



Ticket to Ride Rulebook

AI-MAS Winter Olympics 2011

© 2010 AI-MAS Group



Table of Contents

Glossary	3
Game Concept	4
Introduction.....	4
Goal of the game	4
Game components	5
Game Rules.....	6
Drawing Train Cards	6
Claim a Direct Route	7
Drawing Destination Tickets	9
Build a Train Station	9
Game progress.....	11
Setting up the game	11
Beginning the game	11
Game Turn	11
Game End	12
Game Score.....	13
Technical Specifications.....	14
Game components	14
Map.....	14
Destination Tickets	15
Train Car Cards	15
Actions	16
Drawing a card.....	16

Glossary

SDK - kit that is given to the participants to support the implementation of the Solutions for the Game

Framework - all the classes/libraries that are needed to run a Solution for the Game

Solution - code that adheres to the Specifications given in the SDK, and that, when executing within the proper context, plays according to the Rulebook

Scenario - a set of values for the parameters established at the beginning of the Game.

Solution Instance - a running Solution, including its internal state.

Game Instance - a running Scenario, including its internal state, that represents the progress of the Game.

Server - executable that creates a Game Instance, and to which Clients can connect.

Client - executable that creates a Solution Instance, connects to a Server, and submits the Solution's actions to the Server

Action - discrete event that alters the state of the Game.

Turn - time frame in which one action is performed by one player.

Direct Route - connection between two adjacent cities on the map.

Route - connection between two cities that are not adjacent on the map, and that can contain train stations.

Continuous Route - a series of connected direct routes, connecting two cities that are not adjacent on the map, that doesn't contain Train Stations and passes through each direct route only once.

Game Concept

Introduction

Ticket to Ride is the implementation of the popular board game “Ticket to Ride”, a cross-country train adventure, in which players collect and play matching train cards to claim railway routes connecting cities on the map.

Goal of the game

Ticket to Ride is a turn based strategy game that challenges the player to gain as many points as possible by making train routes between cities, following a set of rules.

Points are awarded for:

- Successfully claiming a direct route
- Successfully completing a route listed on a Destination Ticket
- Having the longest continuous route on the map at the end of the game
- For each unused Train Station (some maps may not have train stations)

Points can be deducted for:

- Any Destination Ticket owned by the player and not completed at the end of the game

Each solution will have a total of two minutes for the entire game. If a current player’s time ends, and there are still other players with available time, this current player won’t be able to perform any other actions. The game will go on until no player has available time or the players finish the final turn.

Game progress

Ticket to Ride is played by 2, 3, 4 or 5 players at a time. The players will take turns to perform one of 4 actions possible in order to gain points.

Game components

- 1 map with all the train routes
- 46 Destination Ticket Cards (the number may be subject to change, depending on the map)
 - 40 regular routes
 - 6 long routes
- 45 Train Cars for each player (for each player in a different color)
- 110 Train Car Cards (the number may be subject to change, depending on the map)
 - 12 Train Car Cards for each of 8 types (for some maps the number of types may change)
 - 14 Locomotives

Game Rules

Drawing Train Cards

For the Europe map, there are 110 Train Car Cards: 12 Train Car Cards of each of the 8 color (purple, red, white, green, blue, yellow, black, orange) and 14 Locomotives. (For other maps, the number of types may be different). The routes are either grey or they have one of the colors on the Train Car Cards.

If a player chooses to draw Train Car Cards, he may draw 2 cards per turn. Either of these cards can be drawn from the 5 cards available in the open deck or from the normal deck (in which case the player will not know in advance what cards he will get - blind draw). If a card from the open deck is drawn, it is immediately replaced with the first card from the normal deck.

There are some special situations:

1. If a Locomotive card is drawn from the open deck, then that card will be the only card drawn that turn.
2. If a Locomotive card is available in the open deck, but was not selected first, it cannot be selected as a second card.
3. If a player draws a card from the open deck that is not a Locomotive Card, then he can draw a second card from the open deck, except a Locomotive card, or he can draw a card from the normal deck.
4. If a player draws a card from the normal deck, he can then draw the second card from the normal deck or from the open deck (except a Locomotive card).
5. If a player gets a Locomotive card from the normal deck (blind draw), it still counts as a single card and he can draw another one from the normal deck, or from the open deck, except a Locomotive card.

A player may have any number of Train Car Cards in his hand at any time (the limit being the number of Train Car Cards in the game). When the normal deck is exhausted, the discarded Train Car Cards are reshuffled into a new deck.

In the event that there are no cards left in the deck and there are no discards (because the players have all the cards in their hands), a player won't be able to draw anymore Train Car Cards. Instead, he may perform any of the other available actions.

In the event that the open deck contains 3 or more Locomotive Cards, the open deck cards will be discarded and the next 5 cards from the normal deck will be placed in the open deck. If there are not enough cards in the normal deck, all the cards will be reshuffled until there are less than 3 Locomotive Cards in the Open Deck, or the cards were reshuffled 5 times.

Locomotives

Locomotives are multi-colored and will act as wild cards in the game (they can replace any color Train Car Card).

Locomotive cards can be played along with any set of Train Car Cards when claiming a route.

Claim a Direct Route

A direct route is a number of spaces between 2 adjacent cities on the map. To claim a direct route, a player must use a set of Train Car Cards whose color and quantity match the color and number of spaces (the cost) of the chosen route.

Most direct routes require a specific set of colored cards to claim them. Grey direct routes can be claimed using a set of cards of any one color.

When a route is successfully claimed, the owner uses one of his Train Cars for each space of the route. The number of Train Cars used will be deducted from the number of Train Cars that the player has at that moment. All the Train Car Cards used to claim the direct route are then discarded. The score is immediately updated.

A player may claim any direct route on the map. He is not necessarily required to connect to any of his previously played routes.

A direct route must be claimed in its entirety during a single turn. Only one route can be claimed during one player's turn.

Double Routes

Some cities are connected by Double Routes. There are 2 parallel and equal direct routes from one city to another. One player can never claim both direct routes between the same 2 cities during a game.

Note: In 2 or 3 player games, only one of the 2 direct routes can be used. A player may claim any of the 2 direct routes, but the other one is then closed to the other players for the duration of the game.

Ferries

Ferries are special gray routes linking 2 adjacent cities across a body of water. They require at least one Locomotive (depending on the ferry) in order to be claimed.

To play a Ferry Route, a player must play a Locomotive card for each Locomotive required by that Ferry, and the usual set of cards of the same one color for the remaining spaces of that Ferry Route.

Tunnels

Tunnels are special direct routes. When a player wants to claim a tunnel, he must have the same number of Train Car Cards of one color as the number of spaces in the tunnel (cost of the tunnel). Then the first 3 cards from the normal deck are revealed. For each card revealed whose color matches the color of the cards played to claim the tunnel, an additional card of the same color (or a Locomotive) must be played by the player. Only then the player will successfully claim the tunnel route.

If a player does not have enough additional Train Car Cards of the same color (or Locomotives to replace them) or does not wish to play them, he may take back the cards and his turn ends.

At the end of the turn, the 3 Train Car Cards revealed from the normal deck are discarded.

Because Locomotives are wild cards, any Locomotive drawn from the top of the normal deck will automatically match the color of the Train Car Cards played for the tunnel, forcing the player to use another Train Car Card.

If a player attempts to claim a tunnel using Locomotive cards exclusively, he will only have to play additional cards (Locomotive Cards in this case) if Locomotive Cards show up among the 3 drawn from the normal deck.

If a player attempts to claim a tunnel, and in the normal deck there are less than 3 cards available, but there are more cards in the discarded card pile, then all the cards will be shuffled and 3 cards will be revealed to determine their effect on the cost of the tunnel.

In the event that there are less than 3 cards to reveal and no discarded cards, only the cards available are revealed. If, as a result of players keeping all the cards, there are none to be revealed, a Tunnel can be claimed without any additional cards to the original cost of the Tunnel.

Drawing Destination Tickets

The player draws 3 Destination Tickets from the top of the missions deck. He must keep at least one of them, but he may keep more. Any discarded tickets are placed at the bottom of the Destination Tickets deck.

If there are less than 3 Destination Tickets available, the player draws the Destination Tickets that are available. Tickets drawn, and not immediately discarded, must be kept until the end of the game (they cannot be discarded during a later Ticket draw).

The cities listed on a Destination Ticket represent travel goals for the player, and they can result in a bonus if completed, or in a penalty otherwise.

Destination Tickets are kept secret from other players. A player may have any number of Destination Tickets during the game.

Build a Train Station

A Train Station allows its owner one and only one of the routes belonging to another player, into (or out of) a city, in order to help the player connect the cities on his Destination Ticket.

Stations may be built on any unoccupied city, even if it currently has no claimed routes into it. Two players may never build a Station in the same city.

Each player may build only one Station per turn, and a maximum of 3 Stations throughout the course of the game.

To build his first Station, the player must use 1 Train Car Card of any color before building the Station. To build his second train station, the player must use 2 Train Car Cards of any one color, and for the third a set of 3 cards of any one color.



If a player uses the same Station to help connect cities on several different Tickets, he must use the same route into the city with the Station for all of those tickets.

A player is not required to build any Stations. For each Station a player has not used, 4 points are added to his score at the end of the game.

Game progress

Setting up the game

Each player receives a set of 45 train cars and a set of 3 train Stations (each player will have a different color).

The Train Car Cards are shuffled and each player receives 4 cards.

For the remaining Train Car Cards, the first 5 will be placed in an open deck and the rest will remain in the normal deck. The Train Car Cards in the open deck are visible at all times for all the players, but the ones in the normal deck remain unknown.

From the Destination Ticket Cards each player will receive 1 random long route and 3 short routes. The rest of the cards with short routes will be placed in the Destination Tickets deck and the rest of the cards with long routes will be removed from the game.

Beginning the game

Before the game may begin, players must choose which Destination Tickets to keep among those received. Each player must keep at least 2 Destination Tickets. The discarded Destination Tickets will be removed from the game, and the ones kept by each player must be held until the end of the game. The Destination Tickets that are put away may either be long routes or short routes.

Game Turn

On his turn, a player can perform one and only one of the following four actions:

1. Draw Train Car Cards: The player may draw 2 Train Car Cards from the normal deck, the open deck or one from each (or just one if the card he selects is a locomotive from the open deck) respecting the rules concerning Locomotives.
2. Claim a Direct Route: In order to claim a direct route on the map, the player must use a set of Train Car Cards that he has and that match the color and Car Cost of the route. He scores the number of points equal to the route's value from the score table.
3. Draw Destination Ticket: The player draws 3 Destination Tickets and must keep at least one of them.



4. Build a Train Station: The player may build a Station in any city on the map that does not yet have one.

Game End

When one of the players has at the end of his turn only 2 or less virtual Train Cars, each player, including that player, gets one final turn. The game then ends and the score is computed.

The game may also end when all players have exhausted their time limit.

Game Score

Route scoring is done during the game considering the following table (only for the Europe Map):

Route Length	1	2	3	4	6	8
Points Scored	1	2	4	7	15	21

Table 1. *Route scoring table for the Europe Map*

At the end of the game the value of the successfully completed Destination Tickets is added to the player's total score. The value for any incomplete Destination Tickets is deducted from the total score.

Each player receives 4 points for each of his unused Stations.

The player(s) with the Longest Continuous Path on the board will receive 10 bonus points.

When evaluating and comparing path lengths, only continuous routes are taken into account. Continuous paths may include loops, and pass through the same city several times, but they cannot contain twice the same direct route. Stations, and the opponents' routes to which they may provide access, do not count for the purpose of computing the score and claiming the longest route.

The player with the most points wins the game.

When 2 or more players are tied with the most points, the player who has completed the Destination Tickets having the most points is the winner. If still tied, the player with the most points from unused Stations is declared the winner. If the players are still tied, the player that received the longest route bonus wins. If there are still 2 or more players with the same score, they will play a special round (best of 5 games).

Note: The rules and scoring table are associated with the map of Europe. Any additional maps will be documented in the SDK, in terms of route scoring.

Technical Specifications

Game components

Map

The map is represented as a graph and is implemented through the *TTRMap* class. The graph's nodes define the existing cities on the map, while its edges represent the direct routes between 2 adjacent cities. The map description can be found in the associated .xml file (e.g. europe.xml)

```
<map name="europe">
  <nodes>
    <node name="Edinburgh" x="x" y="y" />
    <node name="London" x="x" y="y" />
    <node name="Amsterdam" x="x" y="y" />
    <node name="Bruxelles" x="x" y="y" />
    <node name="Dieppe" x="x" y="y" />
    <node name="Brest" x="x" y="y" />
    <node name="Stockholm" x="x" y="y" />
    <node name="Kobenhavn" x="x" y="y" />
    <node name="Essen" x="x" y="y" />
    <node name="Berlin" x="x" y="y" />
    <node name="Frankfurt" x="x" y="y" />
    <node name="Danzig" x="x" y="y" />
    <node name="Riga" x="x" y="y" />
    <node name="Wien" x="x" y="y" />
    <node name="Munchen" x="x" y="y" />
    <node name="Warszawa" x="x" y="y" />
    <node name="Wilno" x="x" y="y" />
    <node name="Smolensk" x="x" y="y" />
    <node name="Moskva" x="x" y="y" />
    <node name="Petrograd" x="x" y="y" />
    <node name="Kyiv" x="x" y="y" />
    <node name="Kharkov" x="x" y="y" />
  </nodes>
</map>
```

Image 1: Extract from the europe.xml file

Destination Tickets

Destination Tickets are instances of the *TTRMissions* class and contain a source node value, a destination point, and a mission point value. Short Missions are Destination Tickets with a value of 6-12 points and Long Missions are Destination Tickets with a Value of 20-21 points.

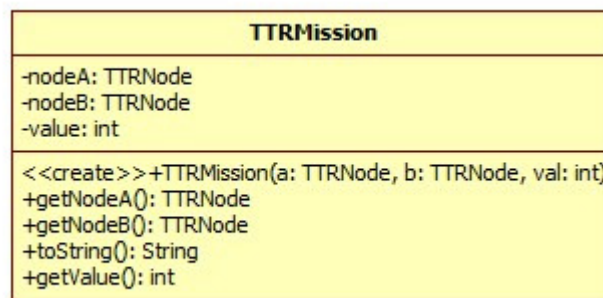


Image 2: TTRMission class diagram

Train Car Cards

The Train Car Cards may belong to one of the following color values: purple 'p', white 'w', blue 'b', yellow 'y', orange 'o', red 'r', green 'g', black 'n', gray '*' (the string 'wildcard' is used as value in the source code) or no color '0' (the value ttnull is used). Locomotives are referred to as engines 'e'.

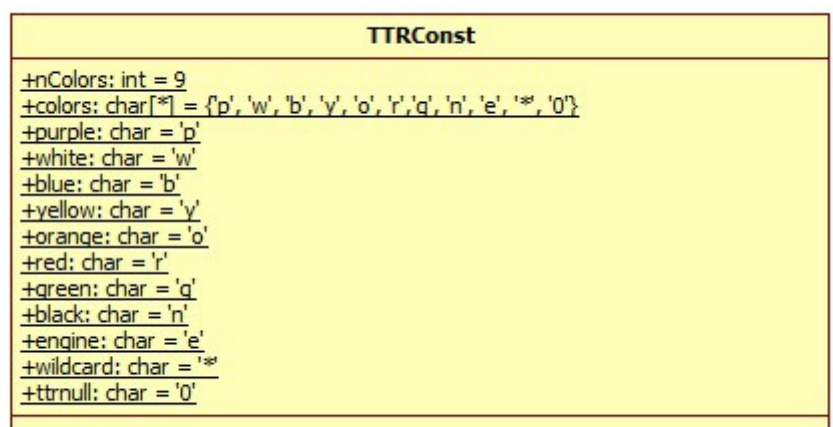
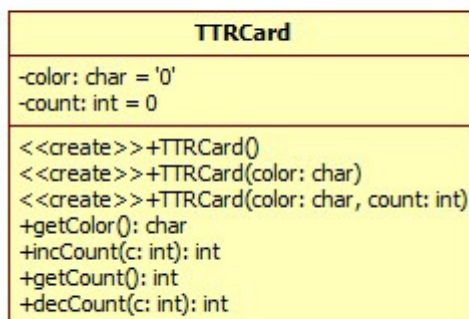


Image 3: TTRCard and TTRCons class diagrams

Actions

Drawing a card

Drawing a card from the normal deck

Attempting to draw a card from the normal deck, by calling the *drawCardFromDeck* method, will return the following values:

- Method call: *drawCardFromDeck*
- Parameters: none
- Returns: integer
 - 1 if successful
 - 0 if successful, but there are no cards left in the deck
 - < 0 if the operation failed (error codes from -1 to -7)

Drawing a card from the open deck

When attempting to draw cards from the open deck, the player will make a call to the *drawCardFromOpenDeck* method mention, sending as parameter the color of the card he wishes to draw. Attempting to draw a card from the open deck will return the following values:

- Method call: *drawCardFromOpenDeck*
- Parameters: char colour
- Returns: integer
 - 1 if successful
 - 0 if successful but there are no cards in the open deck
 - <0 if the operation failed:

Drawing Destination Tickets

Attempting to draw Destination Tickets will return the following results

- Method call: *drawMissions*

- Parameters: none
- Returns: integer
 - The number of Destination tickets drawn if the operation was successful
 - 0 if there are no more Destination Tickets available
 - < 0 if the operation failed (error codes from -1 to -7)

Rejecting Destination Tickets

If the Destination Tickets were drawn successfully, then the player may choose to keep all the missions (with the keepAllMissions method) or reject some of the missions, in which case the player will call the rejectMissions method with the missions he wants to reject as parameters (some of the parameters may be null, depending on how many missions the player wants to reject).

- Method call: keepAllMissions
 - Parameters: none
 - Returns: integer
 - 0 if successful, but there are no cards left in the deck
 - < 0 if the operation failed (error codes from -1 to -7)
-
- Method call: rejectMissions
 - Parameters: 2 TTRMission objects, can be NULL
 - Returns: integer
 - Greater than or equal to 0, returns the number of rejected missions on success
 - < 0 if the operation failed (error codes from -1 to -7)

Claiming a route

For claiming a direct route, the player must use the claimRoute method with the source node, destination node, color of the train car cards to build with, number of wagon cards (normal train car cards)

and number of engine cards. If a player successfully claims a route, a command will be sent to all the players informing them that the route was claimed. Ferry routes are treated like normal routes.

- Method call: claimRoute
- Parameters: integer Node1, integer Node2, char CardColor, integer Wagons, integer Engines
- Returns: integer
 - greater than 0, returns the number of cards required in case of building a tunnel
 - equal to 0, on success and no additional cards required
 - < 0 if the operation failed (error codes from -1 to -7)

In the case that the route claimed is a tunnel, the player might receive a demand for extra Train Car Cards in order to claim the tunnel. If this happens, all the players will be informed of the demand. In order for a Player to claim the tunnel route he has to use the same command as before, only with extra wagon cards or engine cards. If he doesn't want to use extra train car cards, he may answer with a Claim Pass, and in which case his turn ends without successfully claiming the tunnel.

- Method call: claimPass
- Parameters: none
- Returns: integer
 - equal to 0 on success
 - < 0 if the operation failed (error codes from -1 to -7)

Building a station

Building a station requires calling the buildStation method with the source node, destination node, card color and number of wagons and/engines used to build the station. The player will receive a 0 as a response if the operation was successful.

- Method call: buildStation
- Parameters: integer Initial Node, integer Connection Node, char Card Color, integer Wagons, integer Engines
- Returns: integer

- equal to 0 on success
- < 0 if the operation failed (error codes from -1 to -7)

myMethod

When it is the player's turn, the abstract method *myMethod* will be called. This method must be implemented by the player and it represents his solution for the game. An example of a random *myMethod* is implemented in the *RandomSolution* class which extends the *TTRClient* class.

RandomSolution
<u>~DRAW FROM OPEN DECK: int = 0</u> <u>~DRAW FROM DECK: int = 1</u> <u>~DRAW MISSIONS: int = 2</u> <u>~CLAIM ROUTE: int = 3</u> <u>~BUILD STATION: int = 4</u> ~myRand: Random = new Random() ~actionPerformed: boolean
<<create>> +RandomSolution(user: String, pass: String, agent: String, name: String, host: String, port: int) +myMethod()

Image 4: RandomSolution class implementation