Release Notes

- **Version 1.9.3**
  - **Improvements**
    - ORCiD API v2
    - Search Results Performance
    - Direct2Experts Endpoint
    - Improved handling for TDB triple stores
  - **Bug Fixes**
  - **Resolved Issues**

- **Version 1.9.1**
  - **Improvements**
  - **Bug Fixes**
  - **Resolved Issues**

- **Version 1.9.0**
  - **What’s New**
    - Capability Map
    - SEO Improvements
    - AltMetrics Improvements
    - Maven Project Structure
    - Maven Based Installer
    - OpenJDK Support
    - Performance Improvements
    - SPARQL Editor Highlighting
  - **Contributors**
  - **Resolved Issues**

**Version 1.9.3**

**Improvements**

**ORCiD API v2**

This release includes an updated ORCiD integration that can use the ORCiD v2 API. Note that ORCiD are planning to shut down the v1.x API endpoints.

Note that the configuration options have been changed, and you will need to update your runtime.properties.

The only options that are required now are:

```
# orcid.clientId = 0000-0000-0000-000X
# orcid.clientPassword = 00000000-0000-0000-0000-000000000000
# orcid.webappBaseUrl = http://localhost:8080/vivo
# orcid.externalIdCommonName = VIVO Cornell Identifier
# orcid.apiVersion = 2.0
# orcid.api = sandbox
```

orcid.apiVersion is simply the version value (e.g. 1.2, 2.0), and orcid.api is just "release" (for the production API), and "sandbox" for the sandbox.

**Search Results Performance**

Retrieval of Person information has been improved so that any search that returns people is faster.

**Direct2Experts Endpoint**

For any sites wishing to participate in the Direct2Experts federated site - [http://direct2experts.org/](http://direct2experts.org/) - VIVO now includes the necessary endpoints. Please see the Direct2Experts websites for more information on how to participate.

**Improved handling for TDB triple stores**

Isomorphic tests for TDB now handle all of the specialized integer types (NonNegative, Positive, etc) that TDB maps to an internal structure.

Performance for TDB triple stores is improved as the Graph URIs are now cached in the RDF Service.
Version 1.9.2

Improvements
Upgraded GEMET integration to use SSL web service
Upgraded LCSH integration to use current web service

Bug Fixes
Problem with creating new items
Add dependencies for ORCID integration
Spaces in prefix definitions

Resolved Issues

Version 1.9.1

Improvements
Responsiveness of Capability Map
Add Turtle support to file graph loader

Bug Fixes
schema.org errors on profile pages
Maven character set issues on Windows
Maven correctly sets Java 1.7 version
Use background RDFService correctly

Resolved Issues

Improvements

Bug Fixes

Version 1.9.0

What's New

Capability Map

To address the desire for improved expert finding features in VIVO, we are pleased to introduce a capability map. This feature allows you to search for research areas, and see the relationship between them and the researchers.

Initially developed at the University of Melbourne, the code has been updated to remove dependencies on third party search engines. In this implementation, it is using the improved visualisation architecture that was introduced in 1.8.1. This means that the results are coming straight out of the triple store; however, the results may be subject to caching, in the same way as the Map of Science, Temporal Graph, etc.

Notes for Upgrading

The capability map is only linked to from the menu bar. As this is configured through the triple store, and the initial definitions are in a file that only gets loaded the first time VIVO is run, upgrading applications will not have the link present.

To access the capability map in an upgraded VIVO, you need to browse to the following url path:

`<vivo>/vis/capabilitymap`

This can be added to an existing application's menu bar via the Site Admin pages. See Upgrading VIVO for details.

Initial implementation provided by: Simon Porter, Matj Korvas, Martin Kwok, and Melissa Makin; University of Melbourne.

Adapted for VIVO 1.9 by: Graham Triggs; DuraSpace.

SEO Improvements

For better indexing and discoverability of your VIVO installation, a sitemap generator is included - in this release, only profile pages are included in the sitemap.
Additionally, citation meta tags are included on the pages of works.

**AltMetrics Improvements**

Following the addition of AltMetric badges to publications with DOIs in 1.8.1, the support has been expanded to cover PubMed IDs and ISBNs. Also, it is now possible to see AltMetric badges on the publication lists within a profile page.

**Maven Project Structure**

To make it easier for new developers to get started with VIVO, the custom Ant scripts have been replaced with standard Maven project structures. Both Vitro and VIVO have been migrated. As a developer, when you clone the projects from GitHub, you should place them in directories next to each other. E.g.:

```
/projects
/Vitro
  /pom.xml
  /api
  /....
/VIVO
  /pom.xml
  /api
  /....
```

With this layout, you only need to tell your IDE to load or import the pom.xml in the "VIVO" project, and it will automatically load in all of the other projects, including Vitro, setting up your IDE ready to start work with full autocompletion, etc.

**Maven Based Installer**

As a result of the move to Maven, there is now an "installer" project, which will assemble the application and home directories, and copy them to your Tomcat and installation / home directories.

The installer will automatically download all of the necessary dependencies - including pre-built Vitro and VIVO code - in order to complete the installation.

This installer also provides a natural place for sites to add their customisations: e.g. a custom theme or even additional Java classes in the web application; RDF in the home directory, etc.

**OpenJDK Support**

The image processor for uploading and generating thumbnails has been replaced with a new library. As a result, there are no known OpenJDK incompatibilities.

Note: It is advised that you use Java 8 for the best performance.

⚠️ When upgrading, you will need to update your applicationSetup.n3 in order to use the new Image Processor.

**Performance Improvements**

Thanks to Brian Lowe, there are significant improvements writing data to the triple store, with an updated inferencer batching changes in memory.
There are also minor improvements to graph comparison code (improve startup time for large graphs), and memory reductions for the caching of data in the visualisations.

**SPARQL Editor Highlighting**

The YASQE highlighting editor for SPARQL has been incorporated into the SPARQL Query page, thanks to a contribution from Ted Lawless.

```sql
PREFIX vivosocnet: <http://vivo.onsi.iu.edu/ns#>
PREFIX core: <http://vivoweb.org/ontology/core#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX bibo: <http://purl.org/ontology/bibo/>
PREFIX socsci: <http://vivo.library.cornell.edu/ns/vivo/socsci/0.1#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
SELECT ?person (COUNT(?relationship) AS ?count) WHERE {
  ?person a foaf:Person .
} GROUP BY ?person ORDER BY DESC(?count)
```

**Contributors**

Ariel David Moya Sequeira, IT Alkaid Consulting S.A.
Stephan Zednik, Rensselaer Polytechnic Institute
Christian Hauschke, Bibliothek der Hochschule Hannover
Roberto J. Rodrigues, Universidade Federal do Rio de Janeiro
Benjamin Gross, UNAVCO
Chad Nelson, Temple University Library
Brain Lowe, Ontocale
Ted Lawless, Thomson Reuters
Graham Triggs, DuraSpace
Nate Prewitt, CU Boulder
Jim Blake, Cornell
Tim Worrall, Cornell
Rebecca Younes, Cornell
Huda Khan, Cornell
Joe McInerney, Cornell
John Fereira, Cornell
Alvin Hutchison, Smithsonian
Simon Porter, Digital Science / University of Melbourne
Matj Korvas, University of Melbourne
Martin Kwok, University of Melbourne
Melissa Makin, University of Melbourne

**Resolved Issues**
Features

- VIVO-1082
- VIVO-1249
- VIVO-1237

Improvements

- VIVO-1222
- VIVO-1236
- VIVO-1217
- VIVO-1234
- VIVO-1242
- VIVO-1245
- VIVO-1221
- VIVO-1228
- VIVO-1232
- VIVO-1233
- VIVO-1238
- VIVO-1240
- VIVO-1243

Bug Fixes

- VIVO-1218
- VIVO-1223
- VIVO-1235
- VIVO-1241
- VIVO-1253
- VIVO-1224