

# Documentation



Pages here should be assumed to pertain to the **latest release** (currently 1.8 MS2.03).  
For documentation about **older releases**, see the [archive](#).

*This documentation is a work in progress. We are grateful to the many people who have contributed to improving it. Special thanks to eagle-i's early adopter community, and in particular to Karen Hanson, Will Foushee, Sukie Punjasthitkul, Melissa Binde and Ross Davis. Please keep your comments coming.*

## System Overview

- **System Overview:** This document provides a high-level overview of the system. It describes the ontology-centric architecture, an institutional deployment and the eagle-i software stack including the repository, data tools, search and ontology/data model.

## System Administrators

### Installation and Maintenance

This section covers tasks necessary to install and maintain an eagle-i institutional server, or *eagle-i node*.

- **What is an eagle-i node?**
- **Quick Start Installation Guide:** provides a high-level overview of the steps necessary to set up an eagle-i node, with pointers into the detailed guides below.
- **Repository Installation, Upgrade and Administration Guide:** provides instructions and information for installing the repository. It also covers general repository maintenance, logging and performance monitoring.
- **SWEET and Institutional Search Installation and Upgrade Guide:** provides installation instructions and information for SWEET (Semantic Web Entry and Editing Tool), institutional search and the online help.
- **Public SPARQL Endpoint:** provides installation and maintenance instructions for the Sparqler, our public SPARQL endpoint (optional component).
- **Configuration Property Guide:** summarizes configuration properties for SWEET, Sparqler and Institutional Search.
- **Upgrade Checklist:** summarizes the steps necessary to upgrade the software and ontology of an eagle-i institutional server

### Advanced Topics

- **Data Management Guide:** describes the data management toolkit for performing data changes at the triple level.
- **SWIFT and ETL Guide:** includes a system administrator's perspective of the tasks necessary to bulk-upload data into an eagle-i repository.
- **CURL Cookbook - Repository API Examples+:** This page describes CURL commands to interact with the Repository via its REST API
- **Repository Data Model Configuration Properties:** This page describes the way a *data model ontology* is configured into the repository. **The defaults are usually OK.**

## Account Administrators

- **Account Administration Handbook:** This document describes the default repository roles and permissions.

## Developers

### Web Services

- **Web Services Provided by eagle-i Nodes:** Describes public web services for retrieving eagle-i data.
- **Web services Provided Centrally:** Describes public web services for obtaining ontology information.

### Ontology

- [eagle-i resource ontology home](#)
- [Guide to application ontologies](#)

## Repository

- **Repository Design Specification and API\***: This document is intended to be a thorough description of the data repository design; concentrating on its external interfaces. The intended audience is repository coders as well as implementers of other components that depend on the repository. It also serves as an API reference manual.
- **Repository Workflow Design Notes**: Contains requirements, use cases, ontology design principles, access control and API for eagle-i workflow.
- **Repository Embedded Instances Design Notes**: This page describes how *embedded instances* are to be managed by the repository.

## Applications

- **Sweet Developers Guide**: This is a developers' guide to the eagle-i data tools. It's focus is primarily on the SWEET (Semantic Web Entry and Editing Tool).
- **Search Application Developers Guide**: This is a developers' guide to the eagle-i search application. It covers the search backend and Solr.

## Javadocs

- **Current release** : <http://search.eagle-i.net/javadoc>
- **Nightly build** : <http://qa.search.eagle-i.net/javadoc>