

Driver Compensation

Architecture

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 08/21/08 | 1.0 | Created | Jim Heary |
| 05/19/09 | 2.0 | Updated for new requirements | Jim Heary |
| 07/10/13 | 3.0 | Updated for SOA design changes | Jim Heary |
| 07/31/14 | 4.0 | Updated for Bonus requirement | Jim Heary |
|  |  |  |  |

Table of Contents

Introduction 4

Overview 4

Scope 5

Risks 5

Business Activities 6

Manage Compensation 6

Create Compensation 7

System Analysis 8

Overview 8

Actors 8

Use Cases 8

View Routes 9

View Compensation 9

Create Compensation 9

Rate Routes 10

Apply FSC 10

Apply Fees 10

Apply Bonus 10

Edit Compensation 10

Export Compensation 10

Administer Rates 10

Business Rules 10

Non-functional Requirements 10

Presentation 10

Reports 10

Performance 11

Security 11

Availability 11

Concurrency 11

Interoperability 11

Key Abstractions 11

States 12

System Design 14

Overview 14

Design Model 14

Structure 14

Behavior 15

User Interfaces 16

Persistence 16

Availability 17

Concurrency 17

Security 17

Persistence 18

Deployment 18

Project Planning 20

Iteration 1 20

Appendix 20

Export File 20

Reports 22

Cost/Carton Summary 22

Driver Compensation 22

Fleet Owner Compensation 23

# Introduction

## Overview

Finance needs to compensate Argix contract drivers (i.e. non-salaried employees) for pickups and deliveries at each of the Argix Logistics local terminals. This activity is identified as Driver Compensation. The current process uses a custom Excel spreadsheet to apply rates to Roadshow route data, calculate driver compensations, and provide summary information.

The custom Excel spreadsheet requires data entry of Roadshow route data, and encapsulates the business logic and rate data required to calculate driver compensation for the week for each driver. It also includes worksheets that summarize cost per carton and driver compensation. Route rating is determined by vehicle type (i.e. straight truck) for either or both mileage and units (i.e. trips made, cartons delivered, etc.); in addition, it can also be determined by route, regardless of vehicle type, for either or both mileage and units.

The problem is that the current solution is file based (i.e. Excel spreadsheet); thus the following limitations apply:

* compensation information is localized and not easily accessible (i.e. file sharing only)
* compensation information is chunked by week and thus not easily collated for broader reporting
* Roadshow route data must be manually entered
* new rating schemes would be difficult to implement

A new solution is required to:

* eliminate data entry (i.e. entering Roadshow routes into the spreadsheet)
* provide universal access to compensation information
* provide extensibility in rating schemes
* provide enhanced reporting

This document details known requirements and a candidate design. It does not address manpower requirements or project scheduling, although these can be determined from this architecture.

*How to read this document:*

Management- Introduction, Analysis (Overview, Business Activities, Actors), Project Planning

Operations- Introduction, Analysis, Project Planning

IT- Introduction, Analysis, Design, Project Planning

Context

The following context diagram shows the roles and systems involved in the driver compensation system. This helps us to define the system boundary, that is, what is part of Driver Compensation and what is not. Notice, only the Billing Supervisor and Billing Clerk directly access the Driver Compensation system. In addition, the Driver Compensation system automates reading driver routes from the Roadshow system (currently, the Billing Clerk reads the routes from Roadshow). Drivers have no association to the Driver Compensation system; this could change, for instance Driver Compensation could send an email when the compensation is complete, or the Driver could access a web site and review information about his routes, compensation, or current rates.



## Scope

Solution Boundary

The solution will provide interaction with Roadshow for route import; allow rates administration by the Billing Supervisor; allow compensation calculations by the Billing Clerk; and allow the Billing Clerk to export the compensation to a fixed-field text file. Paying drivers is outside the scope of the solution.

Constraints

1. The solution shall conform to the principles of a Service Oriented Architectures (SOA).
2. The solution will use the current Microsoft-based platform and Microsoft-based development tools.

## Risks

1. There are no risks to address at the current time.

## Business Activities

The following diagrams show the business activities for the Driver Compensation system. Business activities are modeled using activity diagrams. Activity diagrams show a series of activities, the actors responsible for each activity, and the information consumed and created during the activities. These diagrams use the language of the business. Activity diagrams drive the discovery of Actors, Use Cases, and Key Abstractions.

Manage Compensation

The following diagram shows a high level view of the business activities centered on the activity of creating driver compensation. The Dispatch Clerk in the local terminal imports delivery orders from Bearware into Roadshow using an import file. He also imports pickup orders from Argix Dispatch into Roadshow using an import file. Driver routes are created by the Roadshow system (i.e. route solution) and assigned to drivers. On Tuesday of each week, the Billing Clerk imports the Roadshow routes from the prior week and creates driver compensation; this information is recorded as Driver Routes in the Driver Compensation system. The activity Create Compensation is broken down into sub-activities in the following activity diagram Create Compensation.



Create Compensation

The following diagram shows the sub-activities of the Create Compensation activity from the prior activity diagram Manage Compensation. This activity is automated by the Driver Compensation system. The Billing Clerk selects one or more imported driver routes and chooses to create compensation. The system responds with a series of activities that rate routes and determine compensation on a per driver basis. This includes rating route miles, units (i.e. cartons, pallets, etc.), applying FSC, administrative fees, and bonus. Once completed, driver routes are in a Rated state.



# System Analysis

## Overview

The purpose of defining system behavior is to discover, capture, and analyze the requirements of the system under discussion. This is achieved by describing the requirements (i.e. the conditions or capabilities to which the system must conform) well enough so that an agreement can be reached between the business users and the system developers on what the system should and should not do. It begins by modeling the business processes with a series of activity diagrams. These diagrams drive discovery of the users of the system (i.e. Actors), the system functionality (i.e. Use Cases), and the vocabulary of the system (i.e. Key Abstractions). From these artifacts, an analysis model is created that drives system design and development.

Driver Compensation is initiated on Tuesday of each week by the Billing Clerk; compensation is determined for each diver in each local terminal for the routes they serviced. Routes are recorded in the Roadshow system; routes are constructed from delivery orders imported from Bearware and pickup orders imported from Argix Dispatch. So, a route, or trip, includes information about the driver’s vehicle, how many miles driven, trips made, stops made, and cartons and pallets delivered and picked up. The compensation is determined from this information and is based upon the driver’s vehicle type or a specific named route; in either case, rates are applied to mileage and or units (i.e. trips made, cartons picked up) to determine compensation.

## Actors

An actor specifies a role played by a user or any other system that interacts with the system under discussion. Actors influence UI design and security concerns. The list below provides definitions for the actors surrounding the Driver Compensation system.

* Driver Compensation- the system to be built.
* Roadshow- third party software system that creates driver routes from pickup and delivery orders and is the source of routes for Driver Compensation.
* Bearware- third party software system that is a source of delivery orders for Roadshow.
* Dispatch- Argix business application that is a source of pickup orders for Roadshow.
* Dispatch Clerk- an Argix employee who uses Roadshow to create driver routes.
* Billing Clerk- an Argix employee who creates driver compensation based upon driver activity as captured by Roadshow.
* Billing Supervisor- an Argix employee who administers the rates used by driver compensation. Additionally, the Billing Supervisor is a Billing Clerk, meaning she can do anything a Billing Clerk can do.
* Driver- an owner/operator who executes pickup and delivery orders based upon assigned Roadshow routes; also, the payee of driver compensation.

## Use Cases

A Use Case is a list of steps, typically defining interactions between a role (i.e. Actor) and a system to achieve a goal. The actor can be a human or an external system. Use Cases describe the functional view of the system under discussion as a set of business transactions. Use Cases influence UI design, domain models, application service interfaces, and define business transactions. The following Use Case diagram shows some, if not all, of the actors and use cases involved in the system.



Import Routes

The Billing Clerk needs to import Roadshow routes for all drivers in a single local terminal for a selected week (Sunday thru Saturday). The system returns a view of imported routes that includes operator name, route date, route name, equipment type, miles, trips, stops, cartons, return cartons, and pallets.

View Compensation

The Billing Clerk needs to view the compensation for one or more drivers in a single local terminal for a selected week (Sunday thru Saturday). This view includes a summary view for each driver; and a detail view for each route for each driver. The detail view includes route date, route name, equipment type, miles/miles amount, daily amount, trips/trips amount, stops/stops amount, cartons/cartons amount, pickup cartons/pickup carton amount, pallets/pallets amount, minimum amount, route total amount, FSC miles/FSC MPG/FSC amount, administrative fees, adjustment charges, rating type (i.e. vehicle, route), and [Roadshow] route import date. The summary view includes operator name, finance vendor number, sums of the individual rated routes for all fields specified in the detail view, and the total compensation amount.

Create Compensation

The Billing Clerk needs to create driver compensation for one or more imported routes.

***Pre-Conditions:***

Routes have been imported from Roadshow.

**Flow of Events**:

Basic Path

1. The user selects one or more imported routes.
2. The user chooses to add the routes to new or existing driver compensation.
3. The system rates each selected route: see use case Rate Routes.
4. The system applies FSC: see use case Apply FSC.
5. The system applies administrative fees: see use case Apply Fees.
6. The system applies bonus: see use case Apply Bonus.
7. The system calculates driver compensation for each driver.
8. The system saves the rated routes and updates the compensation view.

Alternative Paths

None.

Post-Conditions:

The selected routes are saved to a Rated state.

Rate Routes

The System needs to rate each route concerning mileage and units.

Apply FSC

The System needs to apply FSC to each route.

Apply Fees

The System needs to apply administrative fees to each drivers’ compensation.

Apply Bonus

The System needs to apply a bonus to each drivers’ compensation.

Edit Compensation

The Billing Clerk needs to edit existing driver compensation. Rated routes can be deleted from the driver compensation as required. Rated routes can be edited for AdjustmentAmount1, AdjustmentAmount1TypeID, AdjustmentAmount2, AdjustmentAmount2TypeID, and AdminCharge. The system saves the changes and updates the compensation view.

Export Compensation

The Billing Clerk needs to export driver compensation to a fixed-field text file. The Billing Clerk can choose a view the export information. The Billing Clerk can export driver compensation to a new file on their computer. The export file format is listed in Appendix 1.

Administer Rates

The Billing Supervisor needs to administer rates used in rating driver routes.

## Business Rules

1. The compensation week is Sunday thru Saturday.
2. Rates are date-based and cannot be changed- they are overridden by newer rates.
3. Rates are applied to mileage and/or units (i.e. trips, carton delivered, etc.) and are based upon vehicle type by default; however, if rates exist for a named route, then the route rates override the vehicle rates.
4. Fuel surcharges (FSC) apply only if a rate exists for mileage (vehicle or route).
5. Administration charges apply to the compensation for each driver; use the last charge levied against the operator if available, or use a configurable value specified for each terminal (usually -$10).

## Non-functional Requirements

Presentation

Desktop access only. The application should use a spreadsheet paradigm.

Reports

Cost/Carton Summary- a summary for each local terminal for a specified week with a daily detail showing number of routes, total trips, total cartons, cartons/route, total FSC, total cost, and cost/carton; also, totals of the detail for the entire week.

Driver Compensation- a breakdown by terminal, driver, and date showing amounts and cost for miles driven, stops, cartons delivered, pallets delivered, return cartons, compensation adjustments, and FSC; also, daily and weekly cost total.

Fleet Owner Compensation- a breakdown by terminal and fleet owner (payee) showing compensation for the specified week paid to each operator; also, total compensation for the week to the fleet owner.

Performance

Nothing specified.

Security

Authentication: only domain users can access the system; there are no anonymous users.

Authorization: limited system access depending upon role (i.e. Billing Clerk).

Availability

Normal business hours.

Concurrency

This is a single user system.

Interoperability

Roadshow- import driver routes from Roadshow.

## Key Abstractions

The class diagram below shows the key abstractions involved in the system. Key abstractions are the key concepts and abstractions that the system needs to handle. They are those things that, without which, you could not describe the system. Key abstractions drive design of the database schema and the domain model (if applicable).



* Local Terminal- is an Argix-owned agent terminal for delivery and pickup of freight to stores within its designated zone.
* Roadshow Route- summarizes one or more trips made on a single day (i.e. route date) by a driver using a type of vehicle on behalf of the local terminal. It specifies a route name and trip details including miles driven, trips made, cartons delivered, etc.
* Driver Compensation- a summary compensation of all Route Compensations for a single Driver over a specified duration (i.e. one week). It shows summary values and a total compensation amount.
* Driver Route- is a rated Route that captures compensation detail for the miles driven, trips made, cartons delivered, etc.
* Rates- define the compensation due for miles driven, trips made, cartons delivered, etc.
* Vehicle Rates- defines Rates that are based upon the type of equipment (i.e. vehicle type) used for a route.
* Route Rates- defines Rates that are based upon a named route.

## States

The state machine view describes the dynamic behavior of objects over time by modeling the lifecycles of objects of each class. Each object is treated as an isolated entity that communicates with the rest of the world by detecting events and responding to them. Events represent the kinds of changes that an object can detect. Anything that can affect an object can be characterized as an event.

Driver Route

Driver Route is in an imported state...



# System Design

## Overview

The system design includes a rich Windows client that provides management of Driver Compensation. It accesses back-end databases through web services arranged in a Service Oriented Architecture (SOA). Security is provided by an ASP.Net membership database hosting domain accounts. Program services are secured using pre-defined user roles.

## Design Model

Structure

The design is an n-tier physical and logical design composed of a .NET 4.0 Win Forms client (DeliveryPoints.exe), several .NET 4.0 WCF Services (Argix10.Terminal.Services), and a SQL Server 2008 R2 database. The design as a whole is built to conform to a Service Oriented Architecture. Service Oriented Architectures describe ways to build loosely-coupled systems composed from individual services. All software components are built with Microsoft Visual Studio 2010 and compiled to operate in.Net 4.0 framework environments. The structures of the design are defined by drawings of the components and interfaces. The behaviors of the design are defined by sequence diagrams involving the interactions between the components.

Enterprise Patterns

In [software engineering](http://en.wikipedia.org/wiki/Software_engineering), a [design pattern](http://en.wikipedia.org/wiki/Design_pattern) is a general reusable solution to a commonly occurring problem within a given context in [software design](http://en.wikipedia.org/wiki/Software_design). It is a description or template for how to solve a problem that can be used in many different situations. Patterns are formalized [best practices](http://en.wikipedia.org/wiki/Best_practice) that the programmer must implement in the application. [Object-oriented](http://en.wikipedia.org/wiki/Object-oriented) design patterns typically show relationships and [interactions](http://en.wikipedia.org/wiki/Interaction) between [classes](http://en.wikipedia.org/wiki/Class_(computer_science)) or [objects](http://en.wikipedia.org/wiki/Object_(computer_science)), without specifying the final application classes or objects that are involved. The following enterprise design patterns are used in throughout the design:

Transaction Script- a Transaction Script organizes business logic by procedures where each procedure handles a single request from the presentation. They are an excellent choice when we don’t have a middle tier domain model. Transaction scripts are used in the web services to provide business transactions and security for one or more use cases.

Gateway- a Gateway is an object that encapsulates access to an external resource such as a web service or database server. Gateways are used throughout the design for user interface access to middle tier web services, for web service access to backend database services, and for web service access to external hardware including a label printer.

Record Set- a Record Set is an in-memory representation of tabular data; record sets (i.e. Dataset) work well with .NET user interface components.

Data Transfer Object- a Data Transfer Object (DTO) is an object that carries data between processes in order to reduce the number of method calls. Data Transfer Objects are serialized classes exposed by the interfaces of the web services for transactional calls to the middle tier (i.e. Service::Method(DTO dto)).

System Structure

The following drawing shows the components and interfaces (structure) of the Driver Compensation design, and the connections between the various components. Components are denoted with tabbed containers. The rectangles within the components are software units stereotyped with their functional purpose such as a gateway or transaction script.



Behavior

System Behavior

The following drawing shows the behavior of …



User Interfaces

Windows Client

The Driver Compensation Windows application is built using .NET Win Forms for the .NET 4.0 Framework. It communicates with a middle-tier WCF service using the WcfHttpBinding binding. It is secured using an ASP.NET Membership services database through a WCF Role Service.

The application is built using Visual Studio 2010 and is deployed onto a LAN-based IIS 7.5 web server where it is installed onto client computers using the Click-Once technology.

Persistence

Database Schema

The database schema is initially derived from the Key Abstractions. The drawing below

The database is built and deployed with a SQL Server 2008 R2 database server. The schema is exposed through stored procedures.



## Availability

The application will be available during normal business hours. It will be accessible from the SharePoint Portal.

## Concurrency

The system supports single user operation.

## Security

Authentication

The network is responsible for user authorization. This would occur when the user logs into the domain with their domain credentials. Anonymous users are not supported.

Authorization

The application uses ASP.NET role-based security to authorize access to application features and services. The RoleServiceGateway encapsulates the details to access the RoleService from the MembershipServices web service. The RoleServiceGateway also acts as a policy file where role-based access is queried by user interface objects. For instance, winDriverComp checks with the RoleServiceGateway to determine if the user has the rights to calculate driver compensation. The following drawing shows the processes and components that realize the application authorization design.



## Persistence

## Deployment

The current deployment consists of a centralized application on the Jamesburg local area network (LAN) with client computers running on the Jamesburg LAN as well. Access to the application will be via the Argix Corporate Portal (i.e. SharePoint) using a Click-Once deployment that assumes the deployment server is always available; although the deployment is cached on the client, the server is always checked for a version update. Additional information concerning the click-once deployment can be found in the release notes accessible from the Help menu.

The deployment platform is as follows:

* Client- Windows7 operating system with .NET Framework 4.0 (full, not client)
* Web/Application Server- Windows Server 2008 R2 and IIS 7.0
* Database Server- Windows Server 2008 R2 and SQL Server 2008 R2
* Reporting Server- Windows Server 2008 R2 and SQL Server Reporting Services 2008 R2



# Project Planning

## Iteration 1

Implement the following Use Cases:

Use Case 1

Use Case 2

Address the following risks:

Risk 1

# Appendix

## Export File

For each operator in that has compensation, a driver record is added to the export. If administrative fees exist (i.e. > 0), then an admin record is added as well.

Driver Record

All fields fixed 12 position and left justified

022 13418 09/13/08 WE091308 Mi 484 012 08 1424.23 18505000 N

------------============------------============------------============------------============------------============------------============------------

| | | | | | | | | | | | |

a b c d e f g h i j k l m

a) Company Code [12]; always 022

b) VendorFinanceID [12];

c) Date [12, MM/dd/yyyy];

d) Invoice# [12, WE + Date(MMddyy)];

e) Desc1 [12]; always blank

f) Desc2 [12]; always blank

g) Desc3 [12]; always blank

h) Desc4 [12]; always blank

i) Pay period month [12, MM];

j) Pay period year [12, yy];

k) Total amount [12];

l) General LG# [12];

m) TaxID [12]; always N

Admin Record

All fields fixed 12 position and left justified

022 13418 09/13/08 AD091308 Ad 012 08 -10.00 18516000 N

------------============------------============------------============------------============------------============------------============------------

| | | | | | | | | | | | |

a b c d e f g h i j k l m

a) Company Code [12]; always 022

b) VendorFinanceID [12];

c) Date [12, MM/dd/yyyy];

d) Invoice# [12, AD + Date(MMddyy)];

e) Desc1 [12]; always blank

f) Desc2 [12]; always blank

g) Desc3 [12]; always blank

h) Desc4 [12]; always blank

i) Pay period month [12, mm];

j) Pay period year [12, yy];

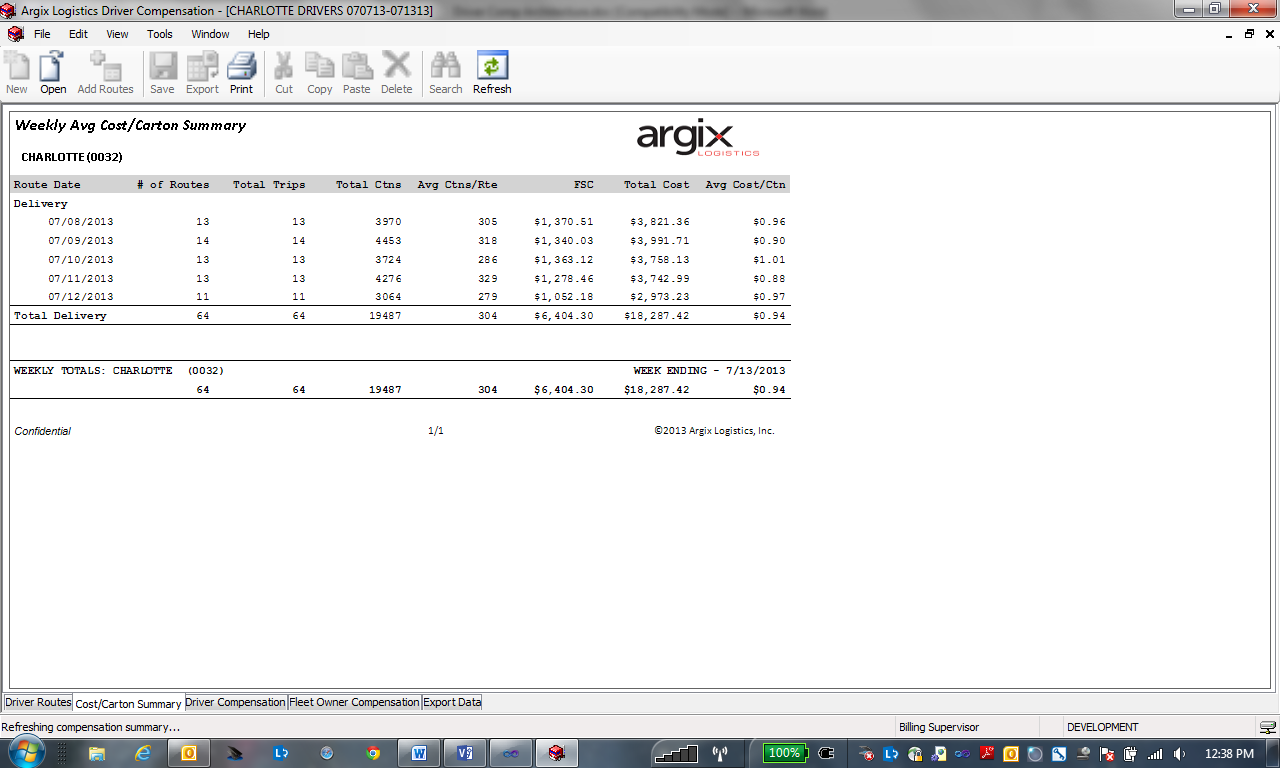
k) Admin amount [12];

l) General LG# [12];

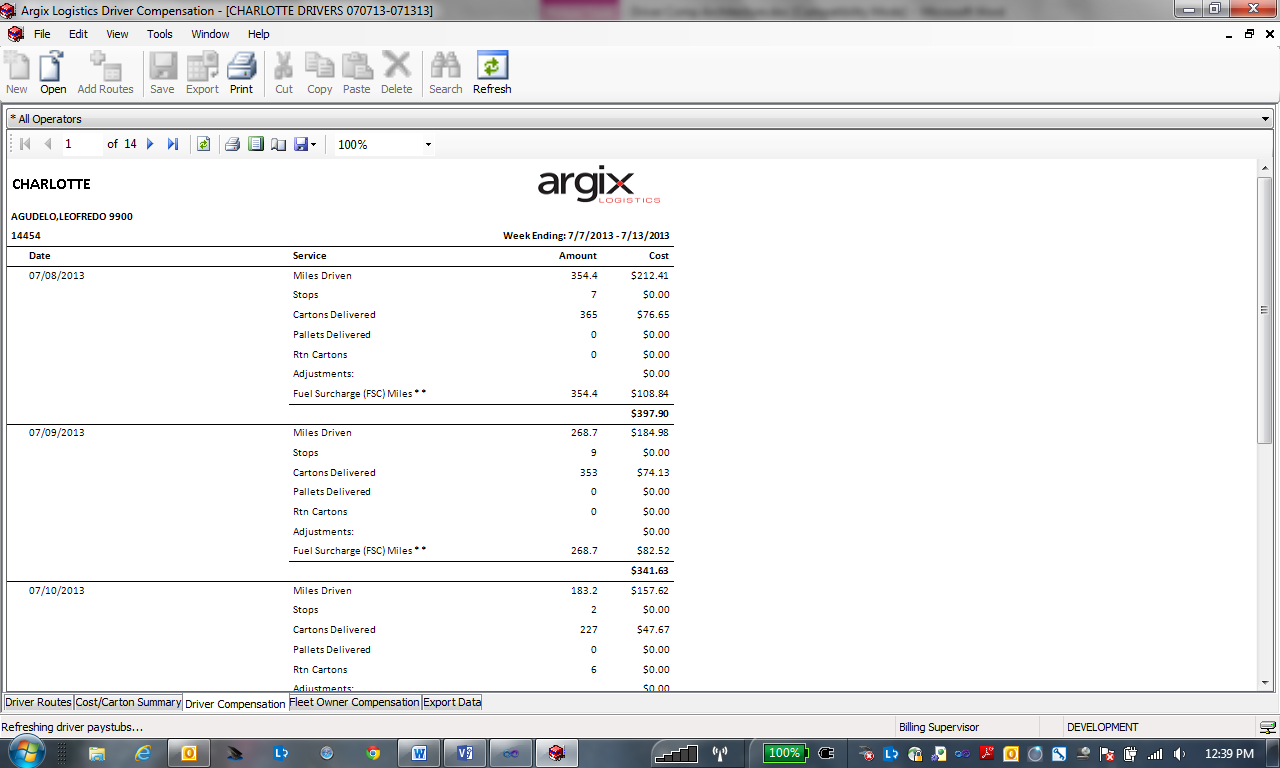
m) TaxID [12]; always N

## Reports

Cost/Carton Summary



Driver Compensation



Fleet Owner Compensation

