Real Time Dashboard/xVR Architecture

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev** | **Date** | **Author** | **Description** |
| 1.0 | 5/1/12 | James Francis, Stephen Madson | Release Version |
| 2.0 | 8/1/13 | Ian Wehmeyer | Revised Version |

Document Approvers & Sign-Off

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Approver** | **Role** | **Document Accept/Reject** |
| 5/1/12 | Ric Merrifield | Release Manager | Accept |
| 5/11/12 | John Stiehl | Release Manager | Accept |

**Table of Contents**

[1 Introduction 4](#_Toc336940571)

[1.1 Purpose 4](#_Toc336940572)

[1.2 Definitions 4](#_Toc336940573)

[2 xi User Interface 5](#_Toc336940574)

[3 xi Web Service Interfaces 10](#_Toc336940575)

[3.1 Executive Summary 10](#_Toc336940576)

[3.2 Entitlements 10](#_Toc336940577)

[3.3 Blue lanes 11](#_Toc336940578)

[3.4 Guests](#_Toc336940579) **[Error! Bookmark not defined.](#_Toc336940579)**

[3.5 Daily Recruiting Report 11](#_Toc336940580)

[4 xi Data store 12](#_Toc336940581)

# Introduction

Please note that throughout this document page names are highlighted in **bold** while page element names are in *italic*.

## Purpose

This document describes the architecture and the design of the Real Time Dashboard user interface, web services, and data store of the system. Explicitly, the dashboards consist of the xi.war file deployed in a TCServer instance, and database tables and stored procedures hosted on SQLServer.

## Definitions

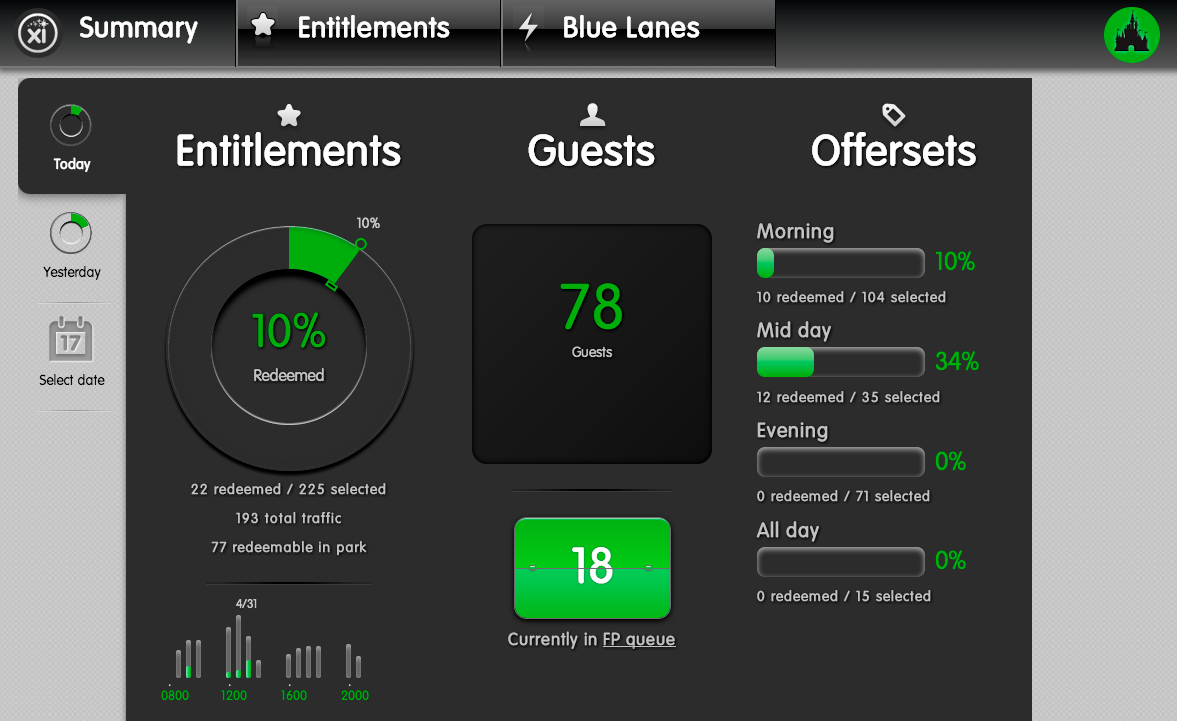
|  |  |
| --- | --- |
| **Term** | **Definition** |
| Entitlements | selected or booked experiences |
| Offersets | a group of entitlements that were booked as a group. all standard entitlements are part of an offerset |
| Redemption | the redemption of an entitlement |
| Blue lanes | extraordinary attempts to redeem entitlements |
| Overrides | a blue lane occurrence that was overridden by a cast member such that a guest was granted access to the experience |
| Total Traffic | all Redeemed Entitlements and Blue Lane Overrides |
| Redeemable in Park | all Entitlements that are currently in the park |
| Visits | Black Bar: All Guests who have entitlements for that day; Green Bar: Guests who have redeemed entitlements |
| Pre-Arrival Metric | How far in advance of their valid entitlement date do Guests book. For instance, if a Guest logs into their account on August 10th and books and entitlement for August 17th we would show them in the -7 bucket |
| Engaged | Guest Distribution throughout the Parks |

# Real Time Dashboard User Interface

The RTD user interface provides the following functionality.

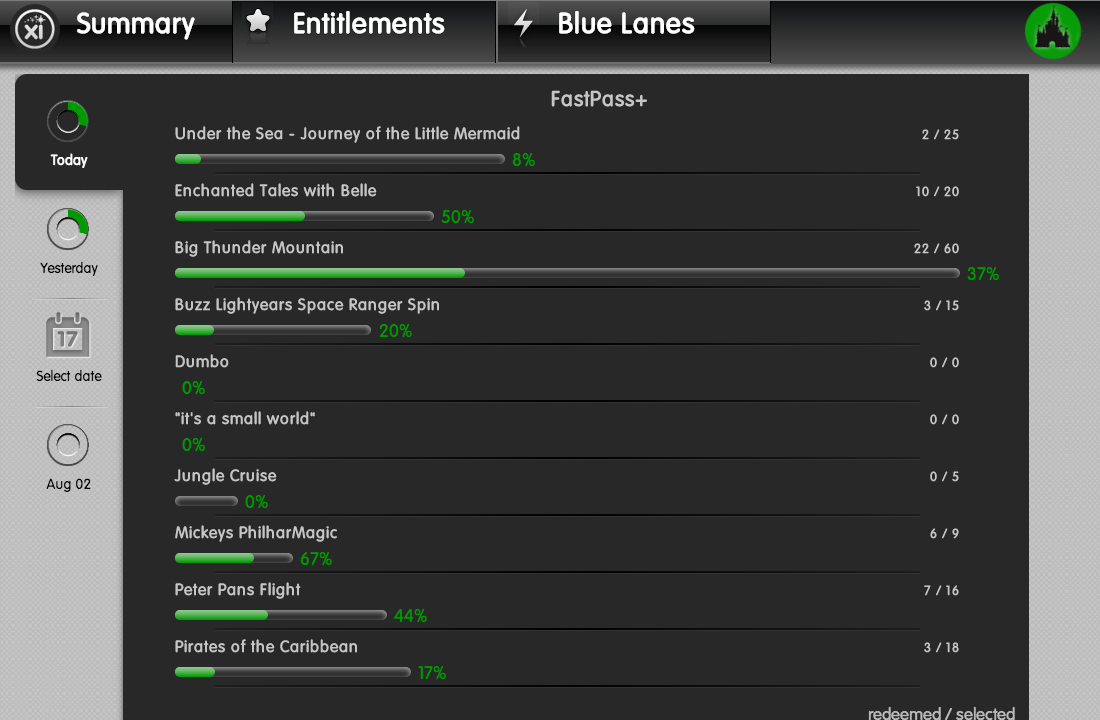
Executive Summary

1. Executive Summary is the main page. It provides a high level view of entitlements, queue counts, total traffic, entitlements in the park, and offerset redemption distribution. Executive Summary is viewable for today, yesterday, to date or for any specified date.



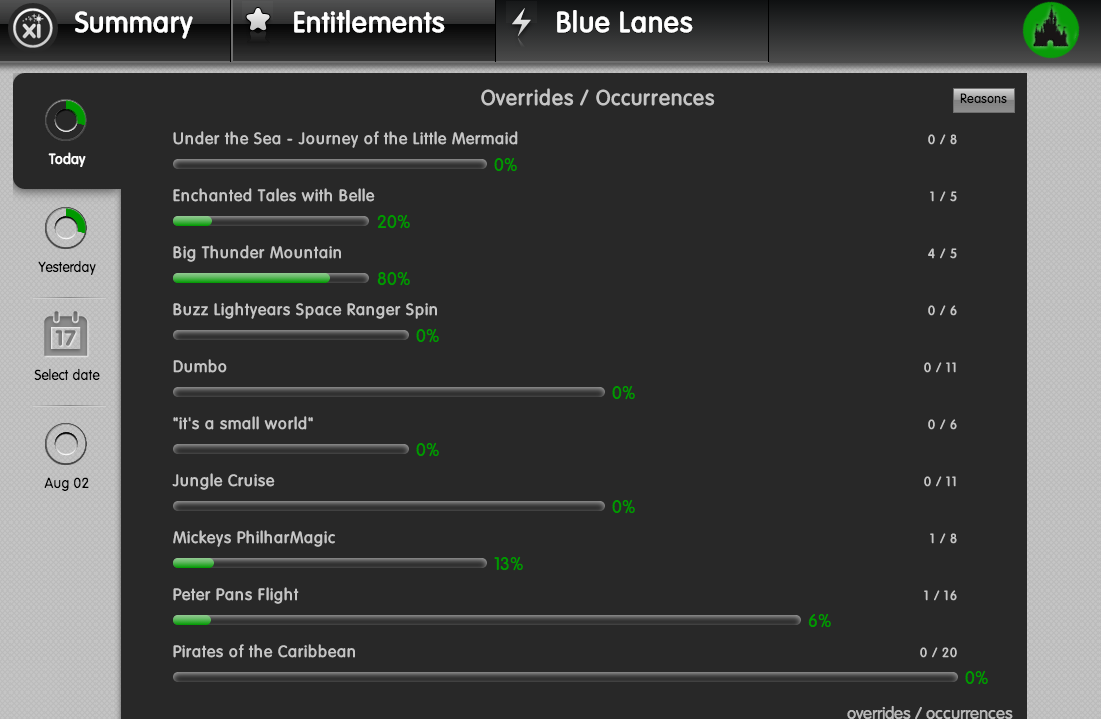
Entitlements

1. The Entitlements view allows a user to drill down into the entitlements data. Redemptions are broken down by attraction. They can be viewed hourly and information about ridership and individual riders can be viewed as well. Entitlements are viewable for today, yesterday, to date or for any specified date.



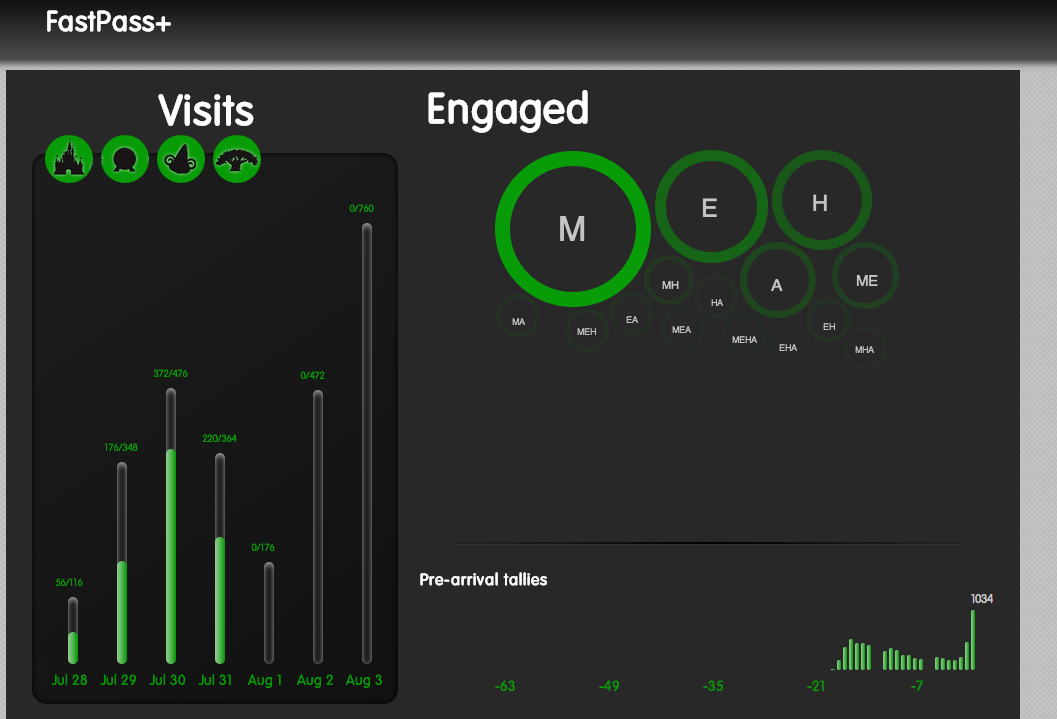
Blue lanes

1. Blue lanes are broken down by attraction. Blue lanes are also broken down by reason code. Blue lanes and overrides are viewable for today, yesterday, to date or for any specified date.



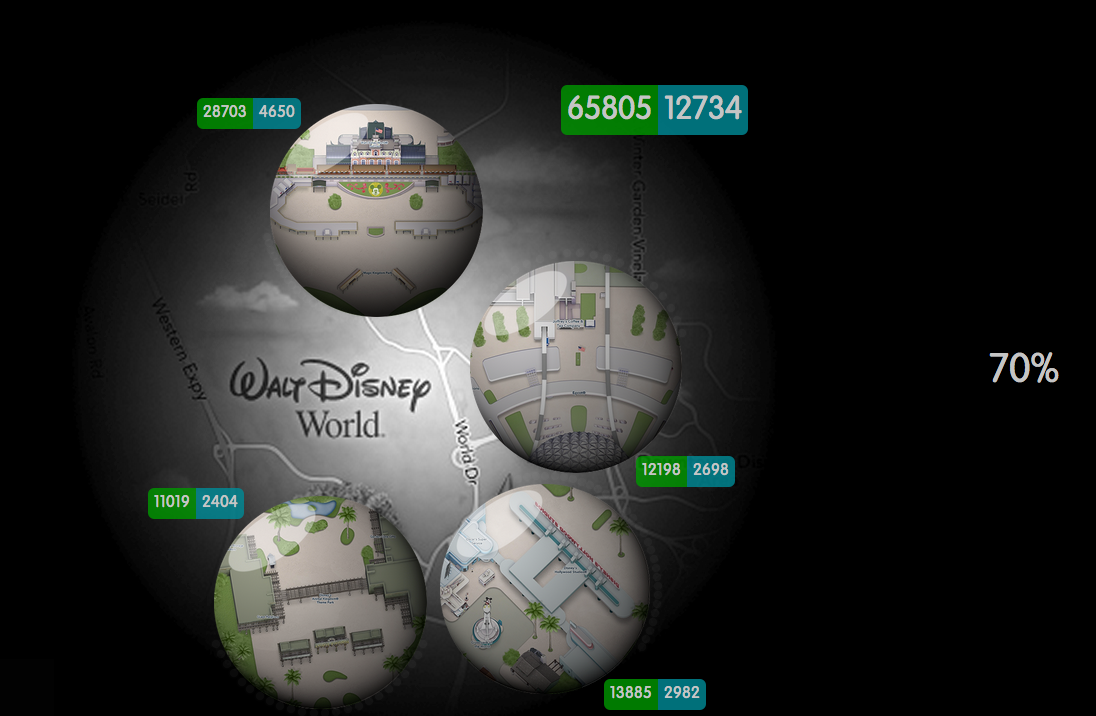
Recruitment Screen

1. The Daily Recruiting Report is a stand-alone page that displays the total number of Guests that are due to arrive for a current day, their distribution throughout the Parks, and how early they booked their experiences.



Park Entry

1. Park Entry shows Green lane and Blue lane occurrences at a Park and World level.



Attraction Tie-Outs

1. Reader locations for MK are displayed on a map of the Park. Reader alerts are highlighted in white to alert a Cast Member that there may be an issue with a reader. Drill down into the issue by clicking on the attraction icon



# xi Web Service Interfaces

## Executive Summary

The Executive Summary Web Service Interface allows the user interface to request executive summary data via Ajax using the JSONP (JavaScript Object Notation with Padding) protocol. A rest style is used for parameters.

## Entitlements

The Entitlements Web Service Interface allows the user interface to request entitlement data via Ajax using the JSONP (JavaScript Object Notation with Padding) protocol. A rest style is used for parameters. Data can be retrieved by facility and by hour.

## Blue lanes

The Blue Lanes Web Service Interface allows the user interface to request blue lane data via Ajax using the JSONP (JavaScript Object Notation with Padding) protocol. A rest style is used for parameters.

## Recruitment

The Daily Recruiting Report Web Service Interface allows the user interface to request recruiting data via Ajax using the JSONP (JavaScript Object Notation with Padding) protocol. A rest style is used for parameters.

## Park Entry

The Park Entry Web Service Interface allows the user interface to request Park Entry data via Ajax using the JSONP (JavaScript Object Notation with Padding) protocol. A rest style is used for parameters.

# RTD/xGS Contacts

|  |  |  |
| --- | --- | --- |
| Name | Role | Email |
| Mark Hadland | Stakeholder | [markh@synapse.com](mailto:markh@synapse.com) |
| Michael Jungen | Stakeholder | michael.g.Jungen@disney.com |
| Ian Wehmeyer | PM | [ianw@synapse.com](mailto:ianw@synapse.com) |
| Amar Terzic | Software Engineer | [amart@synapse.com](mailto:amart@synapse.com) |
| Robert Lantry | Software Engineer | [robertl@synapse.com](mailto:robertl@synapse.com) |
| Stephen Madson | Software Engineer | [stephenm@synapse.com](mailto:stephenm@synapse.com) |
| James Frances | Software Engineer | jamesf@synapse.com |
| Craig Rosenberg | Software Engineer | craigr@synapse.com |
| Nathan Ivey | Industrial Engineer | [nathan.j.ivey@disney.com](mailto:nathan.j.ivey@disney.com) |
| Bo Rankin | Industrial Engineer | john.rankin@disney.com |
| Jenn Fishpool | Industrial Engineer | jennifer.p.fishpool@disney.com |

# RTD Data store

RTD consumes data from three different databases within the SQLServer database instance. We consume event and guest data from the rdr database which is fed by XBRMS. We also consume data from the GxP database which is fed by the JMS listener. The third database contains only support tables and the stored procedures for xi.

