RTT - Reclinathon Tracing Technology a.k.a Reclinathon Time Travel a.k.a ‘Crayon’

Table of Contents

1 | Overview

1.1. Purpose

Reclinathon Tracing Technology (RTT) provides the Reclinathon ecosystem with two major features. First, RTT provides the ability for observers to and participants to track the status of an official RAA Sanctioned LA-Z-DUDE Reclinathon as it unfolds. Second, RTT allows historians to travel back in time to observe and annotate the precise state of any moment during a RTT-enabled Reclinathon.

This technology is intended to supersede the existing mish-mash of web technologies currently available at http://www.reclinathon.com, by providing a Reclinathon's Captain with a toolset to easily manage, display, and record for posterity all of the movies, downtime, meals, etc. needed to successfully guide the event.

1.2. Definitions

Term	Definition
RTT	Reclinathon Tracing Technology
ROC	Reclinathon Organization Committee
Captain	A member of the ROC who is currently responsible for ensuring the success of a Reclinathon. For the purposes of this document, the captain is also responsible for controlling RTT.
Reclinee	An athlete who is currently competing in a Reclinathon (either competitively or for exhibition)
Reclinathon Status	A description of the activity in which Reclinees are currently engaged. (e.g. Reclining, Downtime, Emergency Maintenance, etc.)
Reclinathon Sub-Status	A modifier which provides additional details about a Reclinathon Status (e.g. Meal, Sleeper, Extra-Innings, etc.)
Reclinathon State	The complete context in which a Reclinathon is operating, including but not limited to: Reclinathon Status and Sub-Status, Reclinee List, Reclinee scores, Movie Playing, Movie Queued, Captain, Meal Served.
Record	A data structure representing Reclinathon State at a given point in time
Transition	A shift from one Reclinathon State to another that is caused by natural events that occur during a Reclinathon.
Log	A time-ordered list of Records that spans from a Reclinathon's onset to its conclusion. A transition is defined by appending a new Record to the Log. At the conclusion of a Reclinathon, the Log is a complete transcription of what occurred during the event.

1.3. Design Goals

1. **Easy Real-Time Maintenance**: Minimal effort should be required by the Captain to add new Records to a Log during a Reclinathon. The administrative overhead of RTT should be small enough that the system can be maintained by sleep-deprived Captains without detracting from their Reclinathon experience.

2. **Support for Multiple Data Interfaces:** The main database driving RTT should support multiple interfaces for data entry and retrieval, to allow custom RTT applications to be created for a wide variety of platforms and environments.
3. **Tightly Synchronized In-Arena RTT Clients:** Since a Reclinathon Arena may have multiple RTT clients installed, information should be tightly synchronized between clients and the central RTT data source. Specifically, changes to Reclinathon State should propagate to all in-arena client data views within 1 second of the Transition being recorded by RTT. Therefore, all Reclinees see the same information, regardless of which client they are observing. Note that the synchronization can be much looser, or even on-demand, for RTT clients outside of the Reclinathon Arena.
4. **Easy Integration with the Reclinathon Automation System ('College'):** RTT should support automatic tracing based on input from the Reclinathon Automation System ('College'), by implementing a custom data interface for the Reclinathon Arena server.
5. **Easy Integration with the Fantasy Reclinathon program:** RTT Logs should contain sufficient information for a Fantasy Reclinathon data interface to tabulate scores as a Reclinathon progresses. RTT need not be cognizant of the Fantasy Reclinathon rules or scoring system—it simply acts as a data provider.

2 | Architecture

The architecture of RTT can be summarized by the following class-interaction diagram:

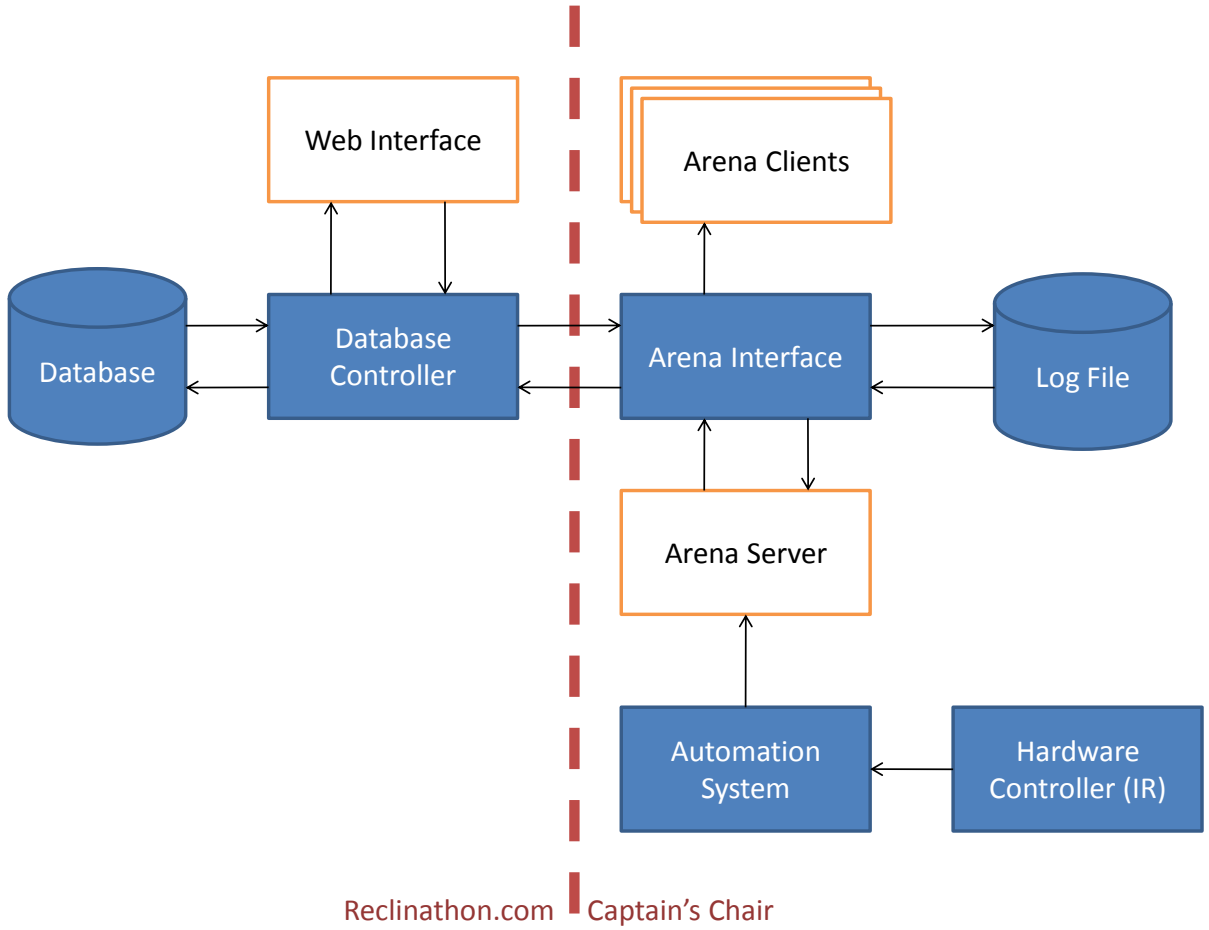


Figure 1: RTT Architecture

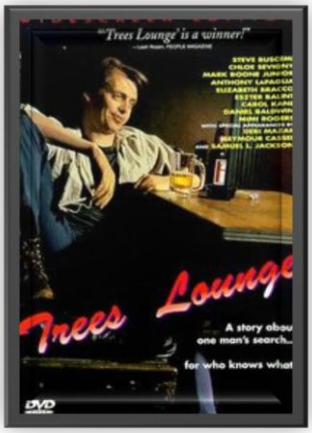
2.1. Mockups

LA-Z-DUDE Reclinathon: Winter 2008

Reclining: Meal	10:24:36 AM
Now Playing:	Trees Lounge
Time Remaining *	1:02:27
Captain:	Dude
Meal:	60 Waffles

History	
Trees Lounge	**10:24:36 AM**
Downtime	10:15:57 AM
The Prophecy	08:35:13 AM
Downtime	08:30:04 AM

Movie Info



Genre:
Comedy / Drama

Runtime:
95 min

Trailer
IMDB

Leader Board

1: Machine	**1200 min**
2: Dude 1	1142 min
2: Dude 2	1142 min
3: Dude 3	1131 min
4: Dude 4	1107 min
5: Dude 5	1038 min
5: Dude 6	1038 min
6: Dude 7	1011 min

Figure 2: Web Interface View of RTT

3 | Public Surface

RTT is publically exposed by each of its supported data interfaces. Each data interface is free to define its own functionality and optional API. In the initial release of RTT, two interfaces will be supported.

3.1. Web Data Interface

The web data interface is implemented as a set of scripts located on the reclinathon.com server. This interface is a stand-alone system that provides no public API.

The interface provides a **Reclinathon Captain Control Page,** which allows the Captain to manually log and modify RTT Records directly to the central database, and a **Reclinathon Tracker Page**, which allows the public to track the status of ongoing Reclinathons and view the history of completed Reclinathons.

3.2. ‘College’-Enabled Arena Data Interface

The arena data interface is implemented as a library, which is linked into the Arena Server and Arena Client binaries. This interface communicates to the Database Controller via HTTP messages to view and update the central database. The interface provides the following API to the Arena Server and Clients:

3.2.1. API:

TBA

4 | Detailed Design

4.1. Data Structures and Stores

TBA

4.2. Database Controller

TBA

4.3. Web Data Interface

TBA

4.4. ‘College’-Enabled Arena Data Interface

TBA

4.5. Arena Server

TBA

4.6. Arena Clients

TBA