Celluride Wireless

Technical & Product Documents

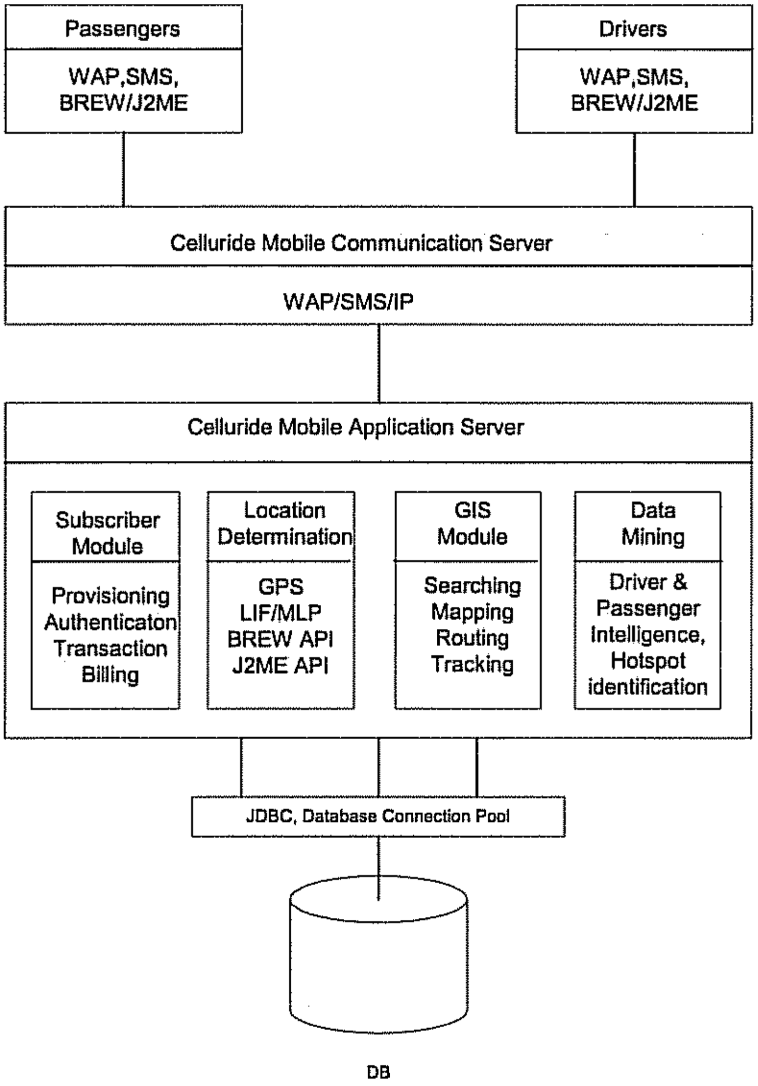


Figure 1. Celluride High-level Architecture

Subscriber Module:

Responsible for all subscriber related operation, including:

(1) Provisioning: all subscribers need to be registered in the system to use the Celluride's service.
(2) Authentication: authenticate an incoming subscriber's identity
(3) Transaction: each time a Celluride transaction occurs, there will be multiple transaction states involved. All transaction details will be recorded thru this module.
(4) Billing: handles all billing related issues. Billing can be per subscriber, per transaction, or tightly integrated with the carriers or service providers.

Location Determination Module:

Responsible for determining the x, y, z of the subscriber based on a variety of methods:

(1) GPS: applied to a GPS-enabled device. Depending on the device's GPS output interface, location information can be fetched via standard API or reading raw GPS sentences from the communication port.
(2) LIF/MLP: using the mobile location protocol defined by the location interoperability forum, the subscriber's location can be queried from a service provider. This is a network-centric solution.
(3) BREW: Qualcomm's IPosDet API supports the fetching of location in a BREW-enabled handset.
(4) J2ME: based in JSR 179 specification, the device's location can be fetched using industry-standard Java Micro Edition.

GIS Module

Responsible for all spatial data oriented operations, including:

(1) Searching: supports the "Find Nearest" spatial searching. Identifies all drivers that are within the passenger's search radius.
(2) Mapping: provides maps to drivers, passengers, and backend system operators.
(3) Routing: finds the shortest path or driving direction for passengers or drivers
(4) Tracking: the movement of passengers and drivers can be continuously tracked to ensure they are moving at the right direction or if they suddenly change course.

Data Mining Module:

An optional add-on to the base Celluride system, aiming to provide intelligence information and data mining capabilities based on the subscriber's database Celluride possess, and the patterns and statistics of subscriber's past transaction behavior and credibility.

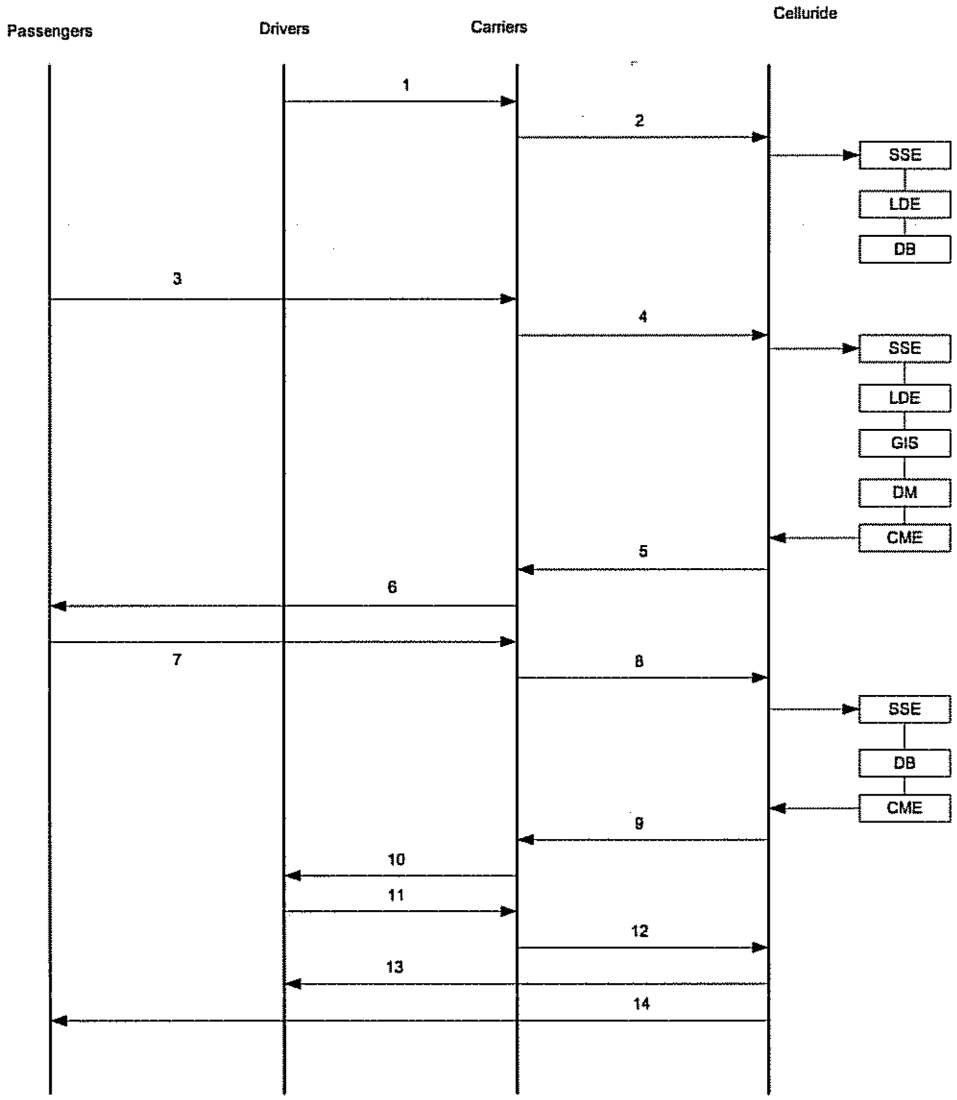


Figure 2. Celluride End-to-End Message Flow

1. A driver who needs to find new passengers pushes a button on his cell phone. A message indicating the driver's availability and location information are sent to the driver's carrier proxy server or gateway.
2. Carrier proxy determines that this packet belongs to Celluride and forwards it to Celluride's server. This packet will first be processed by Celluride's Subscriber & Service Engine (SSE), which authenticates the driver. Then Celluride's Location Determination Engine is called to retrieve the driver's location. The location determination method can be handset based or network based, depending on the cell phones location technology. Finally, the driver's latest location and availability information are saved in the database thru the DB module.
3. A passenger looking for a ride pushes a button on his mobile handset hosting Celluride's application front-end GUI, indicating the passenger's need for a ride. This information with the passenger's location is sent to passenger's carrier proxy server.
4. The carrier proxy determines that this message belongs to Celluride and forwards it to Celluride's server. The server determines this message is from a passenger. It then calls the SSE to authenticate the passenger, retrieves the passenger's location, and then calls the GIS module to find out all the drivers that are close to this passenger's location. The selected drivers are further filtered by the Data Mining (DM) module. The final list of drivers are those that meet the user's preset criteria for vehicle options.
5. The Celluride's Communication Engine (CME) is invoked and the selected lists of drivers are pushed back to the passenger, thru the carrier proxy.
6. The carrier forwards the above message to the passenger.
7. The passenger looks at this list and picks the final choice of the driver. A message is generated.
8. The carrier proxy determines this message belongs to Celluride and forwards it. The Celluride server invokes SSE to log user's transaction information and saves any necessary billing information in the DB
9. Celluride invokes the CME to send a notification message to the selected driver.
10. The carrier pushes this message to the driver's handset.
11. The driver either accepts or denies the request. An indication message was generated.
12. Carrier forwards driver's decision over to Celluride.
13. Celluride logs the transaction information for billing purpose and sends the routing information and maps showing the driver how to get to the passenger's location.
14. Celluride sends a confirmation message to the passenger.

Celluride Service Platform

Technical Architecture Document

Product	**Celluride Service Platform**
Version	**0.3**
Created	**September 27, 2005**
Updated	**November 27, 2005**

Revision History

Date	Author	Rev	Comments
09/27/05	Taspa Alagarsamy	0.1	Created
10/14/05	Taspa Alagarsamy	0.2	Updated with feedback from John, Harry & Kevin
10/25/05	Harry Niedzwiadek	0.3	Updated from CCC Architecture to Celluride Service Platform Architecture and added GIS Module details
11/27/05	Taspa Alagarsamy	0.4	Updated deployment section for details

Table of Contents

1. Introduction

Objective

The objective of this document is to detail the necessary information concerning the technical architecture for the Celluride Service Platform, serving all actors in the Celluride value chain. This is a living document and will evolve as the architecture is refined and extended to fulfill the requirements and changes in the business. In order to convey the full scope of the target platform, some sections may contain information that is beyond the scope of the initial release. The Celluride Project Plan contains detailed information concerning the planned implementation of the platform.

This document:

- Defines the high-level system architecture for a unified service platform to manage the Celluride business flow.
- Details the major functions/services for the system.
- Outlines the major information types for the system.
- Identifies the integration framework between major modules and components and between the platform and external systems.
- Addresses usability, scalability, performance, security and deployment considerations

Audience

This document is primarily for the Architects and Developers who are familiar with software engineering. You will also find some implementation specific details and choices that are available. The reader of this document is assumed to be knowledgeable on the UML notations (also part of Rational Unified Process). Familiarity with enterprise software will be helpful.

Conventions & Terms

Block diagrams are used to describe the dynamic components of the system at a higher level. UML notations are used with stereotypes when needed to detail the static and dynamic structures of the system. Following are the terms and abbreviations used and shall be used in communications in order to minimize any misinterpretations.

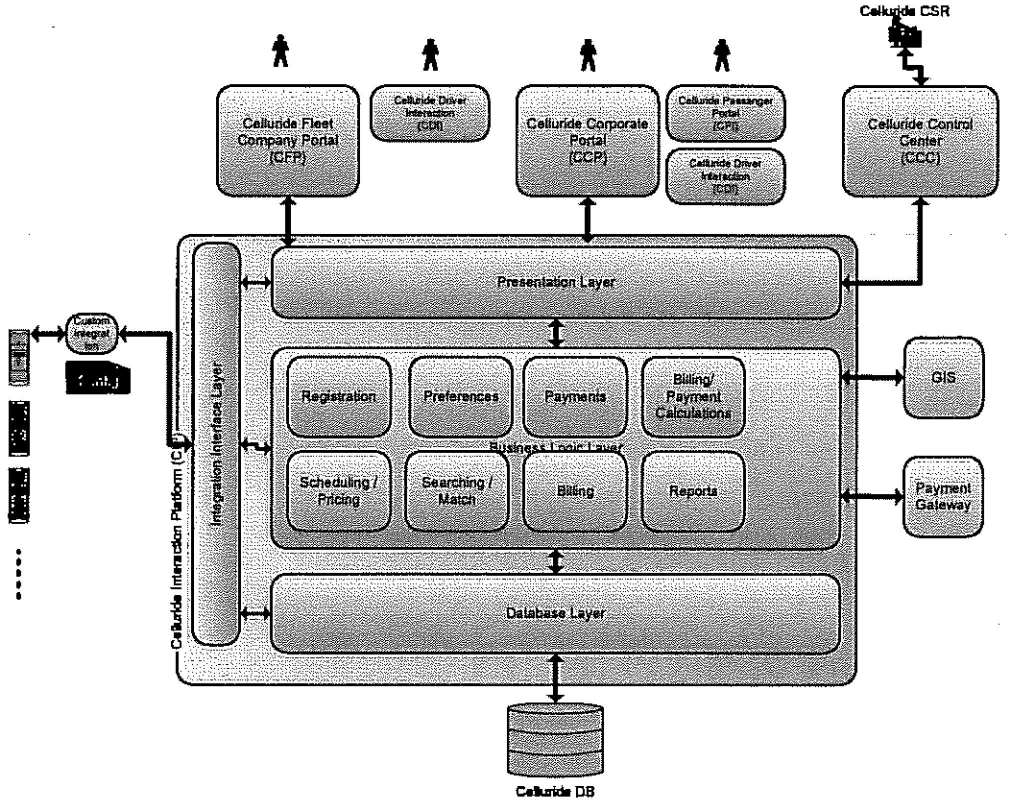
Terms used:

- Celluride Control Center (CCC)
- Celluride Customer Portal (CCP)
- Celluride Fleet Company Portal (CFP)
- Celluride Driver Interaction/Interface (CDI)
- Celluride Passenger Portal (CPP)
- Control Center User (Celluride Employee)
- Customers
- Drivers
- Fleet Company
- Geographic Information System (GIS)
- Location-Based Services (LBS)
- Passengers
- Vehicle

2. Architecture Overview

High Level Abstraction

Following is a high level block diagram that describes the functional elements of the Celluride platform as whole in an abstract manner. Each functional element is described in details categorized as modules in the following sections. These shall be further decomposed into the implementation level details.


Celluride CSR
Celluride Fleet Company Portal (CFP)
Celluride Driver Interaction (CDI)
Celluride Corporate Portal (CCP)
Celluride Passanger Portal (CPI)
Celluride Driver Interaction (CDI)
Celluride Control Center (CCC)
Presentation Layer
Custom Integrat ion
Celluride Interaction Platform (CIP)
Integration Interface Layer
Registration
Preferences
Payments
Billing/ Payment Calculations
Business Logic Layer
Scheduling / Pricing
Searching / Match
Billing
Reports
GIS
Payment Gateway
Database Layer
Celluride DB

Celluride - Highlevel Block Diagram (Level I)

Here are some points describing the above block diagram:

- Different interfaces will be available for each kinds of users.
- Customers and passengers will be interacting through a portal (CCP) to request for rides, payments and setting preferences etc.
- Fleet companies will be interacting through a CFP to set the availability options and pricing. They will also get information regarding the payments received.

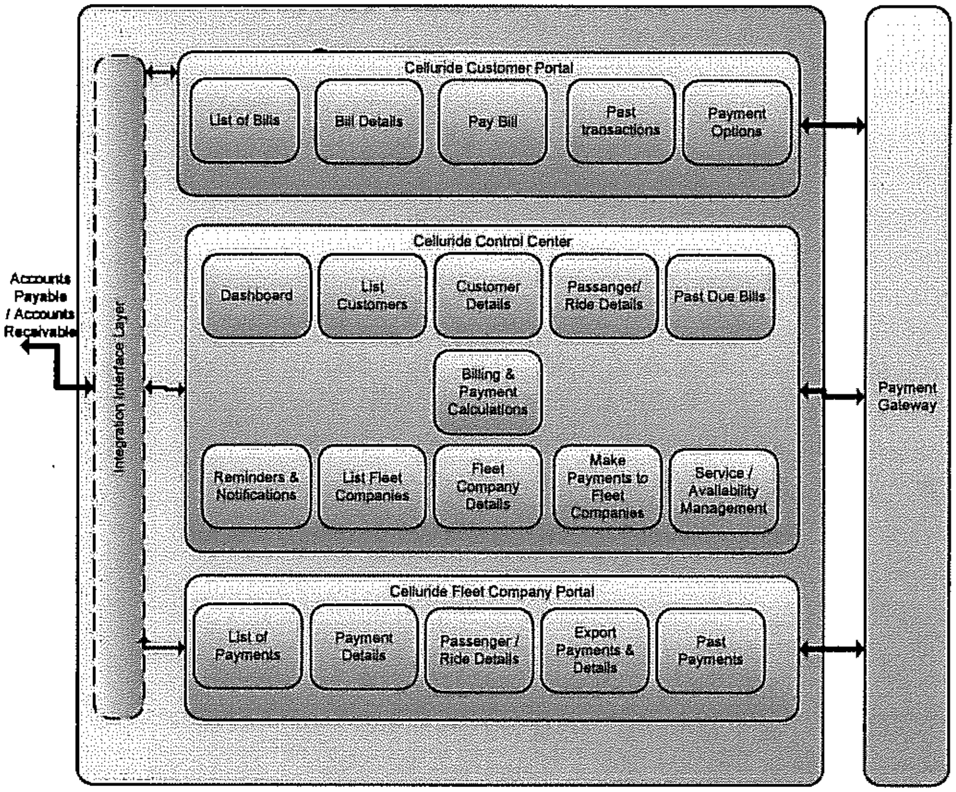
- A web service layer will be provided for integration purposes. One such example is that which the availability / pricing can be automatically communicated from the Fleet companies.
- Search / Availability Match module is responsible for taking the ride requests (reservations) and check for the available fleets and price match.
- Schedule / Pricing module is responsible for marinating the schedules and the prices for different fleet companies that will be available to match at the time of ride requests.
- Billing / Payments module encapsulates all the billing to customers and payments to fleets. This will also contain the modules to calculate the commissions and manage the payment transactions through the control center.
- Logging module will be available log errors.

3. Control Center Module

Overall the high-level architecture is described in terms of components and modules. Further modules are categorized according to the functional groups. Going forward these can be further decomposed into class diagrams for static views and sequence diagrams for dynamic views.
Celluride Control Center is the module that provides the management of Celluride business flow. This will be mainly used internally by Celluride customer service representatives and employees. It consists of sub modules to manage Celluride customers and Fleet Companies and there by managing the transactions between them.

4. Billing & Payment Module

Following is a block diagram that shows the components that are part of billing and payment modules.



Celluride - High-level Block Diagram (Level II) - Billing & Payments

Billing & payment modules can be categorized in to three functional elements according to the targeted users. They are as under:

- Billing interface for the customer
- Payment detail interface Fleet companies
- Control center interface for managing billing and payments

Here are some details regarding each sub modules.

Billing interface for the customer:

<TBD: Info from John's prototype>

Payment detail interface Fleet companies:

Payment will be made to fleet companies by Celluride on a periodical basis. Payment calculations according to the agreement between both the parties. Further Fleet companies will be provided an interface to view the details when needed. Following are the sub modules that will be used by the Fleet companies.

- List of Payments : To list my current payments
- Payment Details: Details of each payment made such as amount breakdown and calculations.
- Passenger Ride Details : For each payment cycle, the list of rides can be viewed or exported for tallying
- Export Payments & Details : Export the details in a csv or excel format
- Past Payments : View all transaction history

Control center interface for managing billing and payments:

Summary page / Dashboard: When the control center user logs in, this is the startup page that s shown. This will contain summary information such as total number of customers/fleet companies/past dues etc.

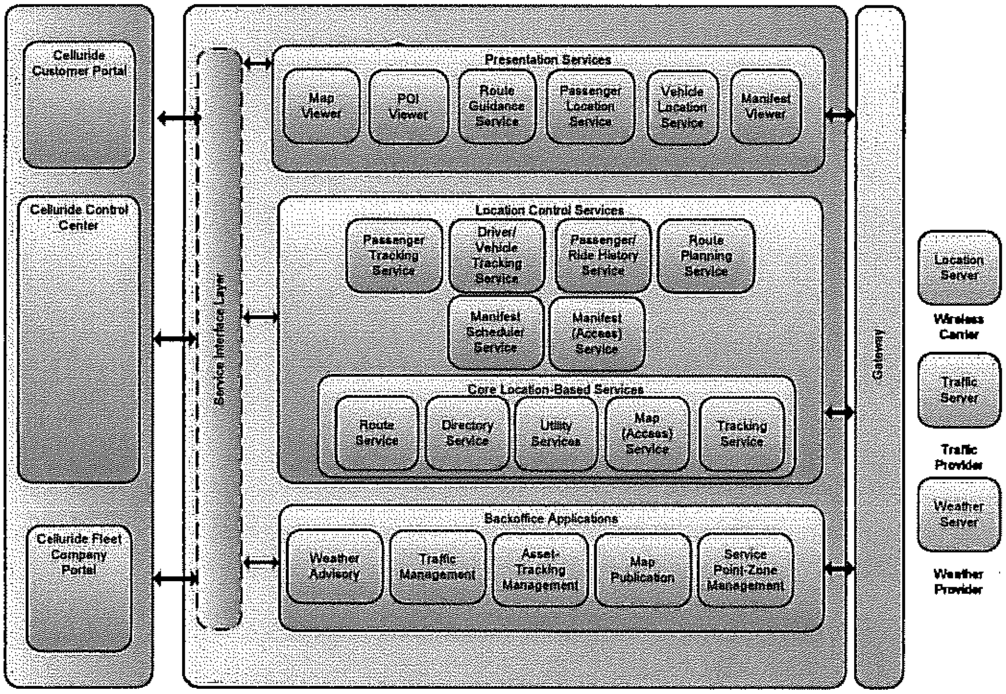
- List of customers
- Customer Details
- Passenger / Ride Details
- Customer Reports
- List of over dues
- Make a payment
- Billing & Payment calculation options: This will be used periodically to calculate and send bills and disburse payments.

- List of Fleet companies
- Fleet Company & Fleet details
- Fleet Reports
- List of payments made
- Payment details

5. Fleet Company Portal

6. Corporate Portal

7. GIS Module



Celluride - High-level Block Diagram (Level II) - GIS

7.1. Services

The Geographic Information System (GIS) module performs all location-based applications and services for the Celluride Service Platform. The GIS Module has three categories of functions/services:

- Presentation services – provide location-based content for passenger, driver, fleet, corporate and control operations.
- Location control services – provide location-based functions in support of Celluride service operations; includes core LBS functions.
- Backoffice applications – provides the location-based applications required to manage and administer location-based data.

Presentation Services

These services provide the means for all types of platform users to view and interact with location-based content.

Map Viewer – renders stylized-symbolized maps for display and interaction.

POI Viewer – renders service point and other point of interest information for viewing and interaction.

Route Guidance Service – renders route turn-by-turn directions, navigation aids and in-route progress information.

Passenger Location Service – renders detailed passenger location information; renders Passenger Location Reports.

Driver-Vehicle Location Service – renders detailed driver-vehicle location information; renders Vehicle Location Reports.

Manifest Viewer – renders manifest information for viewing; supports the ability for users to view this info in a list and/or on a map, and be able to link between the two, select entries for detail view, etc.

Location Control Services

These services create, access and revise location-based content for use throughout the system.

Passenger Tracking Service – accesses current/recent location-time-status of one or more passengers; creates Passenger Location Reports.

Vehicle Tracking Service – access to current/recent location-time-speed-status of all vehicles for one or more fleets; creates Vehicle Location Reports; also calculates and reports ETA information updates for pickups, etc.

Manifest Scheduler (Passenger-Driver Match) Service – uses location-based information for drivers and passengers and planned routes to match a driver(s) with one or more prospective passengers, and optimizing Driver Manifests in the process. Creates, accesses and revises "candidate" proposed (changes to) Driver Manifests. A match/scheduler algorithm optimizes manifests based upon passenger/driver locations, service times (pickup & drop-off locations and times) manifest schedule (i.e., available time slots), and planned (preplanned) routes (i.e., optimizes routes for "traveling salesman" situations). Thus, the match/scheduler algorithm optimizes driver downtime and passenger ride time.

Passenger/Ride History Service – accesses and updates location-based ride history information for use in future route planning and customer support purposes. Used to reduce route planning efforts for repetitive ride requests, e.g., a passenger that makes a series of service requests involving the same pickup and drop-off location-times. Also used to confirm or validate ride history in support of customer service and set/update/optimize customer preferences.

Route Planning Service – enables a Celluride operator to plan and replan routes. Replanning is (may be) necessary when conditions warrant (e.g., traffic problems, insert/delete passenger from manifest, unplanned events, etc). The idea is to normally do replanning in the CCC, through a Celluride Operator using the Tracking Management Application, not on-the-fly with the Driver. This preplanning will be done in the background so that the Driver is just driving and stepping through and working off the dynamically optimized manifest and route.

Manifest Service – creates, accesses and updates a Driver Manifest for a specified driver(s). [Note: The Manifest Service merely creates, updates and provides access to manifest data. Manifest optimization (the smarts) is accomplished through the Manifest Scheduler Service.]

Core Location-Based Services

These services are the basic building blocks for other location-based services and applications. [Note: These services correspond with the OpenLS platform standards, which are available from several vendors.]

Route Service – determines a route between two or more points.

Directory (POI) Service – accesses service points and other points of interest information that may be required for navigation.

Utility Services – performs basic geospatial utility operations, including geocoding (address→lat/long) and reverse geocoding (lat/long→address).

Map Service – accesses basic map layers used in compiling maps for presentation.

Tracking Service – accesses tracking information for passengers and drivers (i.e., location-time-speed-status).

Backoffice Applications

These applications provide management and administration capabilities in support of operations.

Traffic Management – used to monitor and report traffic conditions; creates Spot Traffic Reports.

Asset-Tracking Management – used to monitor and report the location-time-speed-status of active passengers and drivers, as well as reserve assets (vehicles and drivers); also able to access and view Driver Manifest information to determine in-route progress and progress against manifest.

Map Publication – used to import and manage geospatial data and publish all maps (create templates) from geospatial data; creates the map template with features, symbols and styles for Base Maps and any other Maps.

Service Point- Zone Management – used to import and manage directory data containing service point and other point of interest information; also used to create and manage zones (geographic areas) that are used in operations.

Weather Advisory – used to monitor and report weather conditions; creates Spot Weather Reports.

7.2. Primary Information Types

The following list contains the primary location-based information types (classes) for the system:

Base Map – contains the essential layers of geospatial data required for creating "base" maps, including hydrographic, cultural, and transportation data, etc; maps created by the system for presentation to users would normally contain a base map with one or more overlays containing such information as routes, service points, passenger tracks, vehicle tracks, zones, etc.

Map – any rendered map presented for Celluride business operations.

Address – the street address for a location of interest (i.e., building location, street intersection).

Roads & Navigation – contains road and associated navigation data suitable as input for route planning.

Route – A planned route containing one or more stops, with associated route guidance (turn-by-turn directions and navigation aids).

Route Directions – the turn-by-turn directions for planned routes.

Navigation Aids – navigations aids that are used for guidance.

Service Points – contains information about hotels, service stations, restaurants, business centers, transportation nodes, and other points of interest suitable for route planning.

Passenger Tracks – contains passenger [location-time-status] for active passengers.

Passenger Location Report – a location-sensitive passenger location status report.

Passenger History Log – captures the ride history for passengers in terms of pickup & drop-off times, routes, drivers, etc.

Vehicle Tracks – contains driver/vehicle [location-time-speed-status] for active drivers; includes links to active routes.

Vehicle (Asset) Location Report - a location-sensitive vehicle (asset) location status report; may be used in a number of ways, including, for instance, as an alerting mechanism to indicate to a passenger that the driver's ETA at a pickup location is delayed, or as a way to indicate back to the CCC that a driver is off-duty or has an unexpected schedule delay.

Zones – any geospatial areas that are used in operations, e.g., unit areas that may be defined for use in fleet asset management and transportation planning.

Spot Weather Report – a location-sensitive weather information report; used for route planning.

Spot Traffic Report – a location-sensitive traffic information report containing such information as known accident locations, locations for streets under repair, traffic flow data; used for route planning.

Driver Manifest – the location-based information and associated passenger and driver information comprising the manifest for a driver (planned passenger pickup locations and service points, including breaks) and associated planned routes. [Note: The business logic that drives the creation, update and management of the manifest is outside the GIS Module, under CCC. However, the application uses GIS module functions extensively to create, revise, optimize, view and interact with manifest information.]

7.3. External Interfaces

Near-real time data services

Location Server – 3rd party wireless provider service to access locations of mobile terminals.

Traffic Server – 3rd party service to access local traffic information for route planning and replanning purposes.

Weather Server – 3rd party service to access local weather information for route planning and replanning purposes.

Other 3rd party data

Base Map Data – 3rd party data containing basic geospatial data required for maps.

Route & Navigation Data – 3rd party data for detailed route and navigation purposes; also contains address information.

Service Point Data – 3rd party directory information for relevant service points required for route planning and navigation.

8. Passenger / Driver Modules

<TBD: Harry, please provide some info here>

9. Scheduling / Availability Module

10.Search / Match Module

11. Reports

12.Error Handling / Notifications

- Billing notifications to Customers
- Past due notifications to Customers
- Payment notifications to Fleet companies
- User interface error handling

13. Security

Authentication

- Authentication for Customers, Fleet Companies & Control Center

Encryption

- SSL will be used for all the payment and billing activities.
- Passwords that are stored should be encrypted.
- Billing options will be stored in the Celluride database.
- Credit card information and/or banking details will be kept with the payment gateway provider and not in Celluride DB.

14.Usability

15. Scalability & Performance

- App server level clustering
- Database shadow

16. Deployment

The following are some implementation & deployment details for each categorized layer:

Presentation layer

This layer will be implemented mainly using JSPs /HTML and javascripts. Java faces will be used where necessary. This layer is responsible for rendering all the user interface elements. Apache Tomcat will be used to server the presentation pages. Targetted browser for the initial release are IE 5.0 & above, Netscape 7.0 & above.

Business logic layer

The control center's business logic will be abstracted into a middle tier and will be built as managed components. Several stateless and stateful EJB's will hold the business logic and will be available is an clustered environment. Entity beans will be used to interface with the database. This layer will be implemented using JBOSS.

Database layer

This layer abstracts the access to the underlying database. Mysql database will be used to store the data. Please refer to the Datamodel section for details

Appendix A : Open Issues / Questions

- Business model for payment and revenue generation.
- Payment gateway vs. third party billing provider - Feasibility.
-

Software & Providers List

Platform Middleware, Map Content, Navigation, UI

November 19, 2006

Table of Contents

ABI Introduces a navigation solutions database

ABI Research has launched a navigation solutions database. It includes information on pricing, hardware and features searchable for hundreds of navigation devices and services. According to ABI, the new database includes nearly all the world's navigation offerings. The continuously updated database can be searched by a number of identifiers, including: feature set, traffic connectivity, embedded OS, hardware characteristics, screen size, and original device manufacturer.

1. Introduction

…Purpose

This document includes software solutions and application programming interfaces to support LBS.

LBS services require access to more resources within the operator's network than just location such as subscriber validation and subscription verification and application-level billing.

Comment: LBS middlewares contain Web-Services API (AAI - Advanced Application Interface) for accessing common network services, such as:
• Location requests (based on MLP 3.0)
• SMS and WAP-push message delivery
• Subscriber validation and subscription verification
• Application level billing

Some of these solutions focus on providing tools to the wireless telecommunication carriers, while others focus on the handset manufacturer, while still others provide a suite of functions that contain geoprocessing applications and web services.

This document presents an assessment of the value chain of LBS

- Location technology vendors
- Equipment vendors
- Mobile network operators
- Handset manufacturers
- LBS middleware providers
- Content providers
- Application developers
- Portals and location data brokers

Comment: Like Openwave Location Studio?

Comment: Will these be the navigation brokers?

2. Platform Middleware Comparison

...LBS middleware provides the following functionality (out-of-the-box):

- Provide services based on a user location
- Invoke fetching position as needed to update users' location information
- Generate queries for a user's current location, with or without heuristic refinement
- Generate proximity queries for pairs of mobile objects
- Maintain a cached list of mobile objects to monitor
- Perform periodic updates of one or many cached mobile objects to monitor
- Receive notification based on an event generated by a mobile object and its proximity to a static or other mobile object

Spatial and Relational Queries
The following spatial and relational query functions are usually supported by most LBS middlewares:

- Find all features that satisfy a custom SQL Where clause
- Find features closest to a specified location based on straight line (as-the-crow-flies) distances
- Find features closest to a specified location based on travel time or distance using a real street network
- Find closest features along a route. This could be performed on any previously generated route. For example, this search could be used to return all the gas stations along the route
- Find features closest to a specified location within a specified radius
- Find all features that contain a specified location
- Find features that are within a specified distance from a specified feature
- Find features within a specified region. This region could be comprised of the ring, ring sector, or disk sector returned from the position determination technology.
- Find closest features based on travel mode. For example, the closest ATM machine based on driving distance.

Comment: Passengers may request to stop at an ATM on the way to their drop off location. Knowing the location of an ATM will allow the driver to deliver the passenger to both locations quickly and effectively

Some LBS middlewares come with Content Management Services that enable:

- Navigate and query a hierarchy of categorized content
- Manage a user's personal location history
- Manage a user's personal location and route bookmarks

- Interface with external systems
- Refine a location by performing spatial proximity queries

TO DO:

- see if there is difference in product offerings for web based vs cell phone based map services, tracking
- would we want to use ESRI's Tracking Server for the website app?
-

Comment: •Tracking Server Author—Allows users to create a tracking symbology file that contains the symbology for displaying real-time tracking services.
•Tracking Server Designer—Allows users to build and customize a tracking Web site by specifying the ArcIMS image service and real-time tracking services to include in the Tracking Viewer Web site.
•Tracking Server Manager—Serves as the control center for the Tracking Message Server, where users can customize settings to tailor the server's functionality to their needs.

No to Kivera, Webraska, LocationNet
Kivera is now part of TeleCommunicationSystems

Comment: Why not?

Comment: The platform seamlessly integrates with the operator's existing service network, user provisioning, billing, and OMAP systems.

LocatioNet's powerful add-on GIS Engine supports map rendering, geo-coding and routing for all LocatioNet Platform integrated LBS service applications.

- Are all of these hosted?
  - Do we want to setup our own open source server and maintain it?
  - Which open source LBS platforms are good? GeoServer and MapServer are good GIS map servers, but are not LBS platforms, per se.
- Other LBS middleware providers that are not included in the table include:
  - Autodesk Location Services
  - AltaMap
  - Cellvision (US or Europe?)
  - Genasysw
  - Mobilaris – NO
  - Openwave
  - PTV – NO
  - Reach-U – NO
  - Redknee – NO
  - Telenity2 - NO
  - Cityneo (France)
  - MapInfo
  - Wavemarket
  - GeoMicro
  - Integraph IntelliWhere
  - OracleAS Wireless LBS platform
    - Allows to add maps, driving directions, and information from Yellow Pages to the applications.
  - IBM WebSphere Everyplace Server LBS (requires WebSphere Application Server)

In conclusion

- Autodesk LocationLogic
- Openwave
- Wavemarket
- Telcontar
- Oracle (cheaper than going with LBS middleware / just having a db?)

Comment: powers Nextel's and Sprint's location-based services (MapPoint web services are also used by Sprint?)

Factors:

- easy of migration? (from ESRI ArcWeb to Telcontar)
- mobiles with GPS support (there are about 10 GPS cellular solutions as of 2005)

LBS middleware vs hosted LBS Web Services
Benefits of a hosted LBS Web Services:

- provides content and geo capabilities without having to purchase and maintain large datasets
    - This can result in significant savings in time, expense, and computer resources
- Ability to combine multiple services and integrate them into the application
- No need to purchase hardware or software or hire additional staff
-
- Q: What about integration with other systems like billing, CRMs (reservation system), etc. ?
    - A: Everything will stay the same except the plug-ins/connectors b/w our "GIS module" vs a hosted LBS service

LBS middleware

LBS middleware from:	Autodesk LocationLogic	Telcontar DDS	ESRI Tracking Server?	MapPoint Mobile Location Server (MLS)	Oracle Db 10g		MapInfo miAware
					Locator	Spatial	
Platform / Application Server Support	BREW, J2ME, WAP, HTML, messaging (SMS) J2EE	BREW		Microsoft	J2EE		J
Db Server				SQL Server 2000			
Hardware	Sun Solaris Server, BEA WebLogic, Oracle Spatial						
API	Java and XML API (HTTP/HTTPS)		XML API (HTTP & SOAP)	SOAP/XML	J2EE 1.4 Web Services, XML		
Network Service Integration / Plugins				Sprint, Bell Mobility, O2, TellaSonera, Teydo (location aggregator)			
Billing & Reporting	Y						

Comment: Also works with WiFi providers

Comment: The BREW platform's ability to support gpsOne, voice capabilities and client side mapping makes it the logical platform to interface with Telcontar's location-based software solutions.

Comment: SDK toolkits

Comment: BREW support?

Comment: Works with Java, J2ME, .NET, C++ and more

Comment: XML web Services compliannt with OMA and OpenLS

Comment: No SOAP

Comment: Billing records are generated for both platform and application level operations,

Features/Components:						
Route Service	Y (OpenLS)				Y	
Directory (POI) Service	Y (OpenLS)					
Geocoding	Y (OpenLS)				Y	
Reverse Geocoding	Y (OpenLS)					
Map Service	Y (OpenLS)				Y (WMS)	
Tracking Service	Y (OpenLS)					
Event/Alert Service	Y					
Proximity Searches						
Type of Mobile	J2ME, BREW			SmartPhone, PocketPC		
Pricing / Fee				~$6,500 annual (includes 15,000 positions)		
Clients/Users	Disney Mobile	Appello, Networks In Motion, TeleNav (Web: Ask.com, Google, Yahoo); Rich Map Engine for Motorola's UK TrafficMaster's Smartnav		Teydo MobiSPOT, Sprint, Bell Canada; Cubistix	Autodesk LocationLogic, Integraph IntelliWhere, Telcontar DDS; Teydo FleetOnline & MobiSPOT NTT DoCoMo, JPhone, KDDI, Hutchinson 3G	

3.1 Autodesk LocationLogic

3.2 Telcontar DDS

3.3 ESRI

3.4 Microsoft MapPoint Mobile Location Server (MLS)

MLS API is based on SOAP Web services, so it interoperates with many different client platforms (i.e., .NET programming languages, Java or Perl.)
Three providers that work with mobile operator networks are available with MapPoint Location Server:

- **Sprint:**

Comment: This includes rendering?

Comment: LBS middlewares provide event notifications. Subscriptions can be configured to keep track of how and when to notify a subscriber of an event that takes place. For example, a subscription may be configured to send a notification via SMS when the driver is near or has reached the pickup location and waits for the passenger. Furthermore, LBS middlewares provide the targeted functions for managing and configuring subscriptions that respond to events generated from multiple sources. Events may be triggered from dynamic content such as real-time traffic information updates.

Comment: Is part of any of the above?

Comment: Some users use the LBS platform by deploying it themselves.

Others, use the hosted LBS web services

Comment: Disney Mobile has selected Autodesk to be their LBS partner to enable its new suite of cellular services with location-enabled applications. Disney Mobile will launch its services in June 2006. A division of Disney Internet, the company will operate as a Mobile Virtual Network Operator (MVNO) and will be leasing a frequency spectrum from Sprint (CDMA 1xEVDO).

Comment: In the UK -- over 6,000 registered companies (2004)

Teydo BV is a leading LBS provider that owns and operates the MobiSPOT™ Location Based Service aggregator platform with global reach for Positioning, SMS and billing. ... [1]

Comment: Teydo LocationXS "web service" (website)
This is used by Teydo's partners serving the LBS. Location XS is built on Teydo's FleetOnline web service and Trim ... [2]

Comment: Internal LBS based service helps fleet owners control their fleets and reduce operational costs

Comment: FleetOnline has 15,000 subscribers (2006)

Comment: rovide delivery drivers with point-to-point directions and maps for all of their routes and reduces the response time and delivery costs for special deliveries by dispatching drivers c ... [3]

Comment: LBS aggregation platform

Comment: To use MLS in commercial settings, a list of phone numbers that a particular MLS installation is authorized to locate needs to be specified (under a contract) with the carrier. ... [4]

- **Bell Mobility (Canada):** A location provider that works with the Bell Mobility network.
- **Teydo (Europe):** A location, SMS notification, and billing provider that works with the Teydo network

MLS supports plug-ins for Sprint in the U.S. and for Bell Mobility in Canada

A GIS needs to be used along with MLS to translate the coordinate to a street address, correlate it with points of interest, and render maps. Microsoft provides the MapPoint Web Service (MWS), a hosted GIS solution.

Comment: MWS is used with MLS so that we won't have to host the data, maintain it, update it, etc…

Q: Do we want to host our data?
OR, use a web service like MWS, ArcWeb, etc to tap into the data?
If the latter, then it might be better to host the entire LBS solution with map data.

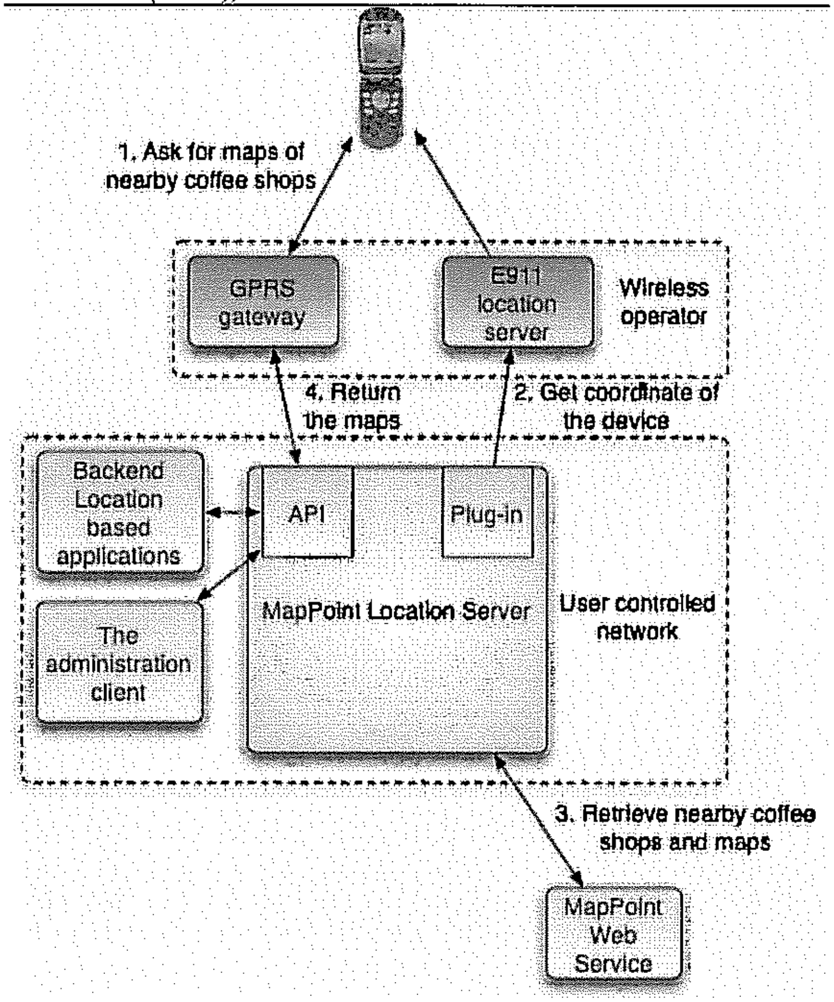

1. Ask for maps of nearby coffee shops
GPRS gateway
E911 location server
Wireless operator
4. Return the maps
2. Get coordinate of the device
Backend Location based applications
API
Plug-in
The administration client
MapPoint Location Server
User controlled network
3. Retrieve nearby coffee shops and maps
MapPoint Web Service

Figure: **MLS Request Process** – A user sends a request to MLS from a Smartphone. MLS then requests user location from the wireless carrier, uses the location to find nearby coffee shops, and returns the highlighted map (using a GIS, the MapPoint Web Service) to the Smartphone   Source: Microsoft

3.4.1 Tedyo MobiSPOT (Europe)

Comment: So what?
Is the Teydo platform similar to DIP?

Their LBS is independent and provided by different LBS providers (MIcrosfot, etc).
Teydo agglomerates location from the GSM providers and sends them to the LBS providers...
We don't need to do location agglomeration because Sprint has an open location API

Teydo is..

MobiSPOT is connecting the MapPoint Mobile Location Server (MLS) to all European GSM operators for LBS positioning and SMS messages. Any developer or partner of Microsoft MapPoint has access to the MobiSPOT platform using the standard API which comes with MLS.
MobiSPOT connector enables any MLS partner to connect to more than 100 million GSM subscribers
The mobile-operator independent API is mapped to individual mobile operators through the MobiSPOT location provider
Tedyo has appointed Fresca to host and manage their system for network performance reasons.

Comment: WHY? Too much of operation costs?

MobiSPOT API connection licensing

- Waterfall-style pricing model.
- No upfront access fees.
- Annual licensing fees including 15.000 position requests.

Transaction-based Fees
For example, Teydo (European aggregator for LBS to GSM operators) bills the API licensee on a monthly basis for number of transactions used by their customers. Transactions are billable if they use any of the following MobiSPOT Web Service functionalities:

- Location requests,
- SMS messages (both MT and MO),
- Premium SMS, and
- Credit card transactions.

API payment options
Payment of executed transactions is on monthly basis following the end of each month. Partners can always check their traffic through the online Tintis web monitor. If required Teydo can collect the payments from customers through Premium SMS and/or credit card.
When using the Premium SMS reversed billing option within MobiSPOT end users are charged for all transactions on their

monthly mobile phone bill from their operator. The out payment is on average 75% across all GSM operators. The out payment term is 60 days after the end of each month.

API Credit Card out payment

When using the Credit card billing option within MobiSPOT end users are charged for all transactions on their credit card. End users can either top up their account or you can define a fixed monthly subscription fee which is charged automatically. The out payment is on average 95% across all credit cards. The out payment term is 30 days after the end of each month. For a detailed cost per transaction please contact us.

MobiSPOT Location Server Pricing Model

Platform Access Fees: US $ 6,000 annually including SLA fees, Operator Connection fees and 15.000 position requests.

Service Level Agreement / Operator Connection Fees: For Partners whom prefer a monthly charge the following fees are applicable US $ 350 monthly (included in Platform access fee). New Teydo operator connections for LBS are automatically accessible for all partners.

3.5 Oracle Db 10g

Oracle Locator

Oracle Locator is a feature of Oracle Database 10g. Oracle Locator provides native data management (spatial object type storage), query, analysis, and indexing of location data, accessible through standard SQL. These limited features are sufficient to support most LBS applications, making the Oracle Database 10g a platform that supports LBS.

Oracle Spatial

Oracle Spatial extends the core location features included in every Oracle database with Oracle Locator. Its advanced data manipulation and spatial analysis features include buffer generation, spatial aggregates, area and length calculations, and linear referencing.

Oracle Spatial and Oracle Locator manage location data in a native type within Oracle Database 10g.

Oracle Spatial and Oracle Locator have been adopted as the preferred location platform by leading GIS and LBS vendors.

Comment: As a LBS platform or does it need to tap into LBS providers/services

Can we host and serve our own data via Oracle Locator to a cell phone?

Oracle Spatial is an option for Oracle Enterprise Edition that provides advanced spatial features to support high-end GIS and LBS solutions. Oracle Spatial is a spatial data engine for complex spatial/GIS solutions (i.e., spatial buffer functions directly in the database server.)

Comment: Is Oracle Spatial for GIS and Oracle Locator for LBS.

Oracle Locator was developed as a spatial database backend to be used in combination with 3rd party GIS tools, or where limited spatial operators and a limited set of spatial functions are required from the database (e.g., enterprise and wireless location services). Oracle Locator is well suited to carry out the relatively simple spatial operations required by wireless LBS.

NTT DoCoMo, JPhone, KDDI, Hutchinson 3G already use Oracle for their LBS solutions. The J-Phone J-Navi LBS applications were written in Java and run on Oracle Spatial. Java Server scripts running in the database and mid-tier provide lightweight and scalable geocoding, map rendering, and location capability. This particular deployment runs nearly all of its LBS functions directly from the spatial database and is able to achieve scalability requirements of 30,000 user sessions per hour. The results is the ability to deliver over 1 million color vector and raster maps per day to a new class of GPRS and UMTS enabled multimedia handsets. The average query processing is less than 200ms, and average download time is two seconds. The J-Phone deployment, in combination with partner technologies and services, leverages performance-enhancing features like caching, parallelism, partitioning, and high availability. This is particularly relevant to wireless location-based applications where new application components may need to be created and enhanced regularly to differentiate service offerings.

Oracle MapViewer is an Oracle Application Server Java component and JDeveloper extension used for map rendering (using a J2EE service) and viewing geospatial data managed by Oracle Spatial or Locator. MapViewer provides services and tools that hide the complexity of spatial data queries and map rendering. MapViewer is designed to integrate with Location-Based services and applications.
Oracle Spatial is available on Oracle Enterprise Edition. It provides advanced spatial functions (including area, buffer, centroid calculations.)
Oracle 10g has built-in, standards-based locator capabilities to store, index, and manage location content and query location relationships using the power of the database. Oracle 10g Spatial adds spatial information management features such as:

- a network data model,
- topology (needed for proximity and search queries),
- GeoRaster support, and
- built-in geocoding

In conclusion, Oracle Locator was targeted at LBS and includes basic geometry types, indexing and geocoding. In order to provide the higher end functionality, a 3rd party LBS middleware is needed.
Oracle brings the performance, scalability, and manageability needed for LBS.

Oracle iAS LBS Platform
Oracle9iAS Wireless is a platform for mobile application development and deployment. It addresses the specific location-related requirements of mobile applications, such as location acquisition, provider selection, mobile positioning and privacy, and geo-fencing (region modeling).

Oracle9iAS Wireless simplifies access to a variety of location based service providers offering services such as geocoding, driving directions, yellow pages, and mapping. New LBS features were recently incorporated into Oracle9iAS like: OpenLS XML interfaces, location privacy management, friend finder infrastructure, map rendering tool, and region modeling utilities. These features work with the leading 3rd party mapping, geocoding and routing tools to bring the necessary reliability, security, and scalability for wireless and enterprise LBS applications

Comment: The LBS platform provides a quick and easy way for developers to add capablities to communicate with new services, This is an example of how you could setup your own geocoder and write a geocoder proxy which would query it for information, This example uses the Oracle 10G database as a geocoder provider,

Comment: Oracle has OpenLS XML,

NOTE: Oracle9iAS is not an LBS platform on its own, but taps into LBS platforms / services, such as MapInfo's miAware LBS platform or Microsoft hosted service, MapPoint, which can be used as a content provider for LBS.

Oracle and MapInfo
Location Gateway: Oracle 9i AS Wireless
Application Platform: Oracle 9iAS Wireless
Geospatial Platform: MapInfo miAware
LBS Applications: A wide variety of off-the shelf, completely customizable and 3rd party applications.

Comment: (This is similar to ESRI's ArcWeb sample apps that we are using for the prototype.)

Teydo MobiSPOT (uses Oracle)
The MobiSPOT platform enables partners to access LBS services through one API. Runs on Oracle database and application server technology and can handle up to 20 requests per second.

Comment: And Geodan Mobile Solutions as the integrator or GIS platform. Geodan Is a expert on Oracle

Comment: That is not enough for our needs

3.6 MapInfo miAware

LBS Hosted Web Services

How many transactions will we make?
During testing of the prototype our credit usage for ArcWeb Services was 2,694 of 7,437 credits. Note, that in ArcWeb Services case, one transaction can be 1 or more credits, depending on what services and maps are used.
TO DO: use the prototype for one user session and see how many transactions get used up. Take that amount and multiply by 24 (as in 24 hrs) * 365 (as in 365 days) to get the minimum transactions per year.
TO DO: The backend LBS provider can be switched…from ESRI to Telcontar and vice versa thanks to OpenLS APIs. What about the front end, the mobile Java app? What will it take to migrate it? Will the code stay the same or will it need modifications?

Hosted LBS Web Services from:	Telcontar (AJAX)	ESRI (ArcWeb)	MapPoint Web Service	MapQuest	PlaceBase	Oracle (eLocation
Platforms (Web and mobile)		J2EE, J2ME, .NET, ColdFusion MX BREW				
Application Server						
API		XML/SOAP API	XML/SOAP API			
Network Service Integration						
Features/Components						
Route Service	Y (OpenLS)	Y				Y
Directory (POI) Service	Y	Y				
Utility Services (Geocoding, Reverse Geocoding)	Y (OpenLS)	Y				Y
Map Service	Y (OpenLS)	Y				
Tracking Service	Y (OpenLS)	Y	Y			Y
Proximity Searches		Y				
Traffic Service						
Type of Mobile		Java with MIDP support				
Pricing / Fee*	$12/yr (+per transaction rate)+ $5 for each 1,000 transactions.	$1,250 for 100,000 transactions	$8K/yr (500,000 transactions)			
Clients/Users			(Location Server; Sprint,			

Comment: BREW support?

Comment: Comes with the Mobile Toolkit

Comment: BREW apps can tap into XML/SOAP API

Comment: With,
Time-Dependent Routing

This section describes how the Celluride platform can provide state-of-the-art reliable on-time pickup/dropoff service.

LBS middleware applications can generate time-dependent routes that account for time sensitive conditions such as turn restrictions, street closures and even real-time traffic events. This also allows for generating routes that are relevant for a specific time of day (i.e., dependent street condition data, etc). Moreover, time dependent routes can be generated or existing routes can be checked for their validity for a specific time.

Time-dependent link speed updates such as those caused by traffic events can be fed into the model through the existing LocationLogic dynamic content framework.

LBS middleware can also provide a means for identifying time dependent travel maneuvers within travel direction narratives.

Combining time-dependent routing with event alert/notification services will enable the Celluride application to send a notification to the driver only when a route exists that is more optimal than their regular route.

Comment: simple or multipoint routes

Comment: This includes rendering?

Comment: Is part of any of the above?

Comment: ArcWeb Mobile Toolkit consists of a Java 2 Platform, Micro Edition (J2ME) toolkit.

Applications built using the toolkit will run on devices that support MIDP.

Comment: As transaction quantities increase, prices drop.

Comment: Some users use the LBS platform by deploying it themselves.

Others, use the hosted LBS web services

			Bell Canada)				
Other		J2ME, Locate phones for Sprint, Bell Mobility; SMS on Sprint phones, get Sprint device status (on/off), Find where device stopped along path, get angle, distance, speed along path					

Comment: Did they deploy the Location Server or do they use the hosted LBS web service?

⬚ → OpenLS APIs. Mark separately if supported.
OpenLS allows seamless access to multiple content repositories
and service frameworks that work across many different networks

* High volume discounts to the pricing scheme as usage increases are also available

Telcontar AJAX Web Services

New as of May 2006. (I don't see anything new that distinguishes it from ArcWeb services from ESRI. ArcWeb exists 4 years already.)

ESRI ArcWeb Services

With ArcWeb Services, we can create a complete spatial solution without hosting any of the content or software on our own machines.

□ Tracking
□ Monitoring of vehicles
□ Creating geo-fences to determine if a field entity has strayed out of an assigned area
□ Field staff completing or receiving work orders wirelessly (this can apply to taxi drivers)

	ArcWeb Services Features	What we need / are using
ArcWeb Services Labs		
Access to Prereleased Technology	X	
Online Discussion Forums	X	
Functionality		
Route Finder	X	
Place Locator	X	

Address Finder	X	
Spatial Queries	X	
Upload User-Defined Points, Lines, and Polygons	X	
Reports—Demographics and Market Profiles	X	
Drive-Time Polygons	X	
Batch Geocoding	X	
Track Cell Phone Location	X	Using now
Map Datasets		
Streets	X	Using now
Addresses	X	
Routes	X	
Landmarks	X	
Aerial Imagery/DOQQ	X	
Premium Aerial Imagery/DOQQ	X	
Traffic	X	
Weather	X	
Thematic Data Sets		
Reports—Demographics and Market Profiles	X	
Business Listings	X	
Phone Number Location Lookup	X	
IP Address Location Lookup	X	
ZIP Code Data	X	
Postal Codes Canada	X	
Postal Codes Europe	X	
Airports World	X	

Comment: What about report that will show passengers usage/behavior on a map for the website admins?

Comment: What about report that will show passengers usage/behavior on a map for the website admins?

ESRI has a Java Mobile Toolkit to allow the creation of applications that can be downloaded to Java enabled cell phones.
These applications can send requests using ArcWeb Services for:

Comment: I need to download the app to the cell phone

- maps,
- geocoding,
- routing,
- address mapping, and
- searches.

The ArcWeb Mobile Toolkit consists of a Java 2 Platform, Micro Edition (J2ME) toolkit. The J2ME toolkit is used to build mobile client applications that seamlessly consume ArcWeb Services and is based on Connected Limited Device Configuration (CLDC) and the Mobile Information Device Profile (MIDP). Applications built using the toolkit will run on devices that support MIDP. eXtensible Markup Language (XML)-based services combined with the wireless application platform provide a rich environment for building browser-based applications for Java-enabled handheld devices.

Comment: So ArcWeb services are dependent on Java phones.

Route Finder Service

Response time is about 9-10 seconds. ESRI expressed (Mar 28, 2006) that there should be an improvement in the performance of the route finder service shortly as ESRI plans to have hardware and software upgrades.
Currently, ESRI is evaluating a new version of routing software.
There might be issues with drive times using ArcWeb

Comment:
Subject ... [5]

Geocoding

Users of ArcWeb Services
Sprint

The Sprint wireless base is 20 million phones and 100 million calls per day (2004). They have raised the intelligence level of their transmission facilities (for faster and more intelligent call and data handling), and in 2005 they will move to EV-00, which will offer data speeds of 300 kps to 500 kps.
This will enhance MMS (multi-media message services), web access and the general movement of data. They do extensive testing of middleware with their network. All new Sprint phones have Aided-GPS (A-GPS) and Sprint partners with other to provide "cool apps" to expand other services.
Sprint works with ESRI's' Web Services to address multi-platforms and the existing 20 million Sprint cell phones.
Sprint is among the fastest available in the U.S

MapPoint Web Services

MapQuest

MapQuest has started to provide more access to their development environment with their Open API initiative: www.mapquest.com/openapi . MapQuest will be moving more toward "open" systems, but only as it makes sense with their core mapping service and map licensing tools. These tools, called MapQuest Advantage, will probably see the most benefits from standards integration. http://company.mapquest.com/mqbs/1.html

Oracle eLocation

eLocation is an Oracle hosted service which can be used as a content provider for LBS. Oracle Spatial 10*g* Release 2 features eLocation Quick Start. Location service Java and XML APIs enable application developers to quickly and easily deploy mapping, geocoding, and routing services right "out of the box", from data stored in Oracle Spatial.

4. Map Content Providers Comparison

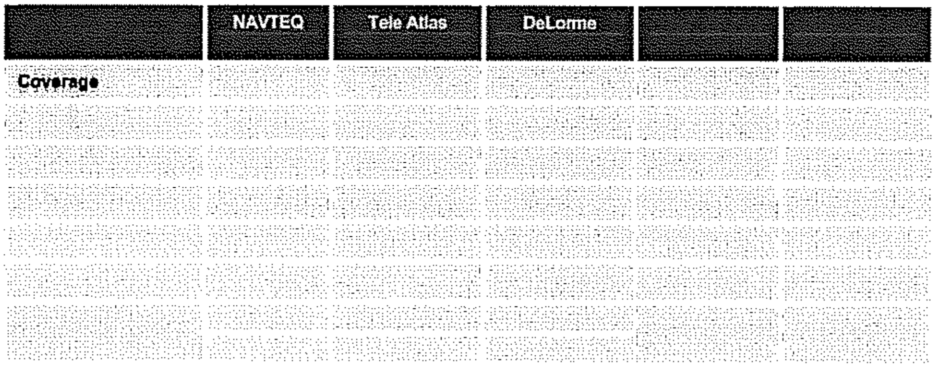
…

Maps providers:

- NAVTEQ
- Tele Atlas
-

GIS providers:

- ESRI
- Geomicro
- Istar
- Mapflow
- MapInfo
- Maporama
- Microsoft MapPoint
- Multimap
- Sagie Research (Enarro)
- Telmap
- Telenavis



	NAVTEQ	Tele Atlas	DeLorme		
Coverage					

NAVTEQ

Their first quarter report for 2006 was terrible, profits were 7.3% lower than most on Wall Street had expected. But I think the overall health of the company is very good. The disappointing results were a result of a weak dollar: a large portion of NVT's revenues come from outside the US, and the exchange rate did not move in their favor. Also, they spun off a business unit, and took a two million dollar one time loss when they sold the unit. Otherwise it looks like they are a great company, and very profitable.

--

Most LBS middlewares integrate with and stores large, multilingual data sets of POI features, street and road networks, and base map geometry data. An extensible database schema also supports dynamic content, such as up-to-date traffic, news, weather and other subscription services.

Applications build on these LBS middlewares can easily scale with database expansion. For example, a geographic service area can double by simply adding to the road network database, and the application can access the additional information with no extra development effort.

5. Navigation Platform Comparison

Q: Will we need this in addition to a LBS middleware that provides routing, etc?
No need for a GIS with NAVBuilder

Route Service

Directory (POI) Service

Utility Services (Geocoding, Reverse Geocoding)

Map Service

Tracking Service

Comment: This includes rendering?

	TeleNavigation	DeLorme	Networks In Motion		
Application Server					
API			BREW		
Route Service (Navigation)			Y		
Directory (POI) Service (Proximity Search)			Y		
Geocoding (Address Lookup)			Y		
Reverse Geocoding			Y		
Map Service (Map Rendering)			Y		
Type of Mobile			BREW 2.1.3, J2ME MIDP 2.0, CLDC 1.1		

5.1TeleNavigation

5.2DeLorme

5.3Networks In Motion

NIM provides wireless navigation solutions and LBS for GPS phones.
NIM developer platform for delivering location-centric information, offering continuous access to turn-by-turn navigation, local directories, maps, directions.
NAVBuilder enables developers to enhance their applications with location-based features including navigation, local directories, maps, directions, and more.

NAVBuilder offers a BREW extension or J2ME library for mobile applications, and a WEB interface for location-enabling web applications. NIM supports developers with carrier-grade network infrastructure and available premium training and support

OTHER

Nextel

Nextel offerings:

- Offer a Locate service for a $15/mo add-on...
- Offer a Navigate service, in real-time routing with voice updates...
- Offer a Tracking service, for either vehicles or an employee monitoring system

For the Locate service, Nextel is anticipating (via the GPS chips) to poll the cell phone and provide the accessible GPS constellation data to obtain a 10 second fix on the phone and also obtain the last five positions of the handset.Locate works with E911, or other applications that need a precise location.

Navigate is a standard offering for simply "pinging" a phone's location and in some cases is a difficult application to maintain with current information. While Navigate can be very useful, it is a very dynamic service to keep up to date. The Track applications were highlighted by a Teen Arrive Alive program. In this application, a teenager's phone is tracked so their arrival time can be determined and mapped. A very interesting problem: privacy vs safety concerns. Nextel will also have a location-based API to serve A-GPS phone applications and will also support Java.

Verizon

Verizon Wireless offers the service called "GeItNow." The traffic updates are verbose and understandable but not very helpful. It is difficult to determine whether the service could actually determine the location of the recipient, but it is suspected that the only way it could work is if each cell tower were broadcasting a traffic alert and when the phone was in proximity to the tower it obtained the broadcasted message. One of the alerts told of an accident; however the notice was received about 2 miles before the location was encountered; more or less a 2 minute warning -- Not enough time to make a change in driving options.

Qualcomm

Qualcomm supports BREW, a middleware that provides access to 100+ gpsOne-enabled handsets and 25 million devices (cell phones and others). There are 145 models and there are 27 manufacturers of BREW enabled phones. They also (via BREW) allow access to ArcWeb Services and GPS-enabled phones.

Qualcomm has changed the name of their location middleware from SnapTrack to QPoint. The QPoint solutions is intended to offer:

- Better Accuracy and Precision for finding locations
- Fewer keystrokes for data input
- Automatic location determination

6. Traffic Service Comparison

6.1Intrix

6.2NAVTEQ Traffic Service & CBS Radio

Q: Is this for Web services?

Comment: -http://www.lbszone.com/content/view/917/2/

Technical Architecture

System Use Cases

June 30, 2006

Document Contact Information:
K Kolodziej

Table of Contents

1. Introduction

This document contains system use cases for Celluride DIP platform, specifically for the identified functional modules and components (see the *Technical Architecture Document.*) Moreover, it highlights implementation tips to be considered when implementing each of the modules.
Section ___ of the document outlines the DIP system use cases. It first starts with the general use cases on authentication, etc. The use cases for the each of the modules follow.

This document is for the FINAL PRODUCT not for the PROTOTYPE.
This should not affect use cases like authentication; regardless of us hosting the LBS platform or using a hosted LBS web services. The LBS authentication is the same.

Comment: We will host our own LBS

Comment: We are using a hosted LBS web service

ADD here or the arch doc:

- Alerts & Notifications
- Most LBS middlewares will integrate with various network components such as systems monitoring, billing, and logging and reporting systems
-

1.1 Purpose

Following up to the *Technical Architecture Document* and the preliminary *User Needs Analysis (UNA) and Process Flow screen flows*, this document presents the system use cases and associates them to the preliminary UNA requirements – the user process flows and functional requirements[1] groupings -- and high-level Web Services (where applicable).

> Note: Refer to the *Technical Architecture Document* for system information pertaining to general application functionality such as user-login and permissions.

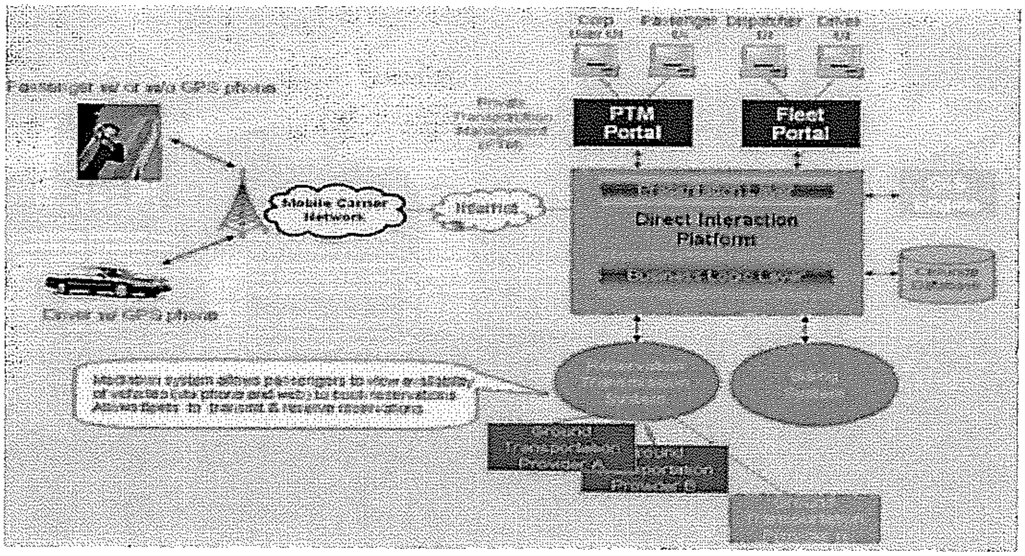
A *use case* defines a goal-oriented set of interactions between external actors and the system under consideration. *Actors* (e.g., admins, end-users or users) are parties outside the system that interact with the system. An actor may be a class of users, roles users can play, or other systems.
A use case is initiated by a user with a particular goal in mind, and completes successfully when that goal is satisfied. It describes the sequence of interactions between actors and the system necessary to deliver the service that satisfies the goal. It also includes possible variants of this sequence, e.g., alternative sequences that may also satisfy the goal, as well as sequences that may lead to failure to complete the service because of exceptional behavior, error handling, etc. The system is treated as a "black

[1] Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform.

box", and the interactions with system, including system responses, are perceived as from outside the system.

Thus, use cases capture *who* (actor) does *what* (interaction) with the system, for what *purpose* (goal), without dealing with system internals. A complete set of use cases specifies all the different ways to use the system, and therefore defines all behavior required of the system, bounding the scope of the system.

Furthermore, the DIP platform is envisioned to be made up of seven modules or components, as indicated below. This document associates the use cases for each module / component.



1.2 Definition of Terms

A variety of technical terms are used throughout this document. To minimize the use of footnotes and increase readability, definitions for commonly used terms are summarized below.

Term	Definition
Client	A software component that can invoke an operation from a server or conceptual role as originator of an operation request.
Displayed data	The subset of active data types that are actually shown on the map.
Interface	Named set of operations that characterize the behavior of an entity.
Map	Pictorial representation or portrayal

	of geographic data.
Operation	Specification of a transformation or query that an object may be called to execute.
Request	Invocation of a server operation by a client.
Response	Result of an operation returned from a server to a client.

NOTES:

- V1 needs to support
  - turn-by-turn navigation
  -
  -
- V1 does not need to support the following:
  - Traffic service
  -

2 General Use Cases

Overview: The following use cases are referred to as "general" as they are common throughout the DIP modules and the corresponding use cases. These general use cases are marked as one step / event in the specific use cases outlined later in the document. These general use cases are to be referenced for the more specific use cases whenever authentication, billing, __________ are at play.
General use cases that apply / are integrated within the other more specific use cases, include:

- *Security (Authentication includedpart of Security module?)*
- *Billing*

A **Security Module** a ..
LBS middleware supports requirements for privacy assurance management, implementing the administrative, technical and physical steps to pre-empt threats to the security and integrity of a subscriber's location information and personal profile.
LBS middleware supports a variety of end-user location privacy models, including opt-in, opt-out privacy (requiring that a subscriber must provide consent, or "opt- in," for a service prior to being located) and password protection. This offers the flexibility to protect user information without sacrificing application usability.
LBS middleware also supports subscriber anonymity, where access to services can be made available to applications without requiring knowledge of the personal identity of the subscriber, thus protecting the user's privacy.

*A **Billing / Payments Module** encapsulates all the billing to customers and payments to fleets. This will also contain the sub-modules to calculate the commissions and manage the payment transactions through the control server.*

Authorization – will users need to do this first time they use the app?
--
Celluride's Mobile Communication Server will use:
SMS
WAP
MMS

SMS
SMS can be used for communication / dialogue between the driver and passengers. (Note, use of SMS is not seen as mandatory.)
SMS can also be used to alert, both, drivers and passengers about delays. LBS middlewares provide event notifications where subscriptions can be configured to keep track of how and when to notify a user of an event that takes place. For example, a subscription may be configured to send a notification via SMS when the driver is near or has reached the pickup location and waits for the passenger. SMS can also be sent at a

Comment: ? the more specific use cases

Comment: LBS middleware like LocationLogic from Autodesk address threats to information security, privacy and integrity of a subscriber's Location information and personal profile.

Comment: Self-Provisioning
Subscriber self-provisioning is desirable whenever possible to reduce the impact of new services on the
operator's support organization. Self-provisioning is provided in Location Studio in two basic forms:
WEB/WAP and SMS interfaces for subscriber personalization of services and their privacy profile
First-time-through service handling to direct subscribers to personalization tools
If a subscriber accesses a service, and the subscriber has not yet authorized the service to make position
requests, there are two ways to support automated first-time-through activation of the service:
If the subscriber accesses a WAP service directly the application can redirect the WAP session to the
subscriber WAP provisioning interface to set the authorization. If access is via SMS a subscription

Comment: Mobile presence/instant messaging is hard.

SMS is going to cost because the operators charge an arm and a leg for SMS gateways and then they nickle-and-dime their customers to death on SMS termination charges.

But, we are just talking about 1 or 2 SMS msgs, I guess. So this will not be that expensive. This also depends on the SMS plan, Some plans have 300 free SMS msgs.

Comment: WAP is an open standard for communication between handsets and the Internet. WAP is a wireless communications environment for delivering web data to wireless terminals with minimal screen display. An initiative started by Unwired Planet, Motorola, Nokia and Ericsson to develop a standard for wireless content delivery on the next generation of mobile communicators. WAP strips all but graphics for display on small screens, such as mobile phones.

specified time interval with alerts such as how many minutes left before the driver reaches the pickup/dropoff location.
Furthermore, LBS middlewares provide the targeted functions for managing and configuring subscriptions that respond to events generated from multiple sources. Events may be triggered from dynamic content such as real-time traffic information updates.

Openwave Location Studio
A key feature of Location Studio is the support of Mobile Originated SMS for LBS service invocation. This feature allows for creation of easy to use messages that will perform a function within an application and return a SMS message with the result. An example would be to send a command word of "FIND" followed by a word describing what you are looking for "Pizza" and the application could then return information about the 3 closest pizza restaurants. This interface can be utilized to SMS enable applications that currently are only Web and WAP enabled.

Comment: Do we need location-enabled SMSs?

WAP

Comment: Transcoder. This will transform the WML request received from the gateway into HTTP request to be forwarded to the host server for processing.

WAP cell phone requests will pass through a gateway that will convert the WAP request to an HTTP and adds the Mobile Identification Number (MIN). The (application) server will authenticate the user and forward the request to the LBS server/service.
WAP gateway converts the WAP query into an HTTP request and adds metadata information about the user's mobile device such as the mobile device screen size and supported font types. The WAP gateway converts the XML content into WML as it stores the metadata of user's device specification.

Comment: This also has authentication processes

Comment: The WAP proxy feature requires that LBS proxy all traffic to and from the specified Client for the duration over which positioning requests are to be authorized. LBS will proxy requests to the client, translating all URLs to and from the client.
LBS will track state on requests to the client in order to translate the URL to the appropriate client URL on the response.

Wireless Transport Layer Security (WTLS) - The security layer of the WAP which provides privacy, data integrity and authentication for WAP services. WTLS, designed specifically for the wireless environment, is needed for the client and server to be authenticated in order for wireless transactions to remain secure and also because the connection needs to be encrypted. WTLS is needed because mobile networks do not provide complete end-to-end security.
In short, WAP comes with built-in encryption mechanisms.

Authentication + Billing Use Cases
The following use cases illustrate how DIP facilitates the delivery of a premium-billed LBS service using WAP (or SMS) as a bearer. This is a user-initiated ("pull") service and the key functionality required for this are:

- Protect the anonymity of the user at all times
- Allow for validation of the subscriber identity
    - Q: will the operator support premium billing for this user?
- Provide support for post-paid and pre-paid users
    - Q: For taxi service only? Or, black car as well?

Comment: LBS middleware support SMS

- Support billing based on an event relevant to the user (ie. obtained premium content) rather than on transactions measurable by the operator (location request, SMS message, content lookup, ...), especially where multiple transactions may be required to complete one relevant service delivery

Comment: ???

- Allow the Administrator to specify the amount of the premium charge
- Support any combination of WAP and SMS for service invocation and content delivery

Service example: a user invokes a service to find the location of the nearest driver that is currently available. The user is positioned by the LBS and the obtained position is used to search the Celluride db of available drivers within a pre-defined radius (or within zones) of the user's reported location.

Use case assumptions
The following assumptions are made:

- The use case / sample call flow below provides a high-level overview of functional interaction with DIP only, and does not deal with other aspects such as application authentication, subscriber privacy or quality of service issues

Comment: Like ArcWeb or LocationLogic authentication ?

Comment: But these modules are within DIP so include these

- Celluride is responsible for hosting its own database of drivers' locations, and a real-time feed that provides driver availability (simple busy/available indicator) for each driver.
- These use cases / call flows assume a North American CDMA network, and that the network is capable of positioning and providing location to the LBS Mobile Positioning Center (MPC).
- This call flow is equally valid in a GSM network, in which case LBS is configured as a Gateway Mobile Location Center (GMLC) and the mobile station is identified by its Mobile Station ISDN (MSISDN).
- The necessary integration work with the operator's billing platform (including Pre-Paid system) interface has been completed, and supports the detailed actions.

2.1 Authentication Use Cases

Wireless...

- Where/when will authentication take place?

- Through the LBS authentication service? Or sooner by utilizing WAP on the mobile device? Or both, on the device and on the server?

Using LBS Middleware
For example, Autodesk's LocationLogic provides secure access through a single point of authentication and a role-based service authorization. That is, users or applications accessing LocationLogic are authenticated by the platform and then utilize a hierarchy of roles to control access to different location-enabling services and content segmentations. This model allows for managing responsibilities of authentication and authorization in different tiers of the architecture depending on requirements.

Comment: Why would we need this? Corporate premium users will see more content?

All transactions against LBS middleware require positive authentication of the requestor's credentials. Basic authentication can be provided as a ClientID and password embedded with each request. More advanced authentication may be incorporated using external third-party products – for example, bi-directional SSL using certificates.
Once a client has successfully authenticated to LBS middleware their request is authorized based on privileges established within a client profile. The client profile is a set of parameters used for identification and authorization of access by the Celluride application. The Client profile contains parameters that define what services a Client may access from the network, the allowed quality of service, as well as physical identification data used for authentication and billing. The Client Profile is stored within LBS middleware.

Comment: Check

Using Hosted LBS Web Services
Using ArcWeb Services encryption is an important part of the ArcWeb Mobile Toolkit because the user name and password is sent to ArcWeb via HTTP. The user name and password are decrypted, and if the user is a valid ArcWeb user, a token is returned via HTTP which is then used to access ArcWeb Services. The token signifies that the user is authenticated to access ArcWeb Services. Authentication is handled by the Java API TokenServiceProxy class.

Comment: Tokens are still being used for mobiles?

Internet
There are four processes that are generally involved in online authentication, which are:

1. Registration - The first process is registering for a credential. This occurs once, the first time that a user decides he wants to access the KLC Portal's services.
2. First time service registration – Once a user has an authentication credential, he can let the service provider (agency, county) he deals with know about his/her credential. The user will have different "customer numbers" with the service providers he has already dealt with. Each service provider needs to link his credential uniquely to the particular customer number that they use when they deal with the user.

3. Requesting services - Before a user can access the Portal, the service provider first needs to be sure that it really is the user. User will be asked to present his credential and then check that it is valid. Once this step is complete, the service provider finds the customer number the user has previously linked to his credential.
4. Service delivery - The service provider applies their standard rules to decide whether or not the user is entitled to receive the service he is requesting (logging into).

Authentication solutions / implementation options generally fall under two categories. These are listed in _____ for which use cases are described in this section. More information on these and other authentication options can be found in IETF's survey[2].

2.1.1 Wireless Use Cases

2.1.2 Internet Use Cases

Authentication

Overview: Authentication is the verification process to assure that a wireless device and its user are compatible with and authorized to access a wireless network, more specifically, the Celluride DIP platform. The DIP platform will authenticate an incoming subscriber's identity. This process is accomplished through transmission of identifying data at the time of connection. Authentication will be used for Customers, Fleet Companies and Control Center.

Within the Celluride DIP platform, the following…

[2] IETF, "A Survey of Authentication Mechanisms:" http://www.ietf.org/internet-drafts/draft-iab-auth-mech-03.txt

Comment: Security
Authentication
•Authentication for Customers, Fleet Companies & Control Center

Encryption
•SSL will be used for all the payment and billing activities.
•Passwords that are stored should be encrypted.
•Billing options will be stored in the Celluride database.
Credit card information and/or banking details will be kept with the payment gateway provider and not in Celluride DB.
•

Comment: → CHECK OGC GeoDRM for info?

Comment: The overall security of a wireless application is only as strong as its weakest link, and in a mobile-commerce network, the weakest link is the mobile device.
→ HENCE, ENCRYPTION SHOULD HAPPEN ON THE MOBILE DEVICE
→ WAP's WTLS can be used for encryoption…?

The interceptable nature of wireless signals and the limited memory and computing power of most mobile devices leave wireless systems vulnerable to data theft

Comment: Additional security measures, such as secure connections and cryptography, are definitely needed, especially for those applications transmitting sensitive data. However, the adoption of HTTPS/SSL for secure connections or cryptographic standards, such as IPSEC and WTLS, brings with it the associated issues

What about XML messaging?

•*Security Assertion Markup Language* (SAML), which is a protocol to transport authentication and authorization information in an XML message. It could be used to provide single sign-on Web services.
•*XML digital signatures* define how to digitally sign part or all of an XML document to guarantee data integrity. The public key distributed with XML digital signatures can be wrapped in XML Key Management Specification (XKMS) formats.
•*XML encryption* allows applications to encrypt part or all of an XML document using references to pre-agreed symmetric keys.
The Web services secure XML protocol family (WS-Security), endorsed by ... [6]

- SSL will be used for all the payment and billing activities. → HAVE A USE CASE FOR BILLING
- Passwords that are stored should be encrypted.
- Billing options will be stored in the Celluride db.
- Credit card information and/or banking details will be kept with the payment gateway provider and not in Celluride DB.

–

–

ArcWeb Services has the Authentication Web Service

Encryption

Using ArcWeb Services encryption is an important part of the ArcWeb Mobile Toolkit because the user name and password is sent to ArcWeb via HTTP. The user name and password are decrypted, and if the user is a valid ArcWeb user, a token is returned via HTTP which is then used to access ArcWeb Services. The token signifies that the user is authenticated to access ArcWeb Services. Authentication is handled by the Java API TokenServiceProxy class.

Comment: Tokens are still being used for mobiles?

–

These use cases on authentication deal with the issue of access to Celluride's services outside of the Celluride architecture. Moreover, this is to ensure that a non-Celluride client cannot circumvent the Celluride security framework.

TO DO: Focus this use case to just outline the authentication and encryption steps.

Use Case Description		
Name	User authentication	
Priority		
Description	User is authenticated in order to be able to use the Celluride cell phone application	
Precondition	User name and password are already provided (cached) to be sent with each request	
Flow of Events – Basic Path		Service or Component
1.	User connects to the Celluride application	CDI
2.	Cell app sends a WAP request to the wireless network (username and password are included)	
3.	The wireless network directs the request to the LBS (LBS fetches the user's position by querying the wireless network – see Use Case)	GIS

Comment: Check the Arch doc – do we need a separate authentication module if the GIS/LBS will take care of it?

Use Case Description		
4.	LBS provides a gateway that converts the WAP request into an HTTP request (which is then forwarded to the LBS applications/services). The gateway encrypts [1] the username and password. A token is assigned. For Cell-Id positioning, the gateway also attaches user's mobile identification number (MIN) [2] to the HTTP flow as part of the HTTP header.	
5.	LBS gateway authenticates user If the user is a valid LBS user: - a token is returned via HTTP which is then used to access LBS applications/services; - the request is forwarded to the LBS component	Service Proxy
6.	Once the LBS receives the HTTP request from the LBS gateway, privacy checks are performed. A check is made to determine if the requested application is a LBS application. A check is made to determine if the application is allowed to receive user's location. If all checks are returned positive, the LBS sends the XML request to a mobile location provider (MLP) to find user's location [3]. (Is this done in this step or step 3?)	
7.	The user's location is returned in a response from the MLP	
8.	LBS now has user's location. It may remove the MIN and User from the HTTP request for privacy reasons, so users and their locations are not associated with each other. And, LBS will augment the HTTP request with the location information and send that request to the LBS application/service, where the closest driver(s) (or closest passenger(s)) are obtained. See Use Case ▒ in Section ▒ for specifics location requests.	

Comment: Q: The user name and password are decrypted here?

Use Case Description		
9.	The LBS application parses the location header for the needed location information. It then sends an XML request to a map service or location content provider (LCP) to fetch the base map for the area of interest. (Same would apply to fetch POIs – See Use Case)	
10.	The map service / LCP determines the requested content and returns an XML response.	
11.	The LBS application uses this information and returns the closest driver(s) or passenger(s)	
Flow of Events – alternative path		
5.1	User is not a valid user and is not authenticated	
Postcondition	User sees the response (a map or text) on his device. If desired, the application could return more content, such as POIs, wait time (for passengers' requests), and so on.	
[1] Encryption: Using LBS, encryption is an important part because the user name and password is sent to the LBS via HTTP. The user name and password are decrypted, and if the user is a valid LBS user, a token is returned via HTTP which is then used to access LBS services. The token signifies that the user is authenticated to access LBS services. In the case of ArcWeb services and the ArcWeb Mobile Toolkit, authentication is handled by the Java API TokenServiceProxy class. [2] The MIN is input into the request, and the user's location information is returned from that MIN in the response. [3] For Cell-Id positioning, the Location Finding Equipment determines that user's WAP phone is located in a particular cell sector by using his MIN, and returns the cell ID. The cell ID is returned to the MLP where it is translated into a latitude/longitude.		

Comment: Encryption needs to take place on the mobile before it sent over the network

Comment: But the request is initially in WAP and is converted to HTTP in the LBS

Comment: Where? In the LBS?

Comment: Tokens are still being used for mobiles?

Comment: The positioning method should be hidden from the use case? Shoujld be independent

Formatted: Bullets and Numbering

Use Case 1: Authentication employing Public Key Cryptography (Secret Type)

The common example of the secret type authentication solution employs Public Key cryptography. The following is a system use case describing the flow of events.

Use Case Description		
Name	User Public-Key Authentication	
Priority		
Description	A KLC is authenticated as part of logging into the Portal to gain access to a particular module(s).	
Precondition	Precondition: - The first party, the party wishing to be authenticated, has a secret. The second authenticating party does not have the secret, but has other information that is mathematically related to the first party's secret. This related information does not disclose anything about the original secret, but does allow someone to prove if the secret has been used.	Postcondition
Flow of Events – Basic Path		
1.	User connects to the integrated client application and requests login.	1P1
2.	Service provider (system) picks random number 'A' and encrypts it with user's public key into 'B'.	n/a
3.	User decrypts 'B' using private key back into 'A.' User hashes 'A' into 'C' and encrypts 'C' using a private key into 'D.'	–
4.	Service provider (system) decrypts 'D' with the user's public key into 'C.' If 'C' is equal to hash of original 'A' user is authenticated.	n/a
Alternate Flows		
Postcondition	- User is authenticated	

Formatted: Bullets and Numbering

Formatted: Bullets and Numbering

OTHER GENERAL USE CASES ON THESE:???
Find Ride Now and Find Ride Later will utilize geocoding → So reference this general use case on geocoding

2.2 Billing Use Case

LBS middlewares support integration with solution components designed to collect information on resource usage for such purposes as billing. Billing records are generated for both platform and application level operations.

For example, complete transactions such as "Find the nearest driver (taxi / black car)" are logged as well as intermediate transactions such as Authenticate/Authorize, "Get location," "Get base map from content provider" or "Get drivers location data from Celluride db."

- Celluride will charge $___ per each transaction as one whole transaction. We want to bundle it as one transaction fee. (We could break it down into subtransactions, but maybe for the taxi market to make it cheaper for basic services.)

Administrators configure when, where and how often the billing records are created. Billing data is extracted and stored in standard charging record (CDR) formats.

--

Geocoding and Reverse Geocoding Services

The Geocoding Service determines a geographic position, given a place name, street address or postal code. It also returns a complete, normalized description of the place (which is useful, say, when only partial information is known). The Reverse Geocoding Service determines a complete, normalized place name/street address/postal code, given a geographic position. Both the geocoder and reverse geocoder may return zero, one, or more responses to a service request, depending on subscriber request information, the algorithm being employed, and the match criteria.

Presentation Service

This service renders geographic information for display on a Mobile Terminal. Any OpenLS Application may call upon this service to obtain a map of a desired area, with or without map overlays that depict one or more OpenLS ADTs, such as Route Geometry, Point of Interest, Area of Interest, Location, Position and/or Address. The service may also be employed to render route directions from Route Maneuver List ADT and/or Route Instructions List ADT.

Comment: So this is for the cell phone users and not for the website users
→ WHAT is the Presenation Service for Websites?

It may be desirable to add a very low overhead dynamic update capability for positions obtained from the Tracking Services.

It may be desirable to allow other depictions of spatial information such as 3D views,

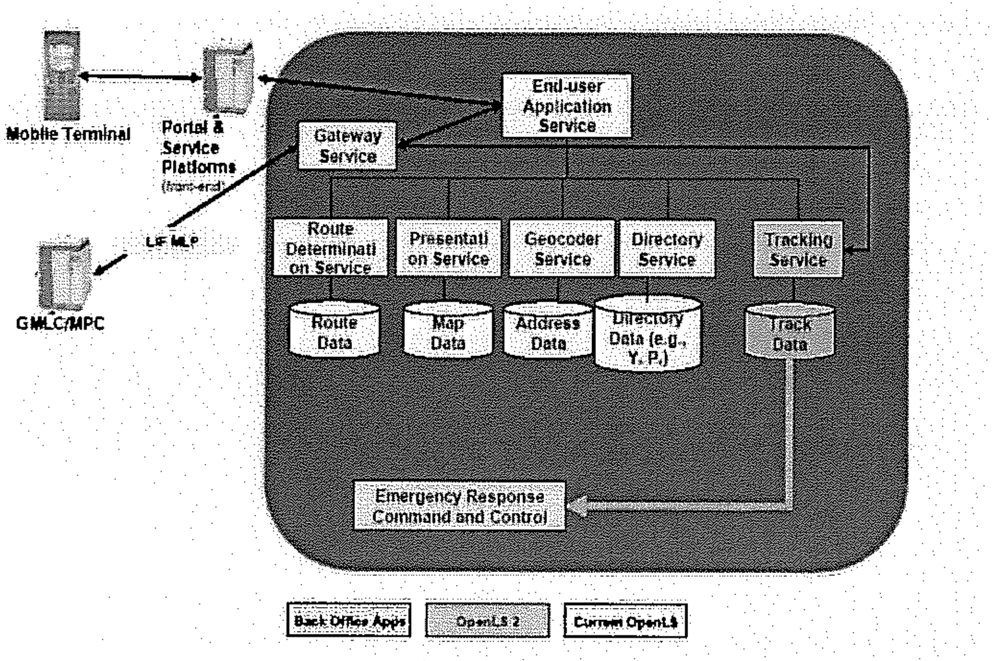
Route/Navigation Service

This service determines a route for a subscriber. The subscriber must use a navigation application to set up the use of the service. They must indicate the start point (usually the position acquired through the Gateway Service, but this could be a planned trip from a specified location, say, from their home), and the endpoint (any location, like a place for which they only have the phone number or an address, or a place acquired through a search to a Directory Service). The subscriber may optionally specify waypoints, in some manner, the route preference (fastest, shortest, least traffic, most scenic, etc.), and the preferred mode of transport. The subscriber may optionally store a route for as long as needed, thus requiring the means to also fetch a stored route.

Comment: This is for Drivers
If we support the Route Service
(and Traffic Service?)

OpenLS Engineering Viewpoint

Figure 25 conveys the general architecture.
Platform calls upon OpenLS services for location-based content and processing, as needed. In this figure we also show a possible configuration for the new traffic and tracking services, with back-office applications. Tracking Service will need timely access to mobile device location information. These services can access the GMLC / MPC directly using OMA/LIF MLP. The challenge OGC addresses through OpenLS is to define the foundational components and related specifications for near-time tracking, and navigation services.


Mobile Terminal
Portal & Service Platforms
(front-end)
GMLC/MPC
LIF MLP
End-user Application Service
Gateway Service
Route Determinati on Service
Presentati on Service
Geocoder Service
Directory Service
Tracking Service
Route Data
Map Data
Address Data
Directory Data (e.g., Y. P.)
Track Data
Emergency Response Command and Control
Back Office Apps
OpenLS 2
Current OpenLS

3 Subscribe Use Cases

Overview: A Subscriber module is responsible for all subscriber related operations:

1. *Provisioning*
2. *Authentication*
3. *Transaction*
4. *Billing*

Comment: Reference above general use case on Authentication

4 Passenger – Driver Interaction Use Cases

Overview: ...
To modules of specific importance:

- *A **Search / Availability Match Module** is responsible for taking the ride requests (reservations) and check for the available fleets and price match.*

  Comment: From the Arch doc.

  - indicate a car's location and status — vacant, occupied, or reserved
- *A **Scheduling / Availability Match Module** is responsible for marinating the schedules and prices for different fleet companies that will be available to match at the time of ride requests.*
- ***A Scheduling / Pricing Module...***
- *A **GIS Module** is responsible for identifying all drivers that are within the passenger's search radius (i.e., Find Nearest.)*

  Comment: Q: What is the default for the search radius?

  - *A Routing operation will be supported in version ___ for finding the shortest path or driving directions (both for passengers and drivers).*
  - *A Tracking operation will be supported in version ___ to continuously track (both passengers and drivers.)*
- ***A Location Determination Module...***
  - *GPS*
  - *LIF/MLP*
  - *BREW*

    Comment: We will be mostly a BREW app (b/c ...?)

  - *J2ME*

4.1 Passenger Viewpoint Use Cases

User = Passenger

4.1.1 Wireless Use Cases

Use cases (wireless):

1. Find Now – Passenger request ride now
2. Find Later – Passenger request ride later at time=T
3. View Reservations –Passenger views reservation
4. Paperless Voucher System

4.1.1.1 Find Ride Now

Use Case Description		
Name	Find ride now	
Priority		
Description	A passenger looking for a ride pushes a button on his mobile handset indicating the passenger's need for a ride.	
Precondition	User connects and is authenticated to the Celluride cell phone application	
Flow of Events – Basic Path		Service (Component) Invoked
1.	User connects to the Celluride application (START WITH USER IS LOGED IN TO THE APP)	CDI
2.	System authenticates the user (THIS STEP IS NOT NEEDED NOW...)	Subscriber Service Engine (SSE) == this is now the Registration module?
3.	User selects to find ride now in the Celluride application (RECOMMENDATION: Combine this step (and, hence, the functionality/UI) with Step 1)	
4.	System locates user's position and sends it to DIP (This info with the passenger's location is sent to passenger's carrier proxy server.) (The carrier proxy determines that this message belongs to Celluride and forwards it to Celluride's server.) (Celluride's server determines this message is from a passenger.)	Passenger Tracking Service (Location Server) → Driver/Vehicle Tracking Service (Location Server)

Comment: Will users want to book a cab in the morning for after lunch due to having a formal meeting?

•FIND THE RIDE LATER: this will probably change. We don't want users to book/reserve in advance... we only want the Find Ride Now because Find Ride later will allow users to cancel in the last minute... or forget to cancel and cause issues with reimbursements to keep customers happy.

How do these impact the backend? If at all...shouldn't, but this deals with the Tracking Service that knows where each vehicle is..but still same process as previous use case.
Same;
•pick up location as current location (in find now)

Comment: Requirement 1: We need to know where all the drivers are at time=T in order to efficiently assign the closest driver in the most optimal manner. Optimization and prediction business rules must be implemented. → Tracking Service will be used for this (The Traffic Server will also be used, if we do decide that we want it for version 1.)
Requirement 2: In an event that passenger cancels the reservation, we will need to re-assign the driver w/o disrupting his remaining ride schedule.

Comment: The usability here should be even more user friendly → "one-click away" (as opposed to two steps, first to open the app, second to select ride, third to select ride now. That is 3 clicks to request a pickup for Find Now, plus extra steps to confirm the order, so 4 steps in total!)

Comment: The service will interface with/into the Location Server

Comment: Location Determination module...

Comment: Make this part of the flows?

Comment: What is the reason for distinguishing b/w passenger vs driver in this way?
Can't we tell by type of cell phone (position determination – we can always assume that drivers will have a A-GPS position)? No, the same cell phone type can overlap b/w both user groups.

[illegible]		
5.	System returns the closest vehicle to the user in the form of distance (and time?)	Driver/Vehicle Tracking Service (Location Server) →Utility Service & Map Service & Directory (POI) Service → Passenger Location Service (Map Viewer)
6.	User continues with the order	Billing/Payment Calculation
7.	System assigns a driver for pickup	Service / Availability Management (WORK OUT A SEPARATE USE CASE FOR DRIVER GETTING NOTIFIED ABOUT A PICKUP)
8.	System displays the location of the driver on a map	Passenger Location Service (Map Viewer)
9.	User views the location of the driver on a map on their mobile device. Optionally, users can also view the distance info and estimated arrival time as text (for non-map phones text info about the driver's whereabouts.	
10.	System sends a SMS message	
11.		
Flow of Events – alternative path		
2.1	User is not authenticated and needs to: 1. login again 2. rest login info 3. etc.	
6.1	User does not continue with the order and cancels out of the Celluride application	
9.1	User optionally can send a SMS to driver	Can this be managed within the J2ME environment?
9.2	User optionally can call the driver	Voice Service
Postcondition:		

Comment: Time can be calculated using the Tracking Service

Comment: Map Service accesses basic map layers used in compiling maps for presentation. OGC's WMS can be used for this.

Comment: Map Service will only be used on cell phones that allow map display. Otherwise, text only (Directory (POI) Service can be used with text only)

Comment: POI's placenames should always be displayed along with the address

Comment: Both Map Viewer and Passenger Location Service are presentation services.

Comment: The Driver/Vehicle Tracking Service will return an XY coordinate.

In order to translate the coordinates to an address, a geocoding service is needed, which is to be provided by the Utility Service.

In order to translate the coordinates to a symbolic name (e.g., place name), a gazetteer or POI directory is needed, which is to be provided by the Directory (POI) Service

Comment: Zingo's Mobile Virtual Private Network (MVPN) enables a voice service b/w the passenger and taxi driver.

FROM OPenLS

Use Case 2: Passenger requests ride for later at time=T

In this use case, the passenger will specify the pickup location. Options for pickup location include:

- Current location
    - Q: what if the passenger changes location but still wants to be picked up at the initial specified time?
- Recent stops
- Favorites
- Key in Address
- Airports
- Other: enter placename / POI
- Other: select/click location on map (advanced feature)
    - Q: Which mobile environment enables sending a POI xy or symbol ID to the Location Server?
- Other: from a calendar (work appointment schedules) on passenger's mobile device

Location Refinement

LBS middleware can be instructed to refine the approximate device GPS/cell-id position to a location that more accurately represents the user's true location or represents a prominent landmark nearby the subscriber that they can readily identify.

For example, a list of probable locations within a region returned by the GPS/cell-id can be calculated and compared against the following features, POIs and locations:

- The locations a user has book-marked ("spatial bookmark")
- The subscriber's history (history is derived from the start and end points of previously calculated routes or manually assigned by the application)
- The nearest links on the underlying street network

Using a configurable rule-base, an inference engine within the LBS middleware can easily be customized to allow for various custom location refining calculations and interactions.

Directory Service

Comment: FROM OpenLS (ADD THIS TO THE ARCH DOC, IT HAS A SENTENCE ON DIRECTORY SERVICE)

This service provides subscribers with access to an online directory to find the nearest or a specific place, product or service. Through a suitably equipped OpenLS application, the subscriber starts to formulate the search parameters in the service request, identifying the place, product or service that they seek by entering the name, type, category, keyword, phone number, or some other 'user-friendly' identifier. A position must also be employed in the request when the subscriber is seeking the nearest place, product or service, or if they desire a place, product or service at a specific location or within a specific area. The position may be the current Mobile Terminal position, as determined through the Gateway Service, or a remote position determined in some other manner. The directory type may also be specified (e.g. yellow pages, restaurant guide, etc). Given the formulated request, the Directory Service searches the appropriate online directory to fulfill the request, finding the nearest or specific place, product or service, depending on the search criteria.

The service returns one or more responses to the query (with locations and complete descriptions of the place, product, or service, depending upon directory content), where the responses are in ranked order based upon the search criteria.

4.1.1.2 Find Ride Later

Use Case Description		
Name	Passenger requests ride for later at time=T	
Priority		
Description		
Precondition		
Flow of Events – Basic Path		Service (Component)
1.	User connects to the Celluride application	CDI
2.	System authenticates the user	(Authentication done by?)
3.	User selects to request ride later in the Celluride application and specifies the time interval T	

Use Case Description		
4.	Carrier proxy determines that this packet belongs to Celluride and forwards it to Celluride's server.	
5.		
6.		
7.		
8.		
9.		
10.		
11.		
Flow of Events – alternative path		
2.1		
6.1		
Postcondition		

Comment: These steps should resemble the steps for the driver (for Find Ride Now or Later?)

Outstanding Issues:

1. We need to determine the reservation business logic in the DIP:
   a. Do we assign a driver for this ride at this time for the specified time? KK: I don't think this is the best approach if we are talking about real-time optimization here. The Tracking Service will figure out which driver to book 30 min prior to the scheduled pickup.
   b. Do we simply store the reservation request in DIP and assign available nearest driver 30 min prior to the specified time? KK: This would seem right.
2. If we go with Option A, then 1) we must know exact schedule for the drivers time and distance where he/she is going to be prior to the ride.; 2) We need to build optimization rule so that no driver is sitting idle before the next pickup 3) if the passenger cancels the reservation, then we have to re-optimize the driver schedule. KK: True, which means that we will need to deploy the Tracking Service and Traffic Service and Route Planning Service for version 1.

Comment: DIP won't really know where the driver will be around the time of pickup…driver might be across town and traffic might cause delay to passenger's location

Comment: We still need a mechanism to make sure a ride will be available for the reservation and that there is no bottleneck in the system that will cause the re-routing, etc delaying the ride..

Option 1: Reserve driver at T=0 (book the driver at the time of reservation)
Reasons why we should go with Option 1:

- law firm prefers to put drivers on schedule rather than in a queue
- drivers want more power over their schedule (drivers own the vehicles)

Comment: Kevin favors this one.

- drivers will indicate at T=0 whether they are available
    - Q: does the system query driver's location and availability every time there is a reservation? Or is this info stored and refreshed every 30min, 1hr, etc?
    -

-

Option 2: Store reservation in DIP and assign driver ~40min prior to pick up reservation time (Reserve driver for T=x)
Reasons why we should go with Option 2:

-

Comment: John favors this one

4.1.1.3 View Reservations

Use Case Description		
Name	View reservation	
Priority		
Description	Passenger views reservation on his cell phone	
Precondition		
Flow of Events – Basic Path		Service (Component)
1.	User connects to the Celluride application	CDI
2.	System authenticates the user	(Authentication done by?)
3.	User selects to view his registration	
4.		
5.		
6.		
7.		
8.		
9.		
10.		
11.		
Flow of Events – alternative path		
Postcondition		

Use Case Description

4.1.1.4 Paperless Voucher System Use Cases

4.1.1.4.1 ?

4.1.1.4.2 ?

4.1.2 Internet Use Cases

Use cases (Internet):

1. Find Now
2. Find Later
3. View Reservations
4. View Complete Billing History

4.1.2.1 Find Now

4.1.2.2 Find Later

4.1.2.3 View Reservation

4.1.2.4 View Complete Billing History

4.2 Driver Viewpoint Use Cases

User = Driver

NOTES:

- Kevin wants to give drivers more power over their schedules (since in NYC, the drivers own the vehicles)
- law firm prefers to put drivers on schedule (instead of putting them to a queue)
-

Outstanding Issues:

1. Driver presence management:
   a. Available?
   b. Not Available?
2. Should the driver presence setting be managed by driver or automatic
3. How does the driver notify DIP when the ride has been completed?

KK: Car has reached the end point drop-off location. LBS middleware can be used for analyzing mobile-to static and mobile-to-mobile proximity relationships and responding to their events. This way, passengers using the Celluride cell phone application will be able, for example, to receive notification when a driver approaches their particular location.

LBS middleware provides the following functionality (out-of-the-box):

- Provide services based on a user location[3]
- Invoke fetching position as needed to update users' location information
- Generate queries for a user's current location, with or without heuristic refinement
- Generate proximity queries for pairs of mobile objects
- Maintain a cached list of mobile objects to monitor
- Perform periodic updates of one or many cached mobile objects to monitor
- Receive notification based on an event generated by a mobile object and its proximity to a static or other mobile object

Because of the abovementioned functionality, it is possible to employ automatic presence management.

Precondition

Passenger clicks order…DIP sends info to driver.

[3] LBS middleware contain geocoding services enabling applications to return a match confidence level for each geocode.

Use cases (Internet):

1. View Complete Billing History
2. View Entire Ride History

4.2.1 *Wireless Use Cases*

Use cases (Wireless):

1. Find Now
2. Find Later
3. RSS (Reservation & Scheduling System)
4. Personalized Driver Schedule
5. Paperless Voucher System
6. Mapping & Routing (turn-by-turn navigation)
    a. Q: Why is this as a separate use case? This is a functionality that will be part of Find Now and Find Later.

4.2.1.1 *Find Now*

Use Case Description			
Name	Find Now		
Priority			
Description	User (driver) requests passengers that need to be picked up at T=0, T=60, or T=90		
Pre-condition:		Post-condition:	
Flow of Events – Basic Path		Process Flow #	Service (Component)
1.	User logs into the Celluride application, entering his phone # and PIN	(From John's Screen Flows PPT)	CDI
2.	System authenticates the user		(Authentication done by?) (this is done in step 4?)

Comment: JOHN TO DO: MARK THE SCREEN FLOWS WITH NUMBERS.

Use Case Description		
3.	System sets drivers preference to 'Available' upon login and queries driver's location and sends the availability and location (done here in a later step, Step 4?) info to driver's carrier proxy server or gateway. ?and stores driver's latest location in the Celluride db	… A-GPS position is queried from driver's cell phone. The Location Determination Engine is called to retrieve the driver's location. Celluride db
4.	Carrier proxy determines that this packet belongs to Celluride and forwards it to Celluride's server. The server determines this message is from a driver. This packet will first be processed by Celluride's Subscriber & Service Engine (SSE), which authenticates the user	… SSE / Authentication
5.	User selects to View Schedule to see the pickup schedule for the first three T intervals (T=0, T=90, T=180). This sends a request to the system to retrieve rides for the first three T intervals.	
6.	System queries the reservation db for scheduled pickups and retrieves the first three T intervals for pickups. - which reservations are assigned to a particular driver? Rides within T=60 are assigned first. - How are reservations assigned? Based on vicinity, time, traffic conditions (time of day), etc. ? The information is pushed back to the user. The carrier forwards the above message to the user.	Reservation DB = ?
7.	User selects Accept for the ride for time interval T [1], and for T = 2 and T = 3…, which sends out a message with this information to the system… ADD FOR SYSTEM?: System (Carrier proxy) determines this message belongs to Celluride and forwards it (to the Celluride server). ADD FOR SYSTEM? The system (Celluride server) invokes SSE to log user's transaction info and saves any necessary billing info in the Cellruide DB.	Carrier proxy determines this message belongs to Celluride and forwards it.

Comment: 4 options for location determination:
1. GPS
2. LIF/OMA MLP
3. BREW
4. Java JSR 179

Comment:
Do we really want this this way? Or will the driver have the power to tell the system when he is available.

I see…rides are scheduled for the first three T intervals (i.e., made the day prior)…but rides can also be reserved for later in the day, so will these be displayed too? Oh ok, it is always the first three T intervals.

Comment: Here or in a later step? Is the request sent now to query for pickups (no, later when driver selects View Schedule)

Comment: Q: Is this re-saved with an updated location after each pickup or…?

Comment: Subscriber and Service Engine is now called == ???

Comment: done in this step or in step 2?

Comment: New name for this is ______, part of the Celluride Control Center

Comment: is this already done in step 2?)

Comment: The Driver will always be shown (only?) the first three T intervals, regardless the time of day.

Comment: For Find (Passenger) Now we really just want T=0, no?

Comment: Q: Why are the T intervals 90min each? What if the first ride will take only 30min. Then the following 60min is wasted? Where's the optimization here?!

Comment: The three T intervals are kept in memory on driver's mobile so that when he views each interval, a separate request to the system is not needed. HOWEVER, THIS MIGHT CAUSE CONFUSIOIN: Q: Is there room for issues/error here? For example, the driver should query the system after he is done with the first ride because a cancellation/rescheduling might have occurred. ... [7]

Comment: Q: Why do drivers need to accept (or decline) if they are on a pre-assigned pickup schedule?

Comment: Doe the user selects Accept for T=0, T=90, T=180 separately? (yes)

Comment: Q: Will this be done for both passenger and driver?

[illegible]		
8.	System…. And sets drivers preferences to Not Available ADD FOR SYSTEM?: System sends a confirmation message to the passenger. ADD FOR SYSTEM?: System logs the transaction info for billing purposes (and then sends the routing info and maps showing the driver how to get to the passenger's location)	
9.	User selects to View Passenger Location	
10.	System fetches a map and displays it on the mobile device [2]	
11.		
12.		
Flow of Events – alternative path		
5.1	User selects to Set Preferences	
5.2	User selects to Log Out	
7.1	User selects Reject the ride for time interval T [1]	
9.1	User selects to Log Out	
9.2	System logs out user and clears …	
Postcondition:		

[1] DIP does not allow driver to reject rides that are within 1hr to pickup

[2] Map is refreshed every 2 minutes with current user's position. DIP will provide a Tracking Service starting in version 1 for continuous tracking.

Comment: Q: Should this be done automatically for T=1?
Also, for T=2 when T=1 is over?

Repeat for when T=3 is T=1

Q: Will additional Ts be assigned when T3 is T1?

Comment: So version 1 will need a Tracking Service in order to be able to refresh the map every 2 minutes with current user positions

4.2.1.2 Find Later

ISSUES/Q here is that drivers will have their pickup schedule assembled for the first T intervals. This will be assembled by the fleet company / Cellruide system automatically …using info such as previous' day booking info, location of passengers and drivers, etc.

--

OpenLS
The OpenLS thread in OWS-3 created a new **Tracking Service** described in OGC document 2006-026R0.
This Tracking Service has three main features:
1) A standard interface to a service that stores and retrieves tracking data.
2) A concise encoding scheme for efficient communication of location data over a low-bandwidth

Comment: SEE OGC document 2006-026R0.

network such as provided by current "2.5G" CDMA or GPRS providers.
3) A standard for efficient storage and retrieval of location data at the server.

OWS-3 addressed only the first item in the above list. The remaining two items were considered in OWS-3 and some design and implementation was completed. Issues remain regarding the necessity, relevance, and commercial value of extension to these items. One of the "refinements" to the Tracking Service addressed in OWS-4 will be a better resolution of those issues and the inclusion or exclusion of various aspects of these items.

The practical application of the Tracking Service is to support near real-time tracking of vehicles and personnel equipped with suitable tracking devices and CDMA or GPRS data networks. "Near real time tracking" is meant to provide for an active display of accurate location information of multiple (up to 100) tracked assets with minimal latency. A use case is an E911 emergency dispatch center geographic display of emergency responder personnel and vehicle locations. There will be some latency in the display system itself and in the database retrieval of the location information to be displayed, but the OLS tracking standard should not introduce latency greater than 5 seconds, including network delays, on current commodity-priced tracking devices such as GPS-equipped smart phones. Ideally, most of the latency should appear as network delays with minimal latency introduced by the CPU cycles required to generate an encoded location on a device and decode it for storage at the server. As true 3G wireless networks appear, the latency should diminish to less than 1000 milliseconds, not including acquisition of GPS data.
The tracking service should support position reports as frequently as once per second for fast moving assets.

The Tracking Service should support a base level of operation just sufficient to support a simple "moving dots on a base map" tracking application. In addition there may additional optional capabilities as follows:
1) The means to specify when (time period) the device or mobile device should be tracked.
2) The means to specify the criteria (temporal & spatial) that should be used to generate a

notification or alert. This includes periodic updates and notifications when a mobile device enters or exits a region, starts, stops, moves a specific distance or when a mobile device interacts with another mobile device.
3) A well-defined mechanism and message format used to send a notification or alert to the client application.
4) The means to access position/time/speed/direction information for one or more mobile terminals/assets through the Gateway Service.
5) The means to store position/time/speed/direction information for one or more mobile terminals/assets for a designated time period and sampling periodicity.
6) The means to fetch position/time/speed/direction information tracks for one or more mobile terminals/assets for a designated time period.

The OpenLS enhancements should be harmonized with JSR 179 and the Parlay X API where reasonable and possible. For example, if the device reports a position due to a geofence violation, the location reporting protocol should include an indication of which fence was violated, along with the location report.
There should be a way to store the fence geometry on the server, and download the fence geometries from the server.

4.2.1.3 RSS (Reservation & Scheduling System) Use Cases

4.2.1.3.1 View Reservation

4.2.1.3.2 Schedule a Pickup

- Scheduling will be done automatically by the System (or Fleet Admin). The moment driver logs in, the system marks him as available for scheduling. If there were no prior reservations, system assigns a reservation closets pickup. If no reservations were made for the zone(s) the driver is located in, the driver will wait till a reservation in his zone comes in.
    - What if: there was a reservation made for a later time in a far away zone for a driver? Reshuffle the reservations so

that it makes sense where each driver drives to (factors: based on proximity, car preference, etc.)

-

4.2.1.4 Personalized Driver Schedule Use Cases

Comment: LBS middleware offer Personalization engine. A good feature to have, as the system will automatically identify the request's owners preferences based on previously configured preferences, and hence speed up querying of databases and the displaying of results.

4.2.1.4.1 Set Schedule Preferences

- Driver views his schedule → BUT, this is the same as the above use case under RSS
-

4.2.1.5 Paperless Voucher System Use Cases

4.2.1.5.1 Use Case 1…

4.2.1.6 Mapping & Routing Use Cases

Comment: Should not be a separate use cases. This is a functionality within the other use cases

General use cases can involve:

- *setting routing preferences*
-

5 Portal Use Cases

Overview: Customers and passengers will be interacting through a portal (CCP) to request rides, payments and setting preferences, etc. Fleet companies will be interacting through a CFP to set the availability options and pricing. They will also get information regarding the payments received.

5.1 Corporate Administrator Viewpoint Use Cases (Internet Only)

5.1.1 PTM (Private Transportation Management Portal) Use Cases

5.1.1.1 Employee Registration

5.1.1.2 Set Preferences & Provisioning

5.1.1.3 Billing History

5.1.1.4 Reporting & Analysis

5.1.2 Paperless Voucher System Use Cases

5.1.2.1 Use Case 1...

5.2 Fleet (Dispatcher) Viewpoint Use Cases

5.2.1 Dispatch & Reservation Portal Use Cases

5.2.1.1 Complete Ride history

5.2.1.2 Billing & A/R

5.2.1.3 Transaction History

6 In-Route Use Cases (version 2+)

- Find best route (route planning)
- Find closest POI along the route
- Time-dependent routing
-

Comment: This can go under Mapping & Routing Use Cases above ...

The above functionalities can be deployed out-of-the-box using an LBS middleware. This can be done for version 1, since this functionality is out-of-the-box.

Find best route
LBS middlewares come with Route Planning Services that can be used to calculate single and multi-leg routes and turn-by-turn travel directions. This way, the Celluride application can generate optimized routes using preferences to minimize the time or distance traveled for a given transportation mode. Calculations consider factors such as time required to complete turns and the side of the road a driver uses in different geographic regions.
Several different cost criteria are supported by LBS middlewares, including:

- Shortest distance
- Shortest time
- Avoid Tunnels or Bridges or Ferries
- Avoid Traffic

With Route Planning Services applications can:

- Calculate routes between two or more locations (single or multi-leg routing)
- Calculate routes or turn-by-turn directions that satisfy specific cost criteria and travel modes
- Calculate routes that are influenced by time-dependent road conditions such as turn restrictions or road closures
- Calculate a route that avoids certain link-based features such as bridges or tunnels
- Calculate a route that avoids real-time traffic incidents

Find closest POI
- Find closest features based on travel mode. For example, the closest ATM machine based on driving distance. Passengers may request to stop at an ATM on the way to their drop off location. Knowing the location of an ATM will allow the driver to deliver the passenger to both locations quickly and effectively. There will be other cases for stops

along the route, for instance, when something needs to be picked up on the way to the meeting drop-off location.

LBS middlewares provide event notification services. Applications can:

- Notify a subscriber when another mobile object such as another subscriber moves within a proximity area
- Notify a subscriber if he or she is within a certain proximity to a particular point of interest (POI) along their travel route
- Notify a business whether a potential subscriber is within walking distance of their storefront
- Support and manage multiple outbound messaging channels such as SMS, SMTP, HTTP agents, JMS, etc.

Time-dependent routing

This section describes how the Celluride platform can provide state-of-the-art reliable on-time pickup/dropoff service.

LBS middleware applications can generate time-dependent routes that account for time sensitive conditions such as turn restrictions, street closures and even real-time traffic events. This also allows for generating routes that are relevant for a specific time of day (i.e., dependent street condition data, etc). Moreover, time dependent routes can be generated or existing routes can be checked for their validity for a specific time.

Time-dependent link speed updates such as those caused by traffic events can be fed into the model through the existing LocationLogic dynamic content framework.

LBS middleware can also provide a means for identifying time dependent travel maneuvers within travel direction narratives.

Combining time-dependent routing with event alert/notification services will enable the Celluride application to send a notification to the driver only when a route exists that is more optimal than their regular route.

...

...

...

...

Page 27: [1] Comment K Kolodziej 10/12/2006 2:52 PM

In the UK -- over 6,000 registered companies (2004)

Teydo BV is a leading LBS provider that owns and operates the MobiSPOT™ Location Based Service aggregator platform with global reach for Positioning, SMS and billing. MobiSPOT™ provides partners like Microsoft and Trimble access to more then 100.000.000 GSM subscribers enabling them to position Mobile Phones on Vodafone, T-Mobile, O2, Orange, E-Plus, Telenor and many other GSM operator networks. The platform can be accessed through an open API for third party application integration. Teydo's internal LBS based service FleetOnline™ helps fleet owners worldwide control their fleets and reduce operational costs while increasing customer loyalty and satisfaction. FleetOnline™ is available for everyone who has an internet connection and a GSM subscription at no additional cost or investment. Today FleetOnline™ has over 9.000 subscribers tracking over 100.000 mobile phones and Trimtrac™ devices on a daily basis

Page 27: [2] Comment K Kolodziej 10/12/2006 2:52 PM

Teydo LocationXS "web service" (website)

This is used by Teydo's partners serving the LBS. Location XS is built on Teydo's FleetOnline web service and Trimble's trimTrac locator, but branded as LocationXS. Partners sell a TrimTrac locator to an end user and open an account for the end user on the LocationXS web site. All TrimTrac units require an annual subscription for the wireless service for data communication and LocationXS access.

Page 27: [3] Comment K Kolodziej 10/12/2006 2:52 PM

rovide delivery drivers with point-to-point directions and maps for all of their routes and reduces the response time and delivery costs for special deliveries by dispatching drivers closest in proximity for Just-In-Time customer service

Page 27: [4] Comment K Kolodziej 10/12/2006 2:52 PM

To use MLS in commercial settings, a list of phone numbers that a particular MLS installation is authorized to locate needs to be specified (under a contract) with the carrier.

Q: this is true for all LBS middlewares? Yes, ArcWeb also locates Sprint and Bell Mobility (they have open location APIs)

Page 36: [5] Comment K Kolodziej 10/12/2006 2:52 PM

Subject	Another drivetime problem
Author	Graham Stewart
Date	Apr 19, 2006
Message	When I create both a 9 and a 10 minute drivetime around POINT(-85.5825 43.076) I observe the following things: 1) That the 9 minute drivetime is not contained within the 10 minute drivetime 2) That the 9 minute drivetime is larger than the 10 minute drivetime 3) That the 9 minute drivetime is considerably more complex than the 10

minute drivetime.
4) I have to go all the way to a 15 minute drivetime to get one that wholly includes the 9 minute one.

This seems counter-intuitive. Is there a good reason why this is the case?

thanks
Graham

- Top Print Reply Bookmark Message View Bookmarks

Subject Re: Another drivetime problem

Author ArcWeb Services Team

Date Apr 25, 2006

Message Graham,

We use less detailed layers to calculate drive times 10 minutes and upwards keeping in mind performance considerations.
The layers for drivetimes 9 minutes and lesser are more detailed and in this case result in a polygon larger than those generated by less detailed layers.

thanks,
Aju
ArcWEB Services

Page 54: [6] Comment **K Kolodziej** **10/12/2006 2:55 PM**

Additional security measures, such as secure connections and cryptography, are definitely needed, especially for those applications transmitting sensitive data. However, the adoption of HTTPS/SSL for secure connections or cryptographic standards, such as IPSEC and WTLS, brings with it the associated issues

What about XML messaging?

Security Assertion Markup Language (SAML), which is a protocol to transport authentication and authorization information in an XML message. It could be used to provide single sign-on Web services.

XML digital signatures define how to digitally sign part or all of an XML document to guarantee data integrity. The public key distributed with XML digital signatures can be wrapped in XML Key Management Specification (XKMS) formats.

XML encryption allows applications to encrypt part or all of an XML document using references to pre-agreed symmetric keys.

The Web services secure XML protocol family (WS-Security), endorsed by IBM and Microsoft, is a complete solution to provide security to Web services. It is based on XML

digital signatures, XML encryption, and an authentication and authorization scheme similar to SAML
All of the above security protocols can bind to Web services messaging protocols. For example, we can embed a SAML segment in a SOAP message header to authenticate and authorize the access to the requested services. We can also embed an XML Digital Signature segment in a SOAP header to authenticate a credit card number in that message.

Page 74: [7] Comment **K Kolodziej** **10/12/2006 2:55 PM**

The three T intervals are kept in memory on driver's mobile so that when he views each interval, a separate request to the system is not needed. HOWEVER, THIS MIGHT CAUSE CONFUSIOIN: Q: Is there room for issues/error here? For example, the driver should query the system after he is done with the first ride because a cancellation/rescheduling might have occurred.

Celluride Product Requirements Document (PRD)

October 19, 2006

Version 1.0

LEGAL NOTICE

The contents of this document constitute valuable proprietary and confidential property of Celluride Wireless and are provided subject to specific obligations of confidentiality set forth in one or more binding legal agreements. Any use of this material is limited strictly to the uses specifically authorized in the applicable agreement(s) pursuant to which such material has been furnished. Any use or disclosure of all or any part of this material not specifically authorized in writing by Celluride Wireless is strictly prohibited.

Document Information

Document title:	Product Requirements Document
Document file name:	PRD_Celluride_101905
Revision number:	1.0
Issued by:	John Yu
Issue Date:	October 19, 2005
Status	Draft

Revision History

Revision	Date	Editor	Description of change
1.0	19-Oct-2005	John Yu	Initial Release

Table of Contents

COMMENTS AND QUESTIONS

For comments or questions related to this document, the concept(s) presented, or scope of the project, please contact:

1. INTRODUCTION

1.1 Market Overview

In their 2003 report, Gartner Dataquest estimates that there are estimated 60 million mobile workers in the United States. As the U.S. workforce is becoming increasingly mobile, the demand for air and ground transportation will increase.

The ground transportation industry is built around a central dispatcher who relays schedules, pickup times, location updates, and service requirements between a customer and driver. Often, customer requirements are not fulfilled due to a number of reasons including, unclear requirements between the customer and the dispatcher, exchange of inaccurate requirements between the driver and dispatcher, overburdened dispatchers, inability for the driver to find the pickup location and/or late arrival, and other human errors. All of these factors contribute to the poor customer service. Therefore, finding reliable, timely and high quality ground transportation can be a laborious, unproductive and often, frustrating experience for the business traveler.

Adding to the already inefficient system is the paper-based voucher system. The fleets invoices the corporations using the voucher that the passenger used to paid for his/her ride. The paper-based voucher system consists of manual process and therefore it is not only highly inefficient but also contains potential room for frauds.

1.2 Business Objective

Celluride offers a complete end-to-end ground transportation system specifically designed for the black car and limousine industry. Celluride has following value propositions:

- The Celluride Direct Interaction Platform (DIP) allows passengers to interact directly with the drivers and share information regarding location awareness, estimated time of arrival, and any changes to the itinerary.
- DIP allows passenger to instantly locate the closest driver in any U.S. city from his/her cell phone.
- DIP will optimize the passenger to the fleets matching process based on availability, passenger preferences, budget, and quality of service.
- DIP will optimize information sharing and matching while reducing or eliminating human time dependency for real-time operations.
- Allows awaiting passenger to view the location of the driver on a map rendered on his phone.
- Provides automated paperless voucher processing by using GPS and location and time stamping mechanism.
- Allows the fleets to track its drivers that are on the road with GPS system.

2. BUSINESS REQUIREMENTS

2.1 Key Actors

Following are the key actors that are associated with the Celluride ecosystem.

- ***Corporate Users*** – Celluride will target businesses whose need is to effectively manage the employees' ground transportation system. The Corporate Users is broadly defined term, and it can mean the corporations or it can refers to the individuals within a company who is responsible for managing employee's transportation needs. Typically these individuals are the Transportation Managers and Administrative Directors.

- ***Passengers*** – The passengers are the end users of ground transportations. They are the mobile professionals and the employees of corporations.
- ***Fleet Operators*** – Fleets Operators are the companies that own and manage the ground transportation vehicles.
- ***Drivers*** – The one who provide transportation service to the passengers. The drivers can either be the employees of the Fleet Operators or independent vehicle owners that are contracted by the Fleets Operators
- ***Celluride System Operators*** – Celluride employees who operate, manage, and support the Celluride ecosystem.

2.2 Corporate User Needs

As mentioned corporate users is Celluride's primary target segment. Corporations are always chasing costs and pursuing ways of running their business more efficiently and improve the bottom line. Corporations want to have control and will demand efficiency in the way they handle employee travel and transportation needs. Specifically, when it comes to managing employee transportations following can be said about their needs:

- Easy management of employee ground transportation needs
- Set rates and vehicle requirements for different groups of employees
- Manage billing
- Ability to find cost effective fleet
- View visibility into ride history of employees
- Validate ride transaction
- Reporting and analysis

2.3 Passenger Needs

Today's mobile professionals are increasingly demanding. They are not only more mobile and busy, but also they have complex itinerary. They require instant service when it comes to ground transportations. In particular, they require awareness of the location of the driver while waiting to be picked up. Additionally, they desire speed and are willing to pay extra to instantly locate available vehicle that meet their requirements.

2.4 Fleet Operator Needs

The fleet operators are looking for ways to increase business and manage their mobile drivers. They need to respond to customers requirements effectively and efficiently. Poor customer service results in loss and discontinued service. One key way to attract retain customer is to offer differentiated in areas of efficiency, customer service, and operational excellence.

2.5 Driver Needs

In general, drivers of taxis, black car's, limo's and para transit vehicles have passenger generation problems that increase the cost of their service. Therefore, drivers are looking for ways to service more passengers and reduce or eliminate idle time.

3. REFERENCE INFORMATION

3.1 References

The following table lists the references used in the development of this document.

Table 1: Document References

Ref. ID	Document Title	Rev.	Company/ Author	Published Date
[1]	Celluride Wireless Executive Summary		Kevin Halpern	July 1, 2005
[2]	Celluride Business Plan		Kevin Halpern	Spring 2004
[3]	Celluride Control Center Technical Architecture Document	0.2	Taspa Alagarsamy	October 14, 2005
[4]				
[5]				

3.2 Assumptions

The following assumptions are made in the creation of this document and the assessment of the messaging. If any of these assumptions are determined not to be true, impacts to the document requirements will need to be assessed.

Table 2: Assumptions

ID Num.	Assumption	Validated (Y/N)
1.		
2.		
3.		
4.		
5.		
6.		
7.		
8.		
9.		
10.		
11.		
12.		
13.		

3.3 Constraints

The following constraints are understood in the creation of this document and the assessment of the messaging. If any of these constraints are determined not to be true, this assessment will have to be revisited.

Table 3: Known Constraints

ID Num.	Constraints	Validated (Y/N)
1.		
2.		
3.		
4.		
5.		
6.		

3.4 Acronyms and Terms

3.4.1 Acronyms

This section lists the acronyms and the terms that comprise the acronyms. The convention used throughout this document is that the first use of the acronym is accompanied by the terms that comprise the acronym. Subsequent instances will just use the acronym by itself.

AGPS	Assisted GPS
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BREW	Binary Runtime Environment for Wireless
CC&B	Customer Care and Billing
DB	Database
GPS	Global Positioning System
http	hyper text transport protocol
https	http Secure
GIS	Geospatial Information System
ID	Identifier
iDEN	Integrated Digital Enhanced Network
IP	Internet Protocol'
J2ME	Java 2 Platform, Micro Edition
JDBC	Java Database Connectivity
lat	Latitude
LBS	Location-Based Services
LES	Location Enhanced Services
LIF	Location Interoperability Forum
long	Longitude
MLP	Mobile Location Protocol
MS	Mobile Station
MT	Mobile Terminated
NDA	Non Disclosure Agreement
OAP	Over-the-Air Provisioning
OMA	Open Mobile Alliance
PPG	Push Proxy Gateway
QoS	Quality of Service
SMS	Short Messaging Service
TBD	To Be Determined
TCP	Transport Control Protocol
URL	Universal Record Locator
WAP	Wireless Application Protocol
XML	eXtensible Markup Language

3.5 Requirements Designation

Each requirement has been given a designation. The designations and the interpretation of the designations are described in the following table.

Table 4: Modified IEEE Definitions of Software Requirements Designations

Designation	Definition
Mandatory[1]	The system will not be acceptable unless this requirement is met in an agreed manner.
Mandatory/Conditional	Mandatory, but dependent on other influences, such as the availability of other requirements and business drivers.
Conditional	This requirement would enhance the system; however, if this requirement is not met, the system will still be acceptable.
Optional	A class of functions that may or may not be worthwhile; gives development the opportunity to propose something that exceeds the minimum requirements.
Informational	The ensuing text is for information purposes only. This text is intended to define the scope of a requirement, used to provide context, or can be used to indicate a requirement that may already be supported.

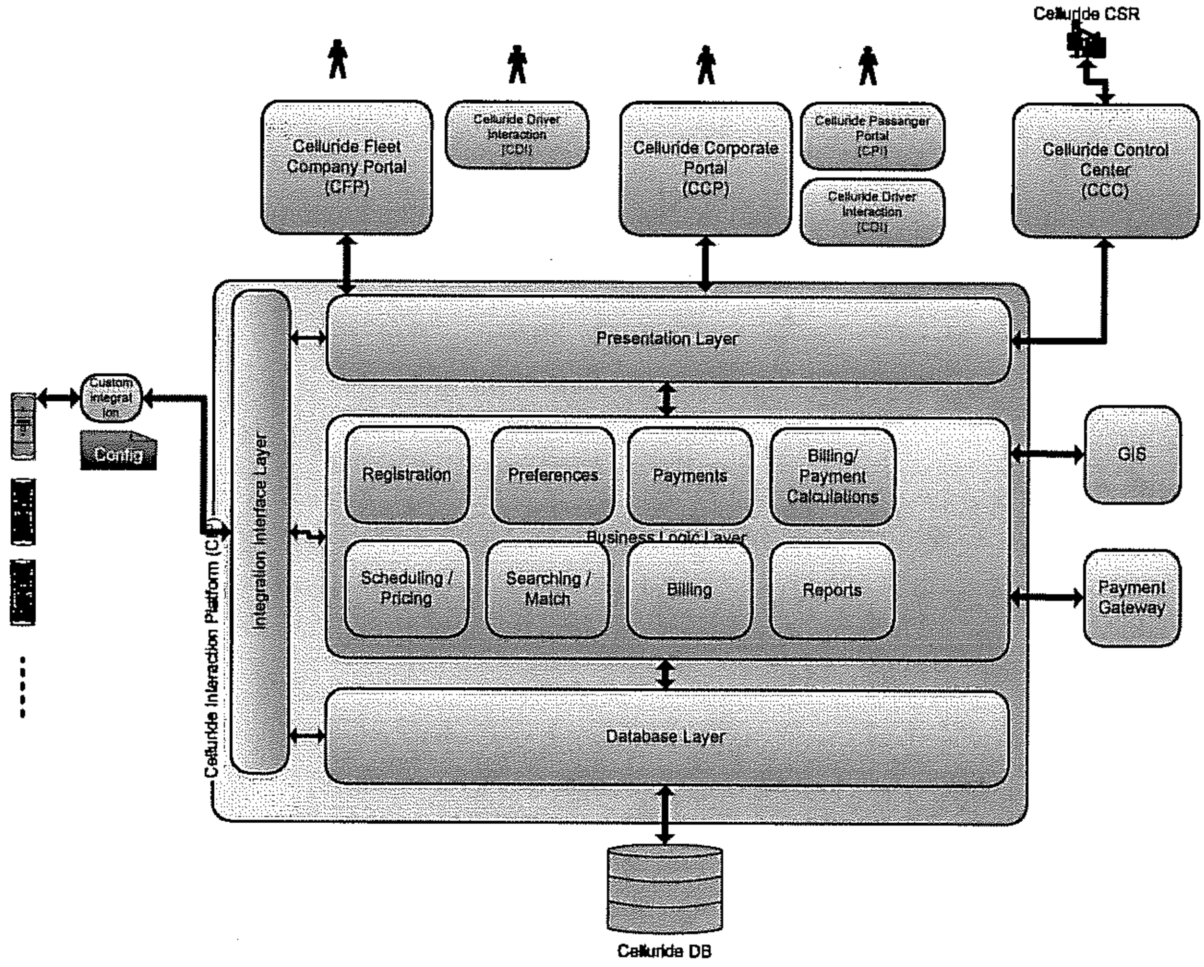
[1] IEEE uses the term "essential" instead of the term "mandatory."

4. CELLURIDE ECOSYSTEM

4.1 High-Level Architecture

The following figure depicts the functional elements that make up the Celluride ecosystem.

Figure 1: Celluride High-level Block Diagram


Celluride CSR
Celluride Fleet Company Portal (CFP)
Celluride Driver Interaction (CDI)
Celluride Corporate Portal (CCP)
Celluride Passanger Portal (CPI)
Celluride Driver Interaction (CDI)
Celluride Control Center (CCC)
Presentation Layer
Custom integrat ion
Config
Integration Interface Layer
Celluride Interaction Platform (CI)
Registration
Preferences
Payments
Billing/ Payment Calculations
Business Logic Layer
Scheduling / Pricing
Searching / Match
Billing
Reports
GIS
Payment Gateway
Database Layer
Celluride DB

Celluride - Highlevel Block Diagram (Level I)

4.2 Functional Components

The following table describes the functional components depicted in **Figure 1**.

Table 5: Celluride Functional Component Description

Component	Component Description
Celluride Direct Interaction Platform (DIP)	
Fleet Company Portal (FCP)	
Celluride Driver Interaction (CDI)	
Private Transportation Management (PTM) Portal	
Celluride Passenger Interaction (CPI)	
Celluride Control Center (CCC)	
GIS	
Celluride Database	

4.3 Interfaces

The following table provides a high-level description of the interface depicted in **Figure 1**.

Table 6: Celluride Ecosystem Interface Description

Interface Number	Interface Description
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

5. FUNCTIONAL REQUIREMENTS

This section describes the requirements that need to be supported in the Celluride ecosystem. The requirements are categorized by the Key Actors defined in Section 2.1.

5.1 Corporate Users

This section describes the requirements pertaining to the corporate users. The corporate users are the primary customer target since they are considered to be the purchase decision makers of the Celluride system. Typically, within a Fortune 500 firm, there's a Corporate Travel Manager or an Administrative Director who oversee the travel and transportation needs of employees. These individuals will be the corporate users. As a part of Celluride offering we intend to develop a web-based portal specifically designed for use by the Transportation Manager. This management portal is called the Private Transportation Management (PTM) Portal.

5.1.1 Private Transportation Management (PTM) Portal

5.1.1.1 Overview

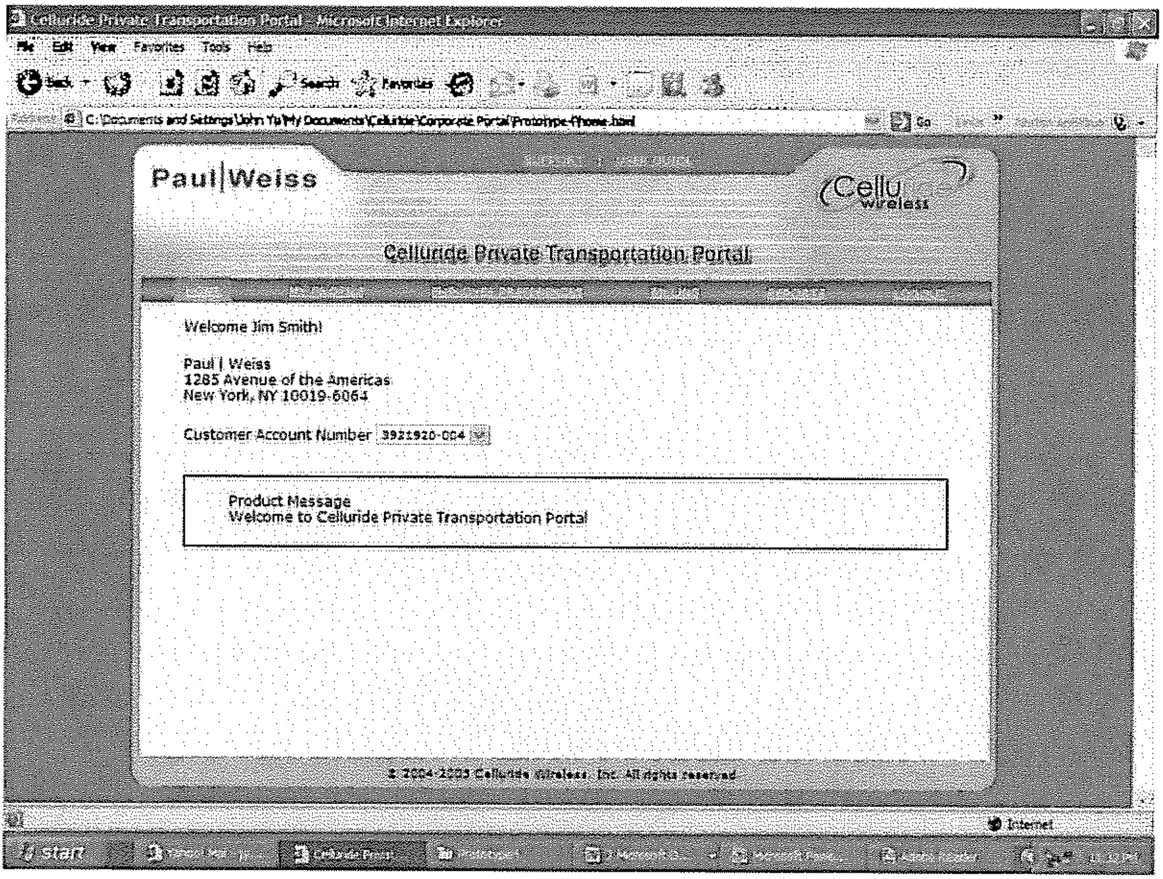
The Private Transportation Management (PTM) Portal is a web-based application that provides corporations to manage their employees' ground transportation needs. The PTM portal provides Transportation manager with a complete visibility of the ground transportation transactions made by the employees. It allows corporate travel managers to define and apply travel policies at the individual level. The PTM portal is hosted by Celluride, and is accessed from Celluride website.

5.1.1.2 Modules

The PTM portal consists of four main modules.

1. Administrator Profile
2. Employee Preferences
3. Billing
4. Reporting

5.1.1.3 ˙Home Page



Celluride Private Transportation Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
C:\Documents and Settings\John Ya\My Documents\Celluride\Corporate Portal\Prototype-Home.html
Go
Paul|Weiss
Cellu wireless
Internet
start

Celluride Private Transportation Portal

Welcome Jim Smith!

Paul | Weiss
1285 Avenue of the Americas
New York, NY 10019-6064

Customer Account Number 3921920-004

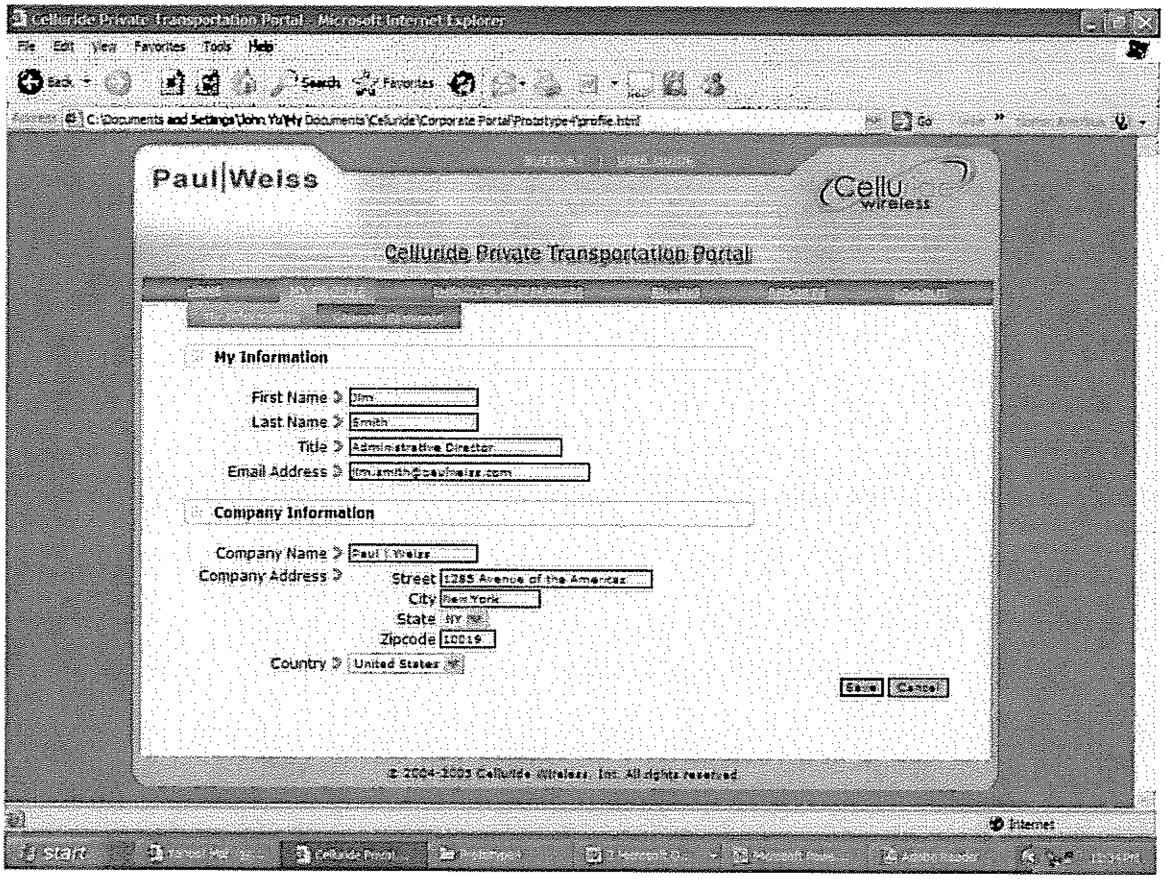
Product Message
Welcome to Celluride Private Transportation Portal

© 2004-2005 Celluride Wireless, Inc. All rights reserved

The home page is the first page the user views when he/she logs into the PTM portal. It is also called the "landing page." The home page requirements are as follows:

- Company logo shall be displayed at the top left corner page.
- Company address and account information shall be displayed
- There should be a section for displaying a global product message. The product message section shall be used to communicate to the users.

5.1.1.4 My Profile Page



Celluride Private Transportation Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
C:\Documents and Settings\John Yu\My Documents\Celluride\Corporate Portal\Prototype4\profile.html
Paul|Weiss
Cellu wireless

Celluride Private Transportation Portal

My Information

First Name > Jim
Last Name > Smith
Title > Administrative Director
Email Address > jim.smith@paulweiss.com

Company Information

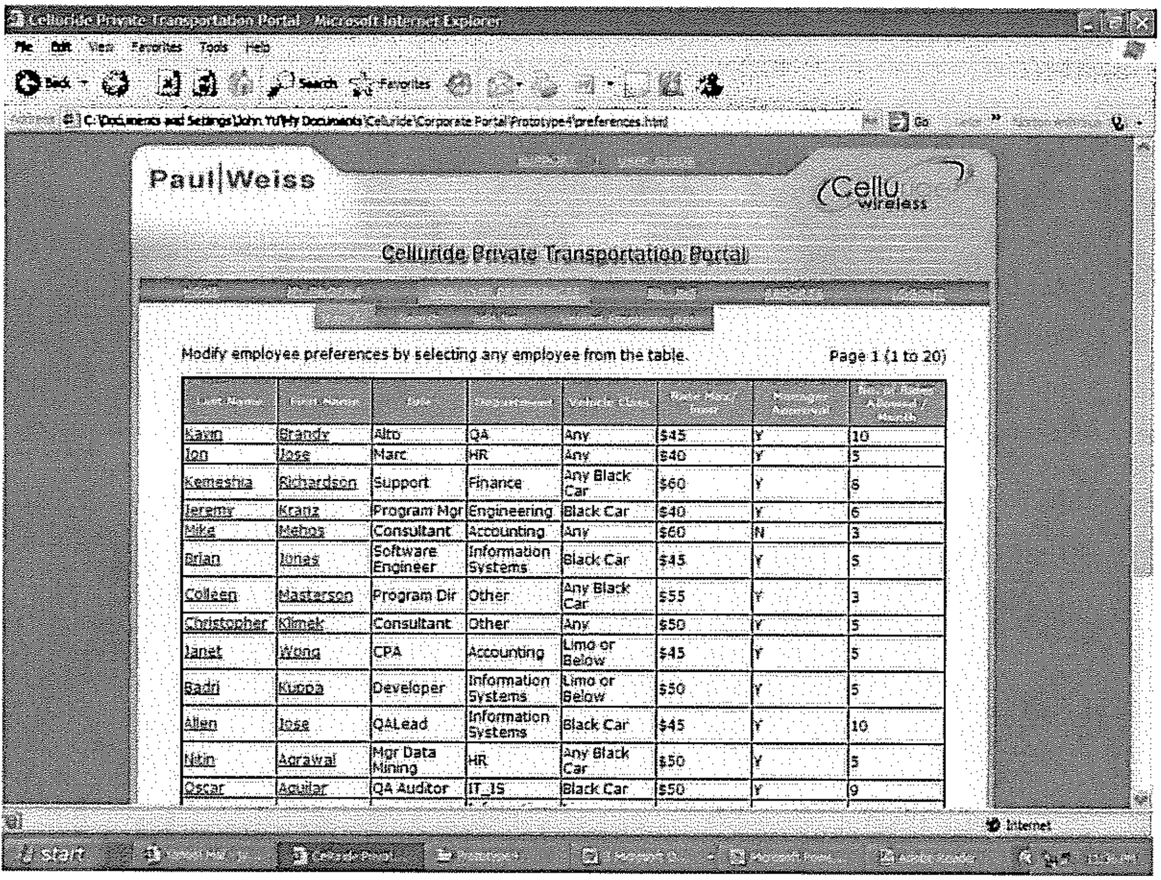
Company Name > Paul | Weiss
Company Address > Street 1285 Avenue of the Americas
City New York
State NY
Zipcode 10019
Country > United States

Save
Cancel
© 2004-2005 Celluride Wireless, Inc. All rights reserved.
Internet
start

My Profile page is used to manage user, account, and company information. Following are the requirements for My Profile page:

- The company administrator shall be able to manage following information from this page:
    - Name
    - Title
    - Email address
    - Company name
    - Company address
    - Login ID
    - Password

5.1.1.5 Employee Preferences Page


Celluride Private Transportation Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
C:\Documents and Settings\John.Yu\My Documents\Celluride\Corporate Portal\Prototype4\preferences.html
Go
Paul|Weiss
Cellu wireless
Celluride Private Transportation Portal
Internet
start

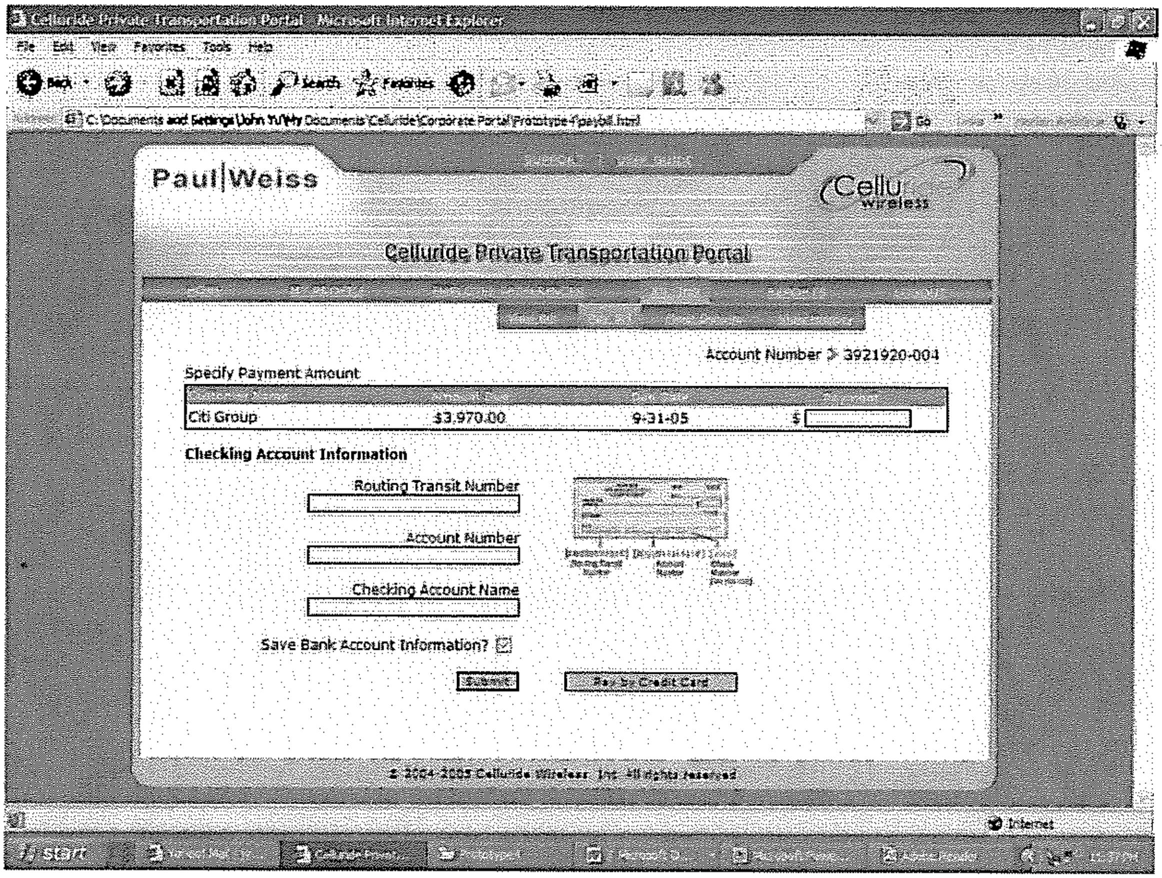
Modify employee preferences by selecting any employee from the table. Page 1 (1 to 20)

Last Name	First Name	Title	Department	Vehicle Class	Rate Max/ hour	Manager Approval	No. of Rides Allowed / Month
Kavin	Brandy	Alto	QA	Any	$45	Y	10
Jon	Jose	Marc	HR	Any	$40	Y	5
Kemeshia	Richardson	Support	Finance	Any Black Car	$60	Y	6
Jeremy	Kranz	Program Mgr	Engineering	Black Car	$40	Y	6
Mike	Mehos	Consultant	Accounting	Any	$60	N	3
Brian	Jones	Software Engineer	Information Systems	Black Car	$45	Y	5
Colleen	Masterson	Program Dir	Other	Any Black Car	$55	Y	3
Christopher	Klimek	Consultant	Other	Any	$50	Y	5
Janet	Wong	CPA	Accounting	Limo or Below	$45	Y	5
Badri	Kuppa	Developer	Information Systems	Limo or Below	$50	Y	5
Allen	Jose	QALead	Information Systems	Black Car	$45	Y	10
Nitin	Agrawal	Mgr Data Mining	HR	Any Black Car	$50	Y	5
Oscar	Aguilar	QA Auditor	IT_IS	Black Car	$50	Y	9

The Employee Preferences page is used to set employee preferences regarding the type of vehicle class and rates he/she is allowed use. Following features shall be supported from the Employee Preferences page:

- Ability to set employee preferences by: vehicle class, maximum hourly rate of rides, maximum number of rides allowed per month, and whether manager approval is required
- Ability to search employees
- Ability to add new employees
- Ability to upload the set of employee data

5.1.1.6 Billing Page


Celluride Private Transportation Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
C:\Documents and Settings\John Yu\My Documents\Celluride\Corporate Portal\Prototype-f\paybill.html
Go
Paul|Weiss
Cellu wireless
Celluride Private Transportation Portal
Submit
Pay by Credit Card
© 2004-2005 Celluride Wireless, Inc. All rights reserved
Internet
start

Account Number > 3921920-004

Specify Payment Amount

Citi Group	$3,970.00	9-31-05	$

Checking Account Information

Routing Transit Number

Account Number

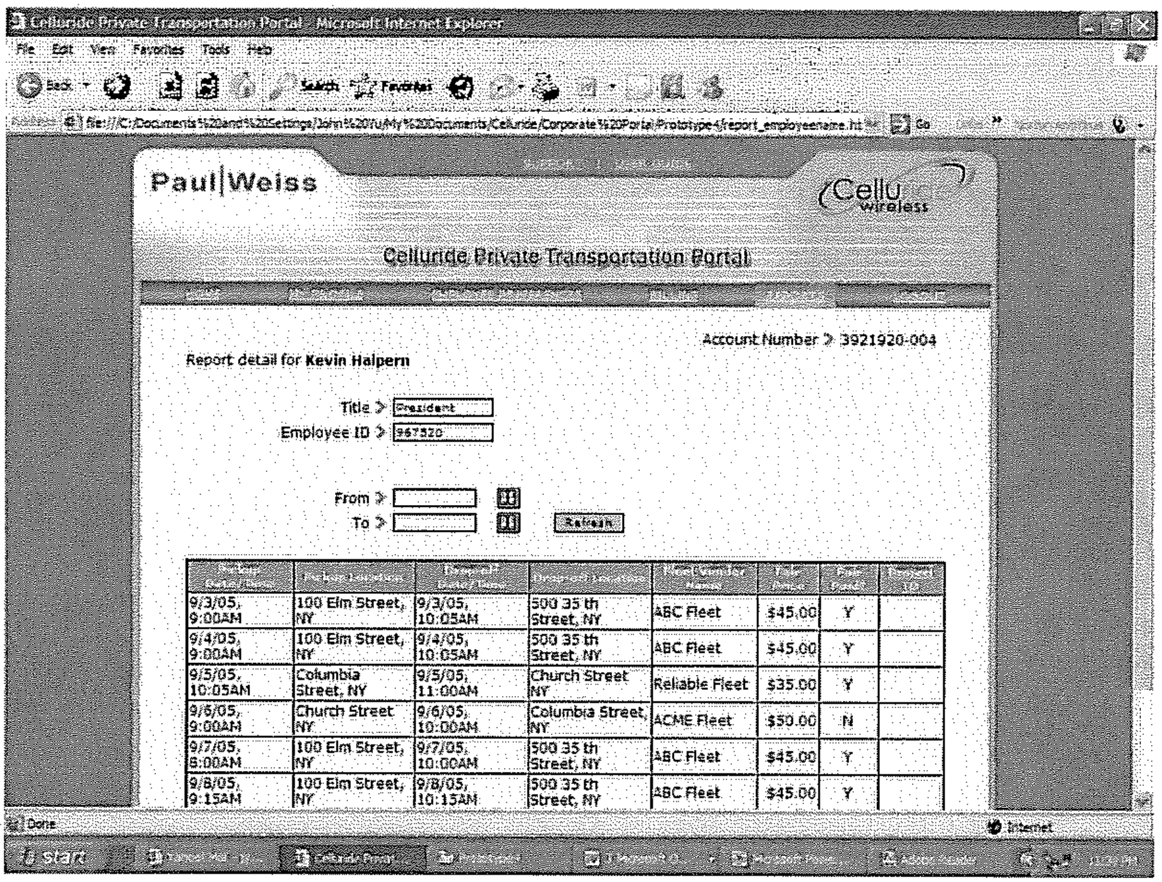
Checking Account Name

Save Bank Account Information? ☑

Following features shall be supported from the Billing page:

- Ability to view monthly bill online
- Ability to pay bill online using either check or credit card
- Ability to view billing history
- View breakdown of charges by fleet vendor

5.1.1.7 Reports Page



Celluride Private Transportation Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
file:///C:/Documents%20and%20Settings/John%20Yu/My%20Documents/Celluride/Corporate%20Portal/Prototype4/report_employeename.ht Go
Paul|Weiss
Cellu wireless
Celluride Private Transportation Portal
Done
Internet
start

Account Number > 3921920-004

Report detail for **Kevin Halpern**

Title > President
Employee ID > 967520

From >
To >
Refresh

Pickup Date/Time	Pickup Location	Dropoff Date/Time	Dropoff Location	Fleet Vendor Name	Trip Price	Fee Paid?	Project ID
9/3/05, 9:00AM	100 Elm Street, NY	9/3/05, 10:05AM	500 35 th Street, NY	ABC Fleet	$45.00	Y	
9/4/05, 9:00AM	100 Elm Street, NY	9/4/05, 10:05AM	500 35 th Street, NY	ABC Fleet	$45.00	Y	
9/5/05, 10:05AM	Columbia Street, NY	9/5/05, 11:00AM	Church Street NY	Reliable Fleet	$35.00	Y	
9/6/05, 9:00AM	Church Street NY	9/6/05, 10:00AM	Columbia Street, NY	ACME Fleet	$50.00	N	
9/7/05, 8:00AM	100 Elm Street, NY	9/7/05, 10:00AM	500 35 th Street, NY	ABC Fleet	$45.00	Y	
9/8/05, 9:15AM	100 Elm Street, NY	9/8/05, 10:15AM	500 35 th Street, NY	ABC Fleet	$45.00	Y	

Following features shall be supported from the Reports page:

- Ability to view ride history of each employee
- Ability to view services provided by each fleet vendor
- Ability to view ride history by project ID and case number
- Ability to define date range for the search criteria

5.1.2 Paperless Voucher System

5.1.2.1 Overview

Often, the fleets use a paper voucher to invoice their client. Once the voucher is submitted to the corporations it has to be processed. Matching vouchers with bills and aggregating individual and departmental data across various vendors is a time-consuming and laborious. Celluride provide an electronic voucher system that will free corporate users and accounting personnel from dealing with paper based confirmations, invoices and receipts. It utilizes the GPS to verify the location of the transaction took place. It will also enable corporations to reduce the possibility of fraud or waste. Following diagram describes Celluride's paper-less voucher system.

5.1.2.2 Message Flow Diagram

Figure 2: Paperless voucher system process

5.1.2.3 Process Flow Description

The following is a description of the steps in this process flow.

1. Passenger confirms the ride has successfully completed
2. Driver information, passenger information, fare, location stamp and time stamp is recorded using GPS
3. Data is sent to Celluride server
4. Celluride server processes and manipulates the data
5. Data is displayed to the PTM and Fleet system

5.1.3 TBD Requirement

5.1.3.1 Overview

This is overview of the TBD requirement.

5.1.3.2 Process Flow Diagram

Figure 3: description of the process flow

5.1.3.3 Process Flow Description

The following is a description of the steps in this process flow.

1. first step.
2. second step.

5.2 Passengers

This section describes the requirements pertaining to passengers. There are two application components for supporting passengers. First component is the application on the handset and second is the Web component.

5.2.1 Passenger Handset Application

5.2.1.1 Overview

The handset application shall allow passengers to find the ride anytime anywhere.

5.2.1.2 Requirement Summary

The following is a summary of requirements for the handset application:

1. Shall support J2ME, BREW, WAP, SMS
2. Shall support passenger authentication and authorization
3. Shall be able to find available car/limo from the phone based on location
4. Shall only display the available car/limo based on the passenger's preferences previously set by the corporate travel manager
5. Shall provide real-time online marketplace for available car/limo by price and by vehicle type
6. Shall allow passenger to book the ride now
7. Shall provide the estimated time of arrival of the vehicle either by text distance countdown or map display.

5.2.1.3 Message Flow Diagram

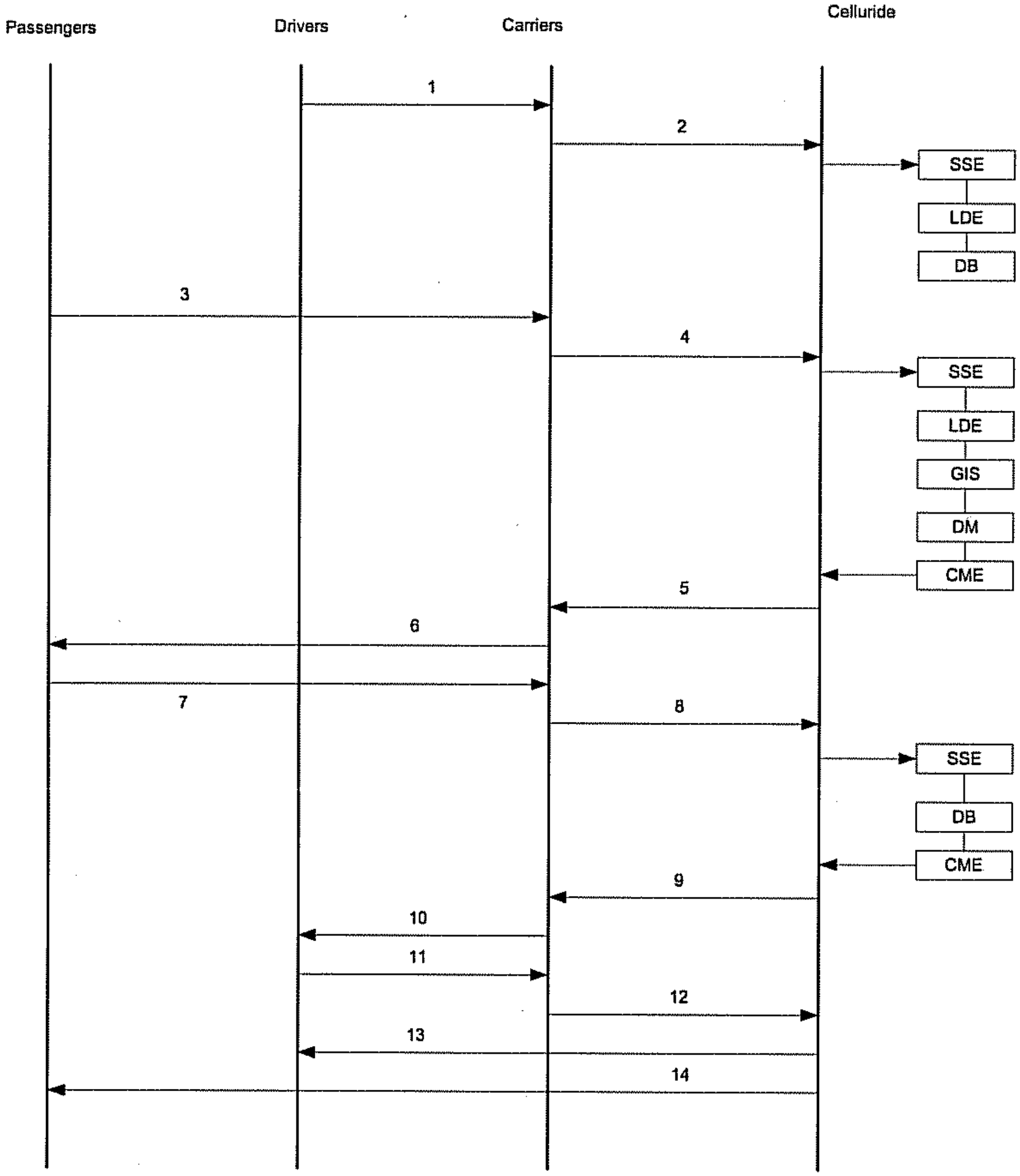


Figure 4: Celluride end-to-end Black Car and Limo Finder Message Flow

5.2.1.4 Message Flow Description

The following is a description of the steps in this message flow.

1. A driver who needs to find new passengers pushes a button on his cell phone. A message indicating the driver's availability and location information are sent to the driver's carrier proxy server or gateway.
2. Carrier proxy determines that this packet belongs to Celluride and forwards it to Celluride's server. This packet will first be processed by Celluride's Subscriber & Service Engine (SSE), which authenticates the driver. Then Celluride's Location Determination Engine is called to retrieve the driver's location. The location determination method can be handset based or network based, depending on the cell phones location technology. Finally, the driver's latest location and availability information are saved in the database thru the DB module.
3. A passenger looking for a ride pushes a button on his mobile handset hosting Celluride's application front-end GUI, indicating the passenger's need for a ride. This information with the passenger's location is sent to passenger's carrier proxy server.
4. The carrier proxy determines that this message belongs to Celluride and forwards it to Celluride's server. The server determines this message is from a passenger. It then calls the SSE to authenticate the passenger, retrieves the passenger's location, and then calls the GIS module to find out all the drivers that are close to this passenger's location. The selected drivers are further filtered by the Data Mining (DM) module. The final list of drivers are those that meet the user's preset criteria for vehicle options.
5. The Celluride's Communication Engine (CME) is invoked and the selected lists of drivers are pushed back to the passenger, thru the carrier proxy.
6. The carrier forwards the above message to the passenger.
7. The passenger looks at this list and picks the final choice of the driver. A message is generated.
8. The carrier proxy determines this message belongs to Celluride and forwards it. The Celluride server invokes SSE to log user's transaction information and saves any necessary billing information in the DB
9. Celluride invokes the CME to send a notification message to the selected driver.
10. The carrier pushes this message to the driver's handset.
11. The driver either accepts or denies the request. An indication message was generated.
12. Carrier forwards driver's decision over to Celluride.
13. Celluride logs the transaction information for billing purpose and sends the routing information and maps showing the driver how to get to the passenger's location.
14. Celluride sends a confirmation message to the passenger.

5.2.1.5 Use Cases

What are different use cases?.

5.2.1.6 Message Flow Diagram

5.2.1.7 Message Flow Description

The following is a description of the steps in this message flow.

5.2.2 Passenger Web Portal

5.2.2.1 Overview

The passenger Web portal is an online tool designed for passengers (or employees) to manage to make online travel reservations for future booking. The passenger Web portal shall be integrated with the PTM portal. This message flow depicts passenger web portal for scheduling and reservation.

5.2.2.2 Modules

The PTM portal consists of three main modules.

1. Passenger Profile
2. Online Reservation System including new/modify/cancel
3. Reservation History

5.3 Fleet Operators

This section describes the requirements for the fleet operators:

5.3.1 Fleet Portal

5.3.1.1 Overview

Fleet operator portal overview.

5.3.1.2 Modules

The Fleet Operator portal consists of four main modules.

1. Administrator Profile
2. Location and Dispatching
3. Billing
4. Reporting

5.4 Drivers

5.4.1.1 Overview

This section describes the requirements for drivers. The driver will have an application on the handset for managing ride schedules and customer pickup.

5.4.2 Driver Handset Application

5.4.2.1 Overview

Overview of the handset application.

5.4.2.2 Message Flow Diagram

Figure 5: User Interface

5.4.2.3 Message Flow Description

The following is a description of the steps in this message flow.

1. Blah blah blah.
2. Blah blah blah…

5.4.2.4 Overview

Blah blah blah blah.

5.4.2.5 Message Flow Diagram

5.5 Celluride System Operators

This section describes the requirements for the Celluride System Operators…

5.5.1 TBD Requirement

5.5.1.1 Overview

Blah blah blah blah….

5.5.1.2 Message Flow Diagram

6. NON-FUNCTIONAL REQUIREMENTS

6.1 Interoperability

State interoperability requirements here…

6.2 Privacy

State privacy requirements….