[http://sharepoint.telenav.com/Logos and TeleNav Art/_w/TN_logo_177x46_gif.jpg](http://sharepoint.telenav.com/Logos%20and%20TeleNav%20Art/TN_logo_177x46.gif)

**Revision: 1.0**

TN 6X Movie CServer

Component Functional Specification Document

**Date of Release: See cBL**

###### **Component Owner: See cBL**

###### **Date Completed: See cBL**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| See cBL | 1.0 | Initial Release | See cBL |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1. Introduction 4](#_Toc290978892)

[1.1 Component Description 4](#_Toc290978893)

[1.2 Changes in This Release 4](#_Toc290978894)

[2. Specific FSD Changes 4](#_Toc290978895)

[3. Requirements 4](#_Toc290978896)

[3.1 Use Cases 4](#_Toc290978897)

[3.2 Service Level Requirements (as needed) 4](#_Toc290978898)

[4. Design 5](#_Toc290978899)

[4.1 Technologies 5](#_Toc290978900)

[4.1.1 Technologies Utilized 5](#_Toc290978901)

[4.1.2 Alternative Technologies Considered (as needed) 5](#_Toc290978902)

[4.2 External Interface(s) 5](#_Toc290978903)

[4.2.1 Communication Protocol(s) Used 5](#_Toc290978904)

[4.2.2 Interface Operations 5](#_Toc290978905)

[4.3 Implementation 6](#_Toc290978906)

[4.3.1 Structural Diagrams 6](#_Toc290978907)

[4.3.2 Behavior Diagrams 11](#_Toc290978908)

[4.3.3 Data Models/Database Schema (as needed) 11](#_Toc290978909)

[5. Service Level Impacts (as needed) 11](#_Toc290978910)

Component

# Introduction

## Component Description

The TN 6.x Movie C-Server is responsible for providing functionality related to Movies search and results. The client interacts directly with Movie C-Server and it further interacts with the backend movie module through provided Java APIs. All the theatres, movies, show times data is passed back to the client for further processing.

## Changes in This Release

|  |  |
| --- | --- |
| **Project Name/Bug Number** | **Description** |
| Please see attached cBLTN6xMovieCServer.html | |

# Specific FSD Changes

*Describe specific sections in this document changed for this release (highlight in yellow):*

| **Section** | **Description** |
| --- | --- |
| N/A |  |

# Requirements

## Use Cases

*This section should list the specific use cases that would cause this component to be invoked.*

| **Use Case #** | **Description** | **Note** |
| --- | --- | --- |
| 1 | Movie Search |  |
| 2 | Movies Search |  |
| 3 | Theaters Search |  |
| 4 | Buy Ticket |  |
| 5 | Share Movie |  |
|  |  |  |
|  |  |  |

## Service Level Requirements (as needed)

*This section should list the service level requirements that this component needs to satisfy.*

| **Requirement #** | **Description** |
| --- | --- |
| N/A |  |

# Design

## Technologies

### Technologies Utilized

* Java SE 6: The primary implementation language for all server-side implementations.
* Apache Tomcat 6: The primary application container platform for application services and business services.
* Apache Axis2 1.5: The primary web services container for business services
* SOAP/HTTP: The primary communication protocol used for communication to business services.
* Spring Framework 2.5.6
* Apache Commons
* Log4J 1.2.8

TeleNav proprietary technologies:

* INode: The interface abstraction for TXNode for client to server communication

### Alternative Technologies Considered (as needed)

*If any other technologies were considered but rejected, they should be listed here, along with the rational for why there were rejected in favor of another choice.*

## External Interface(s)

### Communication Protocol(s) Used

INode: The interface abstraction for TXNode for client to server communication

TXNode/INode: The proprietary TN protocol used for client-server communication. It has been optimized for transfer size. INode represents the generalized interface for TXNode to abstract away the concrete implementation and isolate changes

### Interface Operations

Various operations below are handled by the Common CServer over HTTP using TxNode protocol.

#### Buy Tickets

Users can buy tickets for a selected movie using this interface.

#### Movie Details

Clients get details for a selected movie using this interface. The details include Times, movie theatres, Genre, Rating and small description.

#### Movie Show Times

Show times for a selected movie are provided using this interface.

#### Share Movie

Selected movie can be shared with other users with this interface.

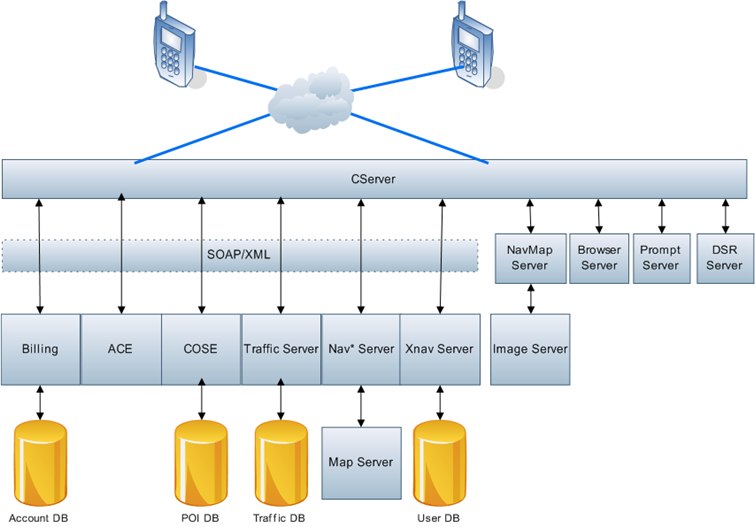
#### 

## Implementation

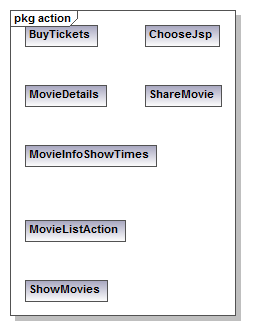
### Structural Diagrams

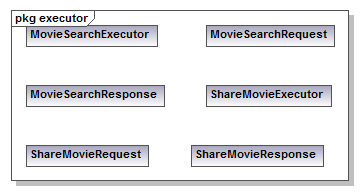


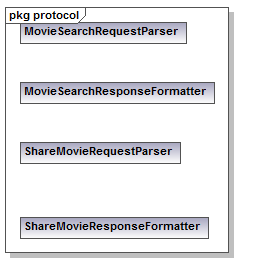
#### Layer Diagrams

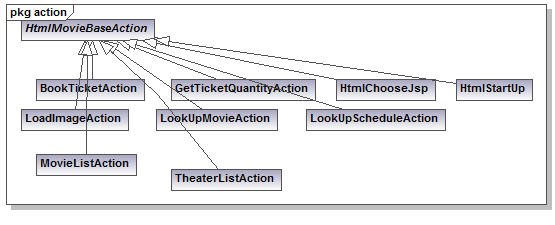


#### Class Diagram

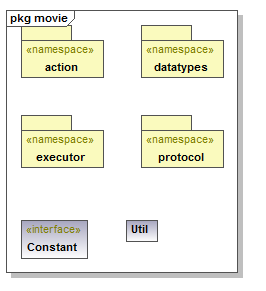


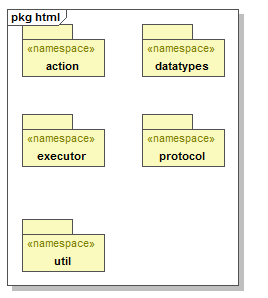






#### Package Diagram





### Behavior Diagrams

#### UC01 Movie Search

1. Search request initiated with a movie name hits Movie CServer along with location and data/time parameters.
2. Movie search server is requested with the parameters.
3. A response is returned with movie details and all theaters playing that movie.
4. Movie CServer returns this list back to the device.

#### UC02 Movies Search

1. Search request initiated without a movie name hits Movie CServer. Other parameters are still sent like location and data/time.
2. Movie search server is requested with the parameters.
3. A response is returned with list of all movies around that location and time/date.
4. Movie CServer returns this list back to the device.

#### UC03 Theaters Search

1. Theater search request initiated with a movie name hits Movie CServer along with location and data/time parameters.
2. Movie search server is requested with the parameters.
3. A response is returned with all theaters playing that movie.
4. Movie CServer returns this list back to the device.

#### UC04 Buy Ticket

1. Buy ticket button is displayed for all Movies and theaters which are associated with Fandango.
2. Clicking the button should take the user directly to Fandango page.

#### UC05 Share Movie

1. Share Movie request is initiated with friend’s phone number and movie details (time, location etc.)
2. Request is sent to xNav server which sends the SMS with all information to share.



### Data Models/Database Schema (as needed)

*N/A*

# Service Level Impacts (as needed)

*N/A*