

### 早会记录

### 下班前工作汇总

1. 完善Burndown、熟悉eService需求，根据反馈修改UI原型。（完成）
2. 使PDF中每个趋势图的标题只显示项目/计算单位，例如速率(m3/min)（完成）
3. 生成的PDF表单中的kg/m3变更为kg/m³（完成）
4. 在打印PDF表单时根据卡车的数量控制泵送表单的Y轴最大值（完成）
5. 在PDF上，将温度/密度放在同一个趋势图内（完成）
6. 打印时生成图表的颜色为标准的蓝/红/绿（完成）

## 2019-07-24

### 工作安排

1. Joe：分析eService需求。
2. Bright：在Job Monitor中实时显示WITS的状态、修改Job Monitor页面中的立方米单位m3为m³、配合部署eService
3. Colin：修改Job Monitor页面Ui，增加WITS status，和WITS趋势浮窗。

### 早会记录

### 下班前工作汇总

1. 分析eService需求。（未完成，需要部署测试版eService，否则看不懂需求）
2. 在Job Monitor中实时显示WITS的状态（完成）
3. 修改Job Monitor页面中的立方米单位m3为m³（未完成，趋势图插件中不支持，还需讨论）
4. 配合部署eService（完成）
5. 修改Job Monitor页面Ui，增加WITS status，和WITS趋势浮窗。（未完成，只完成了初步的数据展示和实时刷新，样式还没调整）

## 2019-07-25

### 工作安排

1. Joe：熟悉eService系统，分析eService需求。
2. Bright：协助雷宇一起搭建CruiseControl的环境、自动化集成代码。
3. Colin：根据反馈继续调整Job Monitor页面Ui。

### 早会记录

### 下班前工作汇总

## 2019-07-14

### 工作安排

### 早会记录

### 下班前工作汇总

## 2019-07-14

### 工作安排

### 早会记录

### 下班前工作汇总

## 2019-07-14

### 工作安排

### 早会记录

### 下班前工作汇总

## 2019-07-14

### 工作安排

### 早会记录

### 下班前工作汇总